Mathematical Foundations for the use of Genetic Algorithms in Economic Models

The University of Adelaide

Department of Applied Mathematics

Scott Wheeler

B.Sc (Ma. & Comp. Sc) (Hons)

Thesis submitted for the degree of Doctor of Philosophy

July, 2002

Declaration

I declare that to the best of my knowledge this document contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text.

I give permission for the University of Adelaide Library to make available this copy for loan or photocopy.

Signed:

Date: July 31, 2002

Acknowledgements

I would like to thank the members of staff at the University of Adelaide, Department of Applied Mathematics. In particular, I wish to acknowledge Dr. Elizabeth Cousins, Dr. Janice Gaffney, and Assoc. Prof. Peter Taylor. I also extend thanks to Assoc. Prof. John Duffy, University of Pittsburgh, and Assoc. Prof. Herbert Dawid, University of Southern California.

Abstract

In this dissertation, we aim to provide a mathematical foundation for the application of genetic algorithms to economic models.

In order to analyse the dynamics of a genetic algorithm in an economic application, we model the representative discrete time stochastic process as a homogeneous Markov chain. The transition matrix for this Markov chain exhibits a number of properties depending on each individual model, the genetic operators, and other parameters. In particular, conditions for the existence and uniqueness of equilibria, and for convergence and stability in the economic system, can be derived from the communication and recurrence properties of the transition matrix. Two concepts from Markov theory are essential to our analysis. First, partitioning states of the Markov chain into disjoint communicating classes is fundamental in establishing the solidarity properties of recurrence or transience for all states within a communicating class. The correspondence between absorbing states and economic equilibria, stable or unstable, is particularly relevant in this approach. Second, by analysing the hitting probabilities of a Markov chain on given states we gain insight into the probability that an economic model will reach equilibrium.

Following the work of Arifovic [3], we illustrate how our theoretical results can be applied in practice in the context of a cobweb model. Using an approach based on Markov chains, we derive conditions for convergence with probability

one. We compare these conditions to the local asymptotic stability conditions derived by Dawid [24]. Calculations of stationary distributions of our chain for sufficiently small state spaces allows us to discuss the rate of convergence and the asymptotic properties of the model. We also consider two overlapping generations economies. The first is a model of constant money supply, the second a model of constant real deficit. In such two population models, it is not practical to calculate a stationary distribution of the Markov chain. However, we do statistically analyse market volatility in simulations. In the model of constant real deficit, we discuss an implementation of Marcet and Sargent's [58] least squares learning algorithm and comment on the differences which arise as a result in simulations.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Adaptive Learning and Genetic Algorithms

Economic systems have been studied, modelled, and subsequently analysed in a number of different ways. Traditionally, analytic models have been proposed to explain the complex dynamical behaviour economic systems exhibit. Laboratory experimentation with human test subjects have provided evidence to give insight to these models. However, in some economic models, experimental results and analytic models have not been in agreement. Traditional assumptions inherent to many analytic models, the theory of rational expectations for example, have been questioned and other plausible hypotheses proposed as alternatives to these assumptions. Learning algorithms offered alternative analytic models of adaptation and learning which incorporated these plausible hypotheses for behaviour in economic systems, see for example Lucas [57].

Conventual genetic algorithms were first proposed for use in optimization problems. They were designed to imitate Darwin's [23] theory of natural selection and

evolution. In this sense, genetic algorithms are not truly function optimizers per se, see [29], but are rather random search methods or heuristics.

The interpretation of a genetic algorithm as a metaphor for cognitive intelligence and decentralized learning made it natural to consider its application to economic systems. In such applications, a population of adaptive agents simulates the behaviour of a population of human agents. These adaptive agents formulate strategic decisions about the economy through the application of the genetic algorithm. The genetic algorithm is itself a synergism of evolutionary pressure and natural selection. Hence, a genetic algorithm, as used in economic applications, is considered to be a model for the behaviour of adaptive agents within the system. The evolution of decisions throughout simulations of the genetic algorithm can then be compared with empirical results, analytic models, and other learning algorithms, see for example Arifovic [3].

Despite the acceptance of genetic algorithms in economics, few results have been derived for their application to economic systems. The work of Dawid [26] is a notable exception. In Dawid's work, several local asymptotic stability conditions are derived. We do not follow Dawid's approach but rather concentrate on the absorbtion and communication properties of a Markov chain model for a genetic algorithm. With this approach, we aim to provide a mathematical foundation for genetic algorithms in economic applications and to derive theoretical results for our models. We discuss particular results using simulations to illustrate our theory in practice.

1.2 Literature Review

The cobweb model was first formulated by Ezekiel [36] and Leontief [55]. Since then, cobweb economies have been analysed by a number of authors, see Chiarella

[22] for a comprehensive listing of publications up to 1990. Carlson [19], Holt and Villamil [49], Wellford [73], and Hommes et al. [50] used human test subjects to explore cobweb economies.

Arifovic [3] applied a genetic algorithm to a cobweb model. By incorporating an election operator into her genetic algorithm, Arifovic modelled the behaviour of agents in the economy. She compared her simulation results for a genetic algorithm to learning algorithms based upon price averages [20], naive expectations [36], and least squares learning [58]. She also compared her results to the laboratory experiments of Wellford [73]. Her simulation results showed that agents in a genetic algorithm modelled the qualitative features observed in laboratory tests.

Dawid and Kopel [28] considered a formulation for the cobweb model which incorporated a market entry and exit decision. That is, agents were allowed to enter or leave the market. Stability results were derived for this model and a genetic algorithm using a Markov chain approach developed by Vose and Liepins [72] and Nix and Vose [62]. See Dawid [24], Dawid [26], and Dawid and Hornik [27] for a detailed description of this approach and its application to economic models. This Markov chain approach provided an insight into the asymptotic convergence properties of genetic algorithms.

Brock and Hommes [14] and Goeree and Hommes [42] analysed a non-linear cobweb model. Agents chose from using either a freely provided naive expectations predictor or a costly rational expectations predictor. In these models, different hypotheses about how agents should respond to market stimulus were proposed. An analysis of alternative behaviour for agents within the context of a genetic algorithm was the topic of Franke [39]. Franke's formulation for a genetic algorithm incorporated adaptive expectations of different orders.

Gaffney et al. [40] discussed a cobweb model which was simulated using a modified genetic algorithm. In this study, an alternative coding mechanism, real coding, was

used. Pearce [63] proposed a general formulation for a genetic algorithm studied by Gaffney et al. [40]. He considered discrete and continuous models of this genetic algorithm and analysed their convergence properties.

Samuelson [68] formulated an overlapping generations model of fiat money in 1958. To explore Samuelson's theoretical model, laboratory tests were performed by Arifovic [3], Lim et al. [56], Marimon et al. [60], and Marimon and Sunder [61]. Lucas [57] and Marcet and Sargent [58] analysed the adaptive dynamics of learning schemes which represent alternative models of human behaviour. Azariadis [9], Evans and Honkapohja [34, 35], and Woodford [75] considered sunspot equilibria. In such models, agents believe ex ante that prices depend upon an exogenous random variable taking on values $\{0, 1\}$ according to some stochastic process. Duffy [30] considered an overlapping generations model in disequilibrium and Bullard [15] examined complicated equilibrium trajectories using a least squares learning model.

Arifovic [5] considered two models of the overlapping generations type. The first model examined was of constant money supply, the second model was of constant real deficit. Arifovic [6] explained local stability with reference to Evans and Honkapohja's [34] [35] E-stability results and used statistical techniques in a model of two currencies [8] to analyse equilibria. In these models, genetic algorithms were shown to model two population dynamics and matched the behaviour of laboratory experiments to a greater extent than other learning algorithms.

The overlapping generations model was used by Dawid [25] to investigate the behaviour of genetic algorithms and cyclic equilibria. Dawid's formulation of an overlapping generations model of constant money supply had both cyclic equilibria of second order and non-cyclic equilibria. Dawid showed that a genetic algorithm could capture the cyclic behaviour of an economic model. He extended his results to a formulation of the model which included sunspot behaviour. Dawid also used

the Markov chain formulation of Vose and Liepins [72] and Nix and Vose [62], see also Dawid and Kopel's [28] analysis of a cobweb model, to derive local stability conditions with respect to the expected dynamics of a genetic algorithm in two population models.

Higher order cycles and the evolution of beliefs in an overlapping generations model were considered by Bullard and Duffy [17]. In this formulation of the model, a genetic algorithm was used to explore how agents forecast, see also Bullard and Duffy [16, 18]. Initially agents had heterogeneous beliefs. Agents must co-ordinate their beliefs in an equilibrium to attain an equilibrium. This approach differed from other applications of a genetic algorithms for which the forecast function is fixed. Agents were said to be "learning how to forecast" and selected from a number of alternative beliefs about the nature of the model rather than directly setting quantity decisions.

1.3 Organisation of Thesis

We present all preliminary results in Chapter 2. In Section 2.1, we discuss stochastic processes and Markov chain theory. This section provides a background for the modelling of genetic algorithms as a Markov chain. We introduce some basic definitions from set theory in Section 2.2. Section 2.3 explains the terms, definitions, and results we use to analyse our dynamic systems. This section introduces concepts for the stability of dynamic systems and defines a first order difference equation.

In Chapter 3, we introduce our genetic algorithm and explain the connection between this algorithm and a Markov chain. We give a general overview of evolutionary algorithms and the role a genetic algorithm plays as a model of behaviour for economic systems in Section 3.1. We examine three types of coding mechanisms

and discuss the fitness function of our genetic algorithm in Section 3.2. In Section 3.3, the genetic operators are discussed. This section details the general framework for our algorithm. We also describe how a state transition matrix for our Markov chain model of a genetic algorithm is obtained.

In Chapter 4, we analyse several common economic models. The first economic model we discuss is a cobweb model, as given in Section 4.1. We analyse an overlapping generations model of fiat money with a constant money supply in Section 4.2. The term, fiat money, refers to an irredeemable paper currency with a declared value derived from the authority of the government issuing it. We describe a particular parameterization for this model in Section 4.3. In Section 4.4, we analyse an overlapping generations model of fiat money with a constant real deficit financed through seignorage. Seignorage is defined as revenue raised by the printing of fiat money.

We analyse the Markov chain model, discussed in Chapter 3, for a genetic algorithm applied to an economic system in Chapter 5. General results are derived in Section 5.1. In Section 5.2, we apply these results to the cobweb model presented in Section 4.1. We analyse the rate of convergence of simulations and the stationary distribution of our Markov chain in Section 5.3.

In Chapter 6, we analyse the application of a genetic algorithm in two population economic models and a respective Markov chain model. In Section 6.1, several general results for our Markov chain representation of such models are derived. In Section 6.2, we consider the overlapping generations model of fiat money with a constant money supply as presented in Section 4.2. Volatility between agents decisions is discussed in Section 6.3. In Section 6.4, we analyse the overlapping generations model of fiat money with a constant real deficit as presented in Section 4.4. We discuss least squares learning in Section 4.5.

In Chapter 7, a summary of findings is given. We draw conclusions from previous

finding in Section 7.1 and present some closing remarks in Section 7.2.

Parameter values for simulations are given in Appendix A. Example simulation results are given in Appendix B.

Chapter 2

Preliminaries

2.1 Discrete Time Markov Chains

In this section we introduce some of the basic theory for Markov chains. A detailed description of this theory as it applies to the definitions and results presented here may be found in the texts of Feller [37] and Ross [66].

Stochastic algorithms are often used to model the temporal behaviour of economic models in discrete time. This behaviour is described in terms of a stochastic sequence in discrete time $X(t)$ over some state space $\mathcal{S}$. The stochastic sequence $X(t)$ is said to satisfy the *Markov* or *memoryless property* if the value of the sequence at t depends only on the value at $t-1$. This is formalized in the requirement

$$\mathbb{P}(X(t) = j_t | X(0) = j_0, \ldots, X(t-1) = j_{t-1}) = \mathbb{P}(X(t) = j_t | X(t-1) = j_{t-1}), \quad (2.1)$$

$\forall\ t \geq 1, \forall\ j_0, \ldots, j_t \in \mathcal{S}$.

The Markov chain is *time-homogeneous*, or just *homogeneous*, if $\mathbb{P}(X(t) = j | X(t-1) = i)$ is independent of t. The time-invariant probabilities

$$\mathbb{P}(X(t) = j | X(t-1) = i) = \mathbb{P}(X(s) = j | X(s-1) = i), \ \forall\ s, t \geq 0,\ i, j \in \mathcal{S}, \quad (2.2)$$

are denoted by $p_{i,j}$. These time-invariant transition probabilities define a single step transition matrix $P = [p_{i,j}]$. This non-negative matrix is stochastic since the sum of all entries along each row is one. The m-step transition probabilities $\mathbb{P}(X(t+m) = j | X(t) = i)$ are denoted by $p_{i,j}^{(m)}$. The m-step transition probability matrix is denoted by $P^{(m)} = [p_{i,j}^{(m)}]$.

Now, we can use the law of total probability and the Markov property (2.1) to write

$$\begin{aligned}
&\mathbb{P}(X(m+1) = j | X(0) = i) \\
&\quad = \textstyle\sum_{k=1}^{|\mathcal{S}|} \mathbb{P}(X(m+1) = j | X(m) = k, X(0) = i) \mathbb{P}(X(m) = k | X(0)) = i), \\
&\quad = \textstyle\sum_{k=1}^{|\mathcal{S}|} \mathbb{P}(X(m+1) = j | X(m) = k) \mathbb{P}(X(m) = k | X(0)) = i),
\end{aligned} \tag{2.3}$$

so that

$$p_{i,j}^{(m+1)} = \sum_{k=1}^{|\mathcal{S}|} p_{i,k}^{(m)} p_{k,j}. \tag{2.4}$$

That is, the probability of entering the state j in $m+1$ steps can be expressed as the sum over all states $k \in \mathcal{S}$ of the probability of entering the state k in m steps starting in i and then entering the state j in a single step starting in k.

Hence, the matrix of m-step transition probabilities $P^{(m)} = [p_{i,j}^{(m)}]$ can be calculated as the product of P with itself m times

$$P^{(m)} = P^m. \tag{2.5}$$

If it is possible to reach the state $j \in \mathcal{S}$ from the state $i \in \mathcal{S}$ in a finite number of steps we say j is *accessible* from i and write $i \to j$.

Definition 2.1. *The state $j \in \mathcal{S}$ is accessible from the state $i \in \mathcal{S}$ iff $\exists\ m \in \mathbb{Z}^+$ such that $p_{i,j}^{(m)} > 0$.*

When $i \to j$ and $j \to i$, the states $i \in \mathcal{S}$ and $j \in \mathcal{S}$ are said to *communicate* and we write $i \leftrightarrow j$.

Definition 2.2. *Two states $i \in \mathcal{S}$ and $j \in \mathcal{S}$ communicate iff i is accessible from j and j is accessible from i. A state which communicates with no other state is ephemeral.*

The communication relationship $\leftrightarrow$ is an equivalence relation since it is reflexive, symmetric, and transitive. This relationship can be used to partition non-ephemeral states into mutually disjoint communicating classes. These communicating classes are an indication of the accessibility of states.

If we wish to examine additional properties of the Markov chain a more detailed approach is required. We are often interested in how many times a particular state is expected to be visited or how many transitions are required before a state may be re-visited.

Let $\Lambda = \{k | p_{i,i}^{(k)} > 0\}$. The *period* of state $d(i)$, $i \in \mathcal{S}$, is the greatest common divisor of all members of the set Λ and zero if $\Lambda = \emptyset$. A state is periodic if it has period greater than one and aperiodic otherwise. Periodicity only partially explains the visiting properties of states. A more general approach is now considered.

Denote by $r_{i,j}^{(m)}$ the probability that the Markov chain enters the state $j \in \mathcal{S}$ for the first time in exactly m steps conditional on starting in state $i \in \mathcal{S}$. These probabilities are called the *first entrance probabilities* if $i \neq j$ and *first return probabilities* if $i = j$.

Definition 2.3. *A state $i \in \mathcal{S}$ is recurrent iff $\sum_{t=1}^{\infty} r_{i,i}^{(t)} = 1$ and transient iff $\sum_{t=1}^{\infty} r_{i,i}^{(t)} < 1$.*

For any $i \in \mathcal{S}$, the events that the Markov chain first visits the state i for different values of $m \in \mathbb{N}$ are mutually exclusive. Thus, the sum $\sum_{t=1}^{\infty} r_{i,i}^{(t)}$ is the probability that the Markov chain ever returns to the state i conditional on starting in i.

A recurrent state may exhibit two types of behaviour. While all recurrent states $i \in \mathcal{S}$ are such that sum $\sum_{t=1}^{\infty} r_{i,i}^{(t)} = 1$, the expected re-visit time may not be finite.

To differentiate between states with a finite and an infinite expected re-visit time an additional classification is necessary.

Let T_i be the random variable which denotes the time that it takes to return to the state i given that the Markov chain starts in that state. We set $T_i = \infty$ if the state i is ephemeral. Then $E[T_i] = \sum_{t=0}^{\infty} tr_{i,i}^{(t)}$ is the expected re-visit time for state $i \in \mathcal{S}$.

Definition 2.4. *A recurrent state $i \in \mathcal{S}$ is positive recurrent iff $E[T_i] < \infty$ and null recurrent iff $E[T_i] = \infty$.*

There is one situation for which the positive recurrence of a state can immediately be identified. If the probability of ever leaving a state once entered is zero then we call that state *absorbing.*

Definition 2.5. *A state $i \in \mathcal{S}$ of a homogeneous Markov process with transition matrix P is absorbing iff $p_{i,i} = 1$.*

Communication between states can be used to show that all states in a communicating class have shared properties. The classification of states can then be approached at the level of a communicating class. That is, null recurrence, positive recurrence, transience, and periodicity are solidarity properties of a communicating class.

Theorem 2.1. *If $i, j \in \mathcal{S}$ belong to the same communicating class $\mathscr{C}$ then*
(a) i and j have the same period,
(b) i is transient iff j is transient,
(c) i is null recurrent iff j is null recurrent,
(d) i is positive recurrent iff j is positive recurrent.

Proof: See Grimmett and Stirzaker [45], pp. 204–205. ◇

When applying stochastic methods to economic models it might reasonably be

expected that the behaviour of the economic model should in some way be related to the communicating class structure of the Markov chain. In particular, absorbing states of the Markov chain should represent stationary equilibria of the economic model. The probability of eventual entry of the Markov chain into a particular recurrent class is also important.

In practice it may be difficult to determine whether a communicating class is null recurrent, positive recurrent, or transient. However, when the state space of a Markov chain is strictly finite we can classify a communicating class by considering classes of *essential* and *inessential* states.

Definition 2.6. *A state $i \in \mathcal{S}$ is inessential iff $i \to j$ and $j \not\to i$ for at least one $j \in \mathcal{S}$ and essential otherwise.*

Theorem 2.2. *If the state space of a homogeneous Markov chain is finite then*
(a) there exists at least one positive recurrent class,
(b) all states in every recurrent class are positive recurrent,
(c) every essential class is a positive recurrent class.
(d) every inessential class is a transient class.

Proof: See Grimmett and Stirzaker [45], pp. 204–206. ◇

By Theorem 2.2 we know that in a finite state Markov chain the positive recurrence, null recurrence or transience of communicating classes can be determined by examining the communication between states. Let us consider a Markov chain in which all states communicate with the entire state space $\mathcal{S}$.

Definition 2.7. *A homogeneous Markov process with transition matrix P and state space $\mathcal{S}$ is irreducible iff for every pair of states $i, j \in \mathcal{S}$, $\exists\ k \in \mathbb{Z}^+$ s.t. $p_{i,j}^{(k)} > 0$.*

We differentiate between irreducible matrices which are periodic and aperiodic by

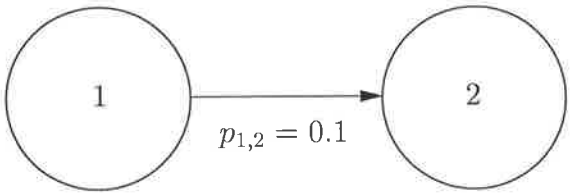
defining a *primitive* matrix.

Definition 2.8. *A matrix P is primitive iff* $\exists\ k \in \mathbb{Z}^+$ *s.t.* $P^t > 0,\ \forall\ t \geq k$.

Theorem 2.3. *A time invariant Markov chain has a primitive state transition matrix P iff the state transition matrix P is irreducible and aperiodic.*

Proof: See Seneta [69], pp. 19–20. ◇

A Markov chain which has entered a positive recurrent class will remain within the class ad infinitum. If the state transition matrix is irreducible we know that the entire state space is a single recurrent class. However, it is not immediately obvious which states a Markov chain can be expected to inhabit in the long term if the state transition matrix is not irreducible. In particular, it is not obvious why a Markov chain should not be expected to remain in transient states for all time if one or more transient states exist.



$$P = \begin{array}{c} \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{bmatrix} 0.9 & 0.1 \\ 0.0 & 1.0 \end{bmatrix} \end{array}$$

Figure 2.1: Example chain

Let us consider the Markov chain with a single distinct recurrent class given by Figure 2.1. The state 1 is a self-communicating inessential state, $1 \leftrightarrow 1$. There is a single positive recurrent class $\mathscr{C} = \{2\}$ and this class is accessible from the entire state space. It is clear there is a particular realization of the Markov chain which never leaves the state 1, the realization $\{1, 1, 1, \ldots\}$. However, the probability that the recurrent class $\mathscr{C}$ is entered grows geometrically with time and approaches one asymptotically. To see this, note that the probability of remaining in the state 1 after a single transition is $p_{1,1} = 0.9$. The probability of remaining in the state 1 after two transitions is $p_{1,1}^{(2)} = 0.9^2$. The probability of remaining in the state 1 for

ever is $\lim_{t\to\infty} 0.9^t = 0$. We deduce that the recurrent class $\mathscr{C} = \{2\}$ is entered *with probability one.*

In our example there is only one realization of the Markov chain which does not enter the state 2. It is easy to imagine an arbitrary Markov chain with infinitely many possible realizations which involve inessential states only. Markov theory tells us that the probability of this set of realizations is zero.

Our example, given in Figure 2.1 above, illustrates an instance of convergence with probability one which assigns a probability mass of one to the state 2 and zero to all others. We say the chain *converges* to the state 2 *with probability one.*

Definition 2.9. *A stochastic sequence $X(t)$ is convergent to the state $j \in \mathcal{S}$ with probability one if $lim_{t\to\infty} X(t) = j$ with probability one.*

For a finite state Markov chain we can establish the following result.

Theorem 2.4. *A finite state Markov chain has a finite mean absorption time and converges to the state $i \in \mathcal{S}$ with probability one iff i is accessible from all states $j \in \mathcal{S}$ and i is absorbing.*

Proof: See Kemeny and Snell [53], pp. 43, 46. ◇

We now have the tools to discuss the long term behaviour of a Markov chain and to consider the stationary behaviour of the chain. We call the distribution π *stationary* if

$$\pi = \pi P. \tag{2.6}$$

Hence, $\pi = \pi P^m$, for all $m \geq 1$.

Let $F^{(m)} = (f_1^{(m)}, \ldots, f_{|\mathcal{S}|}^{(m)})$ denote the distribution of $X(m)$. Then, it follows from equation (2.5) that

$$F^{(m)} = FP^m, \tag{2.7}$$

where $F = F^{(0)}$ is the initial distribution of the chain. A sequence of distribution functions $\{F^{(m)}\}$ is said to *converge in distribution* if the sequence approaches some limiting value, $\lim_{t\to\infty} F^{(t)} = \pi$, for some distribution $\pi = (\pi_1, \ldots, \pi_{|\mathcal{S}|})$. All such limiting distributions must be stationary. However, it does not follow that all stationary distributions are also limiting.

Theorem 2.5. *An irreducible positive recurrent Markov chain has a unique stationary distribution* $\pi = (1/E[T_i])$, $i \in \mathcal{S}$.

Proof: See Grimmett and Stirzaker [45], p. 208. ◇

Hence, by Theorem 2.5, an irreducible and positive recurrent Markov chain will converge in distribution to $(1/E[T_i])$, $i \in \mathcal{S}$. When the chain is also aperiodic a stronger result applies.

Theorem 2.6. *An irreducible and aperiodic positive recurrent Markov chain converges in distribution to* $\pi_j = \lim_{t\to\infty} p_{i,j}^{(t)}$, *for all pairs* $i, j \in \mathcal{S}$.

Proof: See Whittle [74], p. 172. ◇

These theorems for convergence in distribution can be generalized to chains which are not irreducible, see Karlin and Taylor [52] pp. 89–92. Let us consider an arbitrary Markov chain containing both positive recurrent and transient classes. Label each positive recurrent class $1 \ldots n$. Each positive recurrent class $\mathscr{C}^{[1]}, \ldots, \mathscr{C}^{[n]}$ is then itself an irreducible Markov chain independent of any other recurrent class with a corresponding stationary distribution $\pi^{[l]}$, $l = 1 \ldots n$. For every $l = 1 \ldots n$ and every $j \in \mathscr{C}^{[l]}$, let $\pi_j^{[l]}$ denote the value corresponding to the state j from the l^{th} stationary distribution $\pi^{[l]}$. Let $\alpha_i^{[l]}$, $l = 1 \ldots n$, denote the probability that our original chain ever reaches the l^{th} recurrent class conditional on starting in the state i. We know that with probability one, $\lim_{t\to\infty} p_{i,i}^{(t)} = 0$, for every transient state $i \in \mathcal{S}$, and $\sum_{l=1}^{n} \alpha_i^{[l]} = 1$, for every $i \in \mathcal{S}$. Hence, $\alpha_i^{[l]}$, $i \in \mathcal{S}$, are probability distributions over the state space $\mathcal{S}$ which assign a value of zero to every transient

state. For $i \in \mathcal{S}$, we can see that $\lim_{t\to\infty} p_{i,j}^{(t)} = \alpha_i^{[l]}\pi_j^{[l]}$, for every $j \in \mathscr{C}^{[l]}$, $l = 1 \ldots n$, and $\lim_{t\to\infty} p_{i,k}^{(t)} = 0$, for every transient state $k \in \mathcal{S}$.

A stationary distribution can be interpreted in two ways according to the *Ergodic Theorem for Markov chains*, see Karlin and Taylor [52], pp. 487–488. First, a stationary distribution gives the asymptotic probability that the Markov chain is in each state. Second, a stationary distribution gives the asymptotic proportion of time spent in each state. See Billingsley [11] and Kingman and Taylor [54] for more on the convergence of probability measures.

The value of P^k for stochastic primitive matrices can be calculated using *Perron-Frobenius theory*, see for example Seneta [69], pp. 3–11.

Theorem 2.7. *The following properties apply to stochastic primitive matrices. There exists a Perron-Frobenius eigenvalue r with left and right Perron-Frobenius eigenvectors v_L and v_R respectively such that*
(a) $r = 1$ with multiplicity one,
(b) $r > |\lambda|$ for any eigenvalue $\lambda \neq r$,
(c) r has a strictly positive left eigenvector,
(d) r has a right eigenvector $[1, \ldots, 1]^\top$.

Proof: See Seneta [69], pp. 3–7. ◇

The stationary distribution π can be calculated as the normalized Perron-Frobenius left eigenvector. Label the n distinct eigenvalues of a stochastic primitive matrix in decreasing order $r > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|$. It can be shown that the Markov chain approaches the stationary distribution geometrically with time depending on the magnitude of the second largest eigenvalue λ_2.

Theorem 2.8. *Let P be a $n \times n$ stochastic primitive matrix with Perron-Frobenius eigenvalue r with corresponding positive normalized left eigenvector v_L, second largest magnitude eigenvalue λ_2 with multiplicity m_2, and right eigenvector $v_R =$*

$[1, \ldots, 1]^\top$, *then*

(a) if $\lambda_2 \neq 0$, *then elementwise as* $t \to \infty$

$$P^t = v_R v_L^\top + O(t^{m_2 - 1}|\lambda_2|^t), \tag{2.8}$$

(b) if $\lambda_2 = 0$, *then for* $t \geq n - 1$

$$P^t = v_R v_L^\top, \tag{2.9}$$

where v_L *is normalized such that* $v_L^\top v_R = 1$.

Proof: See Seneta [69], pp. 9–11. ◇

2.2 Set Notation

In this section we introduce some of the definitions commonly used in set theory. This theory may be found in the texts of Hausdorff [46] and Pinter [64].

We shall assume that all sets are in the k dimensional Euclidean space $\mathbb{R}^k$. In this space we assume the metric function $\rho : \mathbb{R}^k \times \mathbb{R}^k \to \mathbb{R}$ is an Euclidean distance $\rho(x, y) = \sqrt{\sum_{i=1}^k (x_i - y_i)^2}$.

Definition 2.10. *A subset of* $\mathbb{R}^k$ *is*

(a) null or empty and denoted by $\emptyset$ *iff the set contains no elements,*

(b) countable iff the set has a one-to-one correspondence with $\mathbb{N}$.

We will denote the number of elements in a countable set by $|\cdot|$.

An ϵ-*neighbourhood* of b is a defined as ball of radius ϵ in $\mathbb{R}^k$. That is, $\{x \mid \rho(x, b) < \epsilon\}$. We denote such a ball by $\mathscr{B}(b, \epsilon)$. On the real number line this is the open interval $(b - \epsilon, b + \epsilon)$.

Let Λ be an arbitrary set in $\mathbb{R}^k$. A point $b \in \mathbb{R}^k$ is a *boundary point* of Λ if every non-empty neighborhood of b intersects Λ and its complement. The set of all such

points is called the *boundary* of Λ.

Definition 2.11. *A non-empty subset Λ of $\mathbb{R}^k$ is*
(a) closed iff the set contains all its boundary points,
(b) bounded iff $\exists\ b \in \Lambda$, $\epsilon > 0$, such that $\Lambda \subset \mathscr{B}(b, \epsilon)$.

Let us define the *diameter* of a bounded, closed, and non-empty set as follows.

Definition 2.12. *The diameter of a bounded closed non-empty set $\Lambda \subset \mathbb{R}^k$ is defined as $diam(\Lambda) = max_{x,y \in \Lambda} \rho(x, y)$.*

We define *convexity* in the usual manner, see for example Valentine [71].

Definition 2.13. *A non-empty set $\Lambda \subseteq \mathbb{R}^k$ is convex iff for every pair of values $x, y \in \Lambda$ and every $\lambda \in [0, 1]$, $\lambda x + (1 - \lambda)y$ is a member of Λ.*

We shall denote set intersection, union, and exclusion by $\cap$, $\cup$, and $\setminus$ respectively.

2.3 Discrete Time Dynamic Systems

In this section we introduce some of the basic theory for discrete time dynamic systems. The definitions and results we introduce here can be found in the texts of Dawid [26], Elaydi [33], and Giancarlo [41].

A dynamic system in discrete time is a set of equations which describe the behaviour of some system in discrete intervals. A difference equation on some subset $\mathscr{D} \subseteq \mathbb{R}^k$ is a dynamic system which defines the current state of the system in terms of past values. We say the difference equation is of order n if the current state of the system is a function of the past n states. Let $x_t \in \mathscr{D}$, $t \geq 0$, and $F : \mathscr{D} \to \mathscr{D}$. Then

$$x_t = F(x_{t-1}), \forall\ t \in \mathbb{Z}^+, \tag{2.10}$$

denotes a *first order difference equation.*

We denote the j^{th} iterate of the difference equation (2.10) starting in x_0 by $F^j(x_0)$. We call the sequence $\{x_t\}_0^\infty$ a *trajectory* or *solution* of the difference equation (2.10).

A *fixed point* $x \in \mathscr{D}$ of this difference equation satisfies

$$x^* = F(x^*). \tag{2.11}$$

A fixed point of a difference equation can be characterized as *stable* or *unstable* by introducing a small perturbation to the system about the fixed point. If after an arbitrary perturbation, the system remains in some neighbourhood of the fixed point for all time then we say that the fixed point is stable.

Definition 2.14. *A fixed point $x^* \in \mathscr{D}$ of the difference equation (2.10) is stable iff for any $\epsilon > 0$, $\exists\ \delta > 0$, such that $F^t(x_0) \in \mathscr{B}(x^*, \delta)$, $\forall\ x_0 \in \mathscr{B}(x^*, \epsilon)$, $\forall\ t \in \mathbb{Z}^+$.*

We call a fixed point which is not stable, unstable.

The stability of a fixed point does not tell us much about the limiting behaviour of the system except that a trajectory which is perturbed a sufficiently small distance from a stable fixed point is confined to some neighbourhood of the fixed point for all time. Our Definition 2.14 does not make a distinction between a trajectory which converges to some limiting value and a trajectory which does not. We differentiate between these two situations by defining the *asymptotic stability* of a fixed point.

Definition 2.15. *A fixed point $x^* \in \mathscr{D}$ of the difference equation (2.10) is locally asymptotically stable iff the point x^* is stable and $\exists\ \epsilon > 0$ such that $lim_{t\to\infty} F^t(x_0) = x^*$, $\forall\ x_0 \in \mathscr{B}(x^*, \epsilon)$.*

Definition 2.15 is a local concept of stability. We now consider the asymptotic behaviour of all possible trajectories of a difference equation. That is, a global concept of stability.

Definition 2.16. *A fixed point $x^* \in \mathscr{D}$ of the difference equation (2.10) is globally asymptotically stable iff the point x^* is stable and $lim_{t\to\infty} F^t(x_0) = x^*$, $\forall\ x_0 \in \mathscr{D}$.*

The stability of a difference equation may be difficult to determine. Let us consider the linearization of F using a Taylor series expansion about a fixed point x^* of the difference equation (2.10)

$$F(y) = x^* + (y - x^*)A + G(y), \tag{2.12}$$

where A is a $k \times k$ matrix and $G : \mathscr{D} \to \mathscr{D}$ is a continuous function.

This gives the linearized system

$$F(y) = x^* + (y - x^*)A. \tag{2.13}$$

The stability of the linearized system (2.13) can be used to determine the stability of the original non-linear system (2.12).

Theorem 2.9. *Let x^* be a fixed point of the linearized system (2.13) such that $lim_{y \to x^*} ||G(y)||/||y - x^*|| = 0$ then*

(a) the original non-linear system (2.12) is locally asymptotically stable at x^* if every eigenvalue of A has a magnitude strictly less than one,

(b) the original non-linear system (2.12) is unstable at x^* if at least one eigenvalue of A has a magnitude strictly greater than one,

(c) the stability of the original non-linear system (2.12) at x^* is undetermined otherwise.

Proof: See Dawid [26], p. 174 and Elayda [33], pp. 198, 203. ◇

Chapter 3

Genetic Algorithms

3.1 Evolutionary Algorithms and Economic Models

Evolutionary algorithms are commonly employed as random search methods to provide heuristic solutions in optimization problems, see Alander [2] and Bodnovich and Wong [13]. Each paradigm in this class of evolutionary algorithms share a conceptual design rooted in the principle of natural evolution, see Spears et al [70], Darwin [23]. Individual structures adapt according to a complex evolutionary process modelled by a sequence of selection, reproduction, and mutation operations. A generation of such individual structures forms a basis for a simulation model in which population pressure and competition are an instrument for natural selection and evolution, see Forest [38]. The genetic algorithm, see Goldberg [43] and Holland [48], is a type of evolutionary algorithm.

In economic models a genetic algorithm need not be interpreted as a function optimizer or biological model but more as a model for the behaviour in an agent-based economy. In such a model, an economy populated by human agents is

simulated by a population of artificially intelligent individuals. These artificial agents make strategic decisions about an economic system according to the genetic algorithm and its operators. As the genetic algorithm is itself a model of natural selection and evolution, the genetic algorithm can be interpreted in an economic context as a model of competition and adaptive learning, see Beckenbach [10], Birchenhal et al. [12], Edmonds [31], Edmonds and Moss [32], and Riechmann [65].

3.2 Coding and Fitness

To formulate a genetic algorithm for economic applications, two things must be addressed. First, one must describe the genetic algorithm itself and the genetic operators. We provide these details in Section 3.3. Second, it is necessary to describe which variables in our model are under investigation and how these variables are represented by the genetic algorithm. We can then relate these variables to values in a simulation of our model and study the behaviour of our model over time.

In order to represent numbers in a way that can be manipulated by an algorithm we need to have a method of coding these numbers in a way that can be represented in a computer program. Three ways of doing this are to use a *real code*, a *binary code*, or a *Gray code*.

In real valued coding a variable is represented directly as a number. Double precision, or the equivalent numerical type, is used. It is argued, see for example Gaffney [40], that real coding might realistically model the behaviour of adaptive agents. A description of the genetic operators may be readily interpreted in economic sense when real codes are used.

The numerical precision of a real coded variable is measured in terms of a fixed

number of decimal places accuracy. This fixed number is limited by hardware and software considerations. We will define the *gridspacing* Δ of a coding mechanism to be the difference between two consecutive coded values. For a real coding with n decimal places accuracy this gridspacing is then $\Delta = 10^{-n}$.

Real numbers of fixed precision appear to be a reasonable and obvious representation for a variable. However, a computer translates a real number into a sequence of bits, that is a sequence of binary numbers. The genetic algorithm was first formulated as an evolutionary process to operate on such a sequence.

A binary code of fixed length l is a string $s = (a_{l-1}a_{l-2}\ldots a_1a_0)$, $a_i \in \{0,1\}$, $i = 0\ldots l-1$. The *encoded value* v_e of the string s is

$$v_e = \sum_{j=0}^{l-1} 2^j a_j. \tag{3.1}$$

The number of bits $nBITS$ used by the genetic algorithm to encode values defines the length of the binary code to be used during simulations. Note that $l = nBITS$.

Values for v_e are equally spaced over a domain of the form $\mathscr{D} = [0, M]$. This domain is associated with the domain of some variable under investigation in the economic model. The gridspacing Δ is given by

$$\Delta = \frac{M}{2^l - 1}. \tag{3.2}$$

The *decoded value* $v_d \in \mathscr{D}$ for v_e is

$$v_d = v_e\Delta. \tag{3.3}$$

We define the *Hamming distance* between two binary strings of equal length as the number of bit positions in which the two strings differ in value. It is a property of binary codes that two encoded values which differ by one may have completely dissimilar string representations. Take, for example, two encoded values $v_e^{(1)} = 3$

and $v_e^{(2)} = 4$. Using strings of length three, these values have a string representation of $s^{(1)} = (011)$ and $s^{(2)} = (100)$ respectively. These two strings have a Hamming distance of three. A large Hamming distance between adjacent encoded values is often referred to as a *Hamming cliff*. As noticed by Arifovic [3], this property can adversely affect simulation results. For this reason an alternative coding mechanism has been considered.

Gray coding is a permutation of binary coding. This permutation is done in such a way that two encoded values which differ by one have string representations which differ in exactly one bit position. That is, all adjacent Gray code strings have a Hamming distance of exactly one. In Arifovic's [3] simulations, Gray coding is shown to overcome some of the problems observed when using binary coding. A detailed comparison of binary and Gray coding is given in Section 5.3.

Let $(b_{l-1} \dots b_0)$ be a binary string and $(g_{l-1} \dots g_0)$ a Gray coded string. Let the $\oplus$ operator represent logical xor. To convert from binary coding to Gray coding let $g_{l-1} = b_{l-1}$. Then for $i = l-2, \dots, 0$ let $g_i = b_{i+1} \oplus b_i$. To convert from Gray coding to binary coding let $b_{l-1} = g_{l-1}$. Then for $i = l-2, \dots, 0$ let $b_i = b_{i+1} \oplus g_i$. To give an example, and to demonstrate the Hamming property of Gray codes, let us consider our aforementioned example above with encoded values $v_e^{(1)}$ and $v_e^{(2)}$. The string representations in binary coding of $s^{(1)} = (011)$ and $s^{(2)} = (100)$ become in Gray coding $\hat{s}^{(1)} = (010)$ and $\hat{s}^{(2)} = (110)$ respectively.

We will define a *coded decision*, independent of encoding mechanism, as any member of the set of all coded decisions $\Omega = \{k\Delta | k = 0 \dots m\}$ where $m = M/\Delta$. A *population* is composed of N individuals labelled $i = 1 \dots N$, where N represents the population size $nPOP$ used by the genetic algorithm. We define a *population decision vector* corresponding to the population $(1, \dots, N)$ to be an ordered collection of N coded decisions $(\psi_1, \dots, \psi_N)$, $\psi_i \in \Omega$, $\forall\ i = 1 \dots N$.

Using the notation introduced in Section 2.1, we will define the state space $\mathcal{S}$ of the

stochastic sequence $X(t)$ as the space of all possible population decision vectors Ω^N. That is, each population decision vector represents the decisions made by N individuals in an economic model which is simulated using a genetic algorithm. Following the work of Rudolph [67], a sequence of genetic algorithm population states $X(t)$ can be considered to be a particular instance of a discrete-time homogeneous Markov chain with stochastic transition matrix P and state space $\mathcal{S} = \Omega^N$. Formulations of the Markov chain and its state space vary. An alternative formulation can be found in Dawid [26] and Nix and Vose [62].

Certain states, in which all coded decisions are identical, will be used in subsequent sections, see Dawid [27].

Definition 3.1. *A population decision vector $\psi = \{\psi_i\}$, $\psi \in \mathcal{S}$, is defined to be homogeneous or uniform if each of the individual components $\psi_i \in \psi$, $i = 1 \ldots N$, of the vector ψ are identical. A population decision vector $\psi \in \mathcal{S}$ which is not uniform is defined to be heterogeneous or non-uniform.*

For $k \in \Omega$ the vector $(k, \ldots, k)$ will be denoted by u_k.

We associate with each decision in $\mathscr{D}$ a real number which represents the *fitness* or *payoff* that an individual making that decision would receive conditional on some population decision vector. Let $f_x(\psi)$ be the continuous *fitness function* returning the fitness of any value x in the domain $\mathscr{D}$ conditional on the population decision vector $\psi \in \mathcal{S}$. Note that the value of $f_x(\psi)$ is defined for all $x \in \mathscr{D}$, not only those values which appear in ψ. Also, the value of $f_x(\psi)$ does not depend on which individual $i \in \{1, \ldots, N\}$ evaluates $f_x(\psi)$. We will refer to the fitness function as *state-independent* if the fitness of every decision $x \in \mathscr{D}$ depends only on x and refer to the fitness function as *state-dependent* otherwise.

3.3 Operators

Genetic operators are applied in sequence to construct new generations of individual agents. We shall use Goldberg's [43] and Holland's [48] descriptions of genetic algorithms and operators as a basis for a formulation of the algorithm.

The single population genetic algorithm maintains at each time step $t \geq 0$ a population decision vector which corresponds to the coded decisions made by a population of N individuals. That is each generation index uniquely identifies a population decision vector. A sequence of population decision vectors is generated according to a number of stochastic rules implemented in the form of genetic operators.

The algorithm is described as follows with an exact description of each operator provided below. Let the population decision vector labelled t be given by $\psi(t)$. At $t = 0$ a population of N coded decisions is generated by assigning a value uniformly distributed from Ω to each individual and the population decision vector $\psi(0)$ is formed. At each subsequent generation $t \geq 1$, *selection* operates on the population decision vector $\psi(t-1)$ N times to generate a pool of decisions $\varphi(t) \in \Omega^N$, called the *mating pool.* This mating pool is distinct from our population decision vector in that it is not linked to the state of our Markov process at any time period. The genetic operators, *mutation* and *crossover*, are applied to the mating pool $\varphi(t)$ a fixed number of times with replacement as described below. Finally, a comparison between the population decision vector $\psi(t-1)$ and the mating pool $\varphi(t)$ is conducted by the *election* operator, see Arifovic [3]. The resultant population decision vector $\psi(t)$ is the representative vector of decisions made by individuals at time t. This process is repeated until termination.

Both binary coding and Gray coding use strings in the binary alphabet. Hence, we use the same genetic operators for both mechanisms. We split these genetic operators into three stages during each time period. The first stage operator,

selection, generates a mating pool using the previous population decision vector.

1. *Proportionate selection.* At time $t \geq 1$, decisions are weighted according to fitness. The probability that an individual with decision i is selected from the population $\psi(t-1)$ is $f_i(\psi(t-1))/(\sum_{j\in\psi(t-1)} f_j(\psi(t-1))$. A mating pool of size N is constructed with N independent replications.

Two operators, crossover and mutation, operate on the mating pool during the second stage of our algorithm.

2. *Single point crossover.* Two individuals $a = (a_{l-1} \dots a_0)$ and $b = (b_{l-1} \dots b_0)$ are chosen at random from the mating pool. Crossover is applied with a probability $pCROSS$ and the two individuals remain unchanged if crossover is not applied. If crossover is applied then a point i is randomly chosen between 0 and $l-2$ and two new individuals $\hat{a} = (a_{l-1} \dots a_{i+1}\ b_i \dots b_0)$, $\hat{b} = (b_{l-1} \dots b_{i+1}\ a_i \dots a_0)$ replace a and b respectively in the pool. Crossover is repeated with replacement $\lfloor N/2 \rfloor$ times.

3. *Bitwise mutation.* Denote by $a = (a_{l-1} \dots a_0)$ an individual in the mating pool. The value a_i, $i = 0 \dots l-1$ is replaced with $(a_i + 1) \bmod 2$ with probability $pMUTATE$. Every individual in the pool is subjected to mutation.

We apply election in the final stage of our algorithm at each iteration. The mating pool and the previous population decision vector are compared. The resultant population decision vector is representative of individuals decisions made during that time period.

4. *Election.* Election is applied to all members of the current mating pool and the previous generation of individuals. Individuals are paired from each population in order. Paired individuals are then compared. The individual with highest fitness is accepted into a new population splitting ties randomly.

The real coded genetic operators differ from binary and Gray operators. However, a

three stage framework for our genetic algorithm can still be identified. During each iteration of our algorithm we apply the first stage operator, selection, to generate a mating pool.

1. *Tournament selection.* At time $t \geq 1$, two individuals are selected at random from the population labelled $t-1$. The individual with highest fitness is selected splitting ties arbitrarily. A mating pool of size N is constructed with N independent replications.

We then apply the two second stage operators, crossover and mutation, to the mating pool.

2. *Crossover by inner product.* Two individuals a and b are chosen from the mating pool at random. Crossover is applied with a probability $pCROSS$ and the two individuals remain unchanged if crossover is not applied. If crossover is applied then two new individuals $\hat{a} = (\eta, 1-\eta)\cdot(a, b)$ and $\hat{b} = (1-\eta, \eta)\cdot(a, b)$ are generated, where $\eta \in [0, 1]$ is a uniformly distributed random variable. The individuals $\hat{a}$ and $\hat{b}$ replace a and b respectively in the pool. Crossover is repeated with replacement $\lfloor N/2 \rfloor$ times.

3. *Random mutation.* An individual a in the mating pool is replaced by a randomly generated number from the domain $[0, vMAX]$. Mutation is applied to every individual in the pool once with a probability $pMUTATE$ per trial. No change is incurred to an individual if mutation is not applied.

Election is applied in the last and final stage of our algorithm during every time period to generate a population decision vector which is representive of individuals decisions during that time period.

4. *Election.* Real and binary coded election are identical operations.

Parameter values for the genetic algorithm are given in Appendix A.

We establish a step transition matrix for our Markov process following Ruldolph's [67] work. Ruldolph considered a state-independent genetic algorithm. His "canonical" genetic algorithm was constructed with three genetic operators, selection, crossover, and mutation. As an extension to his model, an elitist operator was also considered. This elitist operator independently stored the decision with highest fitness found to date. He showed that in such models a state transition matrix could be derived by analysing the transitions made by each individual genetic operator. Ruldolph represented the selection, crossover, and mutation operators by stochastic matrices, S, C, and M respectively. He showed that a state transition matrix P of his Markov chain could be derived by the multiplication of these three matrices in an order depending on the implementation of the genetic algorithm. Finally, he proved two things. First, when an elitist operation is not applied, a Markov chain modelling such a genetic algorithm has a unique non-zero stationary distribution. Second, when an elitist operation is applied, a global optimum is located by the genetic algorithm with probability one.

Ruldolph's Markov chain model for the genetic algorithm extends naturally to our state-dependent genetic algorithm. In our model, the selection, crossover, and mutation operations are still associated with stochastic matrices S, C, and M respectively. However, we can not immediately identify the product $\hat{P} = SCM$ of these three matrices with our state transition matrix $P = [p_{i,j}]$ since the influence of election is not accounted for.

Chapter 4

Equilibria and Stability in Economic Models

4.1 Cobweb Model

The formal analysis of economic models is an essential part of social and economic studies. Numerious models have been formulated to describe the behaviour observable in laboratory testing. These models can be divided into two categories.

Formulations of economic models which make the behavioural assumption of rational expectations are placed into the first category. In such a formulation agents are assumed to have identical beliefs about the economic system they inhabit and these beliefs are consistent with the actual observed dynamics of the system. As a consequence, rational agents are assumed to make predictions about the current and future state of the system with perfect foresight. Furthermore, rational agents are assumed to maximize expected utility.

Another formulation of economic models makes the behavioural assumption of adaptive expectations. In such a formulation agents are assumed to have identical

beliefs about the economic system they inhabit. However, these beliefs are ill-specified in that they are not consistent with the true observed dynamics of the system. Hence, adaptive agents do not possess perfect foresight. The assumption of utility maximization is often made although one might also propose a model in which agents do not behave in this way. We will consider only models specified under utility maximization at this point and relax this assumption to formulate our models using a genetic algorithm in the next chapter.

Let us consider a simple market of a single good which commonly occurs in economic literature, a cobweb model. We will formulate a cobweb model as given in Arifovic [3]. The economy is populated by N agents in competition producing a single consumer good. Each individual agent must decide a priori the amount $q_{i,t}$ of goods to be produced for sale during time period $t \geq 0$. Production costs are identical between agents and the cost function $c(q_{i,t})$ under consideration is that studied by Wellford [73] as given in Arifovic [4]

$$c(q_{i,t}) = xq_{i,t} + \frac{1}{2}yNq_{i,t}^2, \tag{4.1}$$

where $x \geq 0$, and $y > 0$ are constant. The price p_t is set by assuming a linear demand curve

$$p_t = A - B\sum_{i=1}^{N} q_{i,t}, \tag{4.2}$$

where $A > 0$ and $B > 0$ are constant. For price to be positive it is sufficient that

$$q_{i,t} < \frac{A}{NB}, \forall\ i = 1 \ldots N. \tag{4.3}$$

Profit is calculated as revenue less production cost

$$\Pi_{i,t} = p_t q_{i,t} - xq_{i,t} - \frac{1}{2}yNq_{i,t}^2. \tag{4.4}$$

We assume that all agents have identical expectations p_t^e for the price per unit good p_t. Hence, agents with price expectation p_t^e have an expectation for profit of

$$\Pi_{i,t}^e = p_t^e q_{i,t} - xq_{i,t} - \frac{1}{2}yNq_{i,t}^2. \tag{4.5}$$

Given p_t^e, each agent maximizes $\Pi_{i,t}^e$ by choosing a production of

$$q_{i,t} = \frac{1}{yN}\left(p_t^e - x\right), \tag{4.6}$$

for $p_t^e > x$. Hence, all agents make identical production decisions at time $t \geq 1$ irrespective of their respective production decision at time $t = 0$. For convenience, we assume that all agents make identical production decisions at time $t = 0$ so that $q_{i,t} = q_t$ for all $t \geq 0$ and the inequality (4.3) is a necessary and sufficient condition for price to be positive.

It is assumed for the cobweb model that

$$p_t^e = p_{t-1}, \ \forall\, t \geq 1. \tag{4.7}$$

Hence, from equation (4.2) and equation (4.6)

$$\frac{A - p_t}{BN} = \frac{1}{yN}\left(p_{t-1} - x\right). \tag{4.8}$$

Re-arranging this equation gives

$$p_t + \frac{B}{y}p_{t-1} = A + B\frac{x}{y}. \tag{4.9}$$

This equation has solution

$$p_t = (p_0 - p^*)\left(-\frac{B}{y}\right)^t + p^*, \tag{4.10}$$

where p^* is given by

$$p^* = \frac{Ay + Bx}{B + y}. \tag{4.11}$$

Under this parameterization, when

$$B < y, \tag{4.12}$$

the trajectories for price given by equation (4.10) are convergent to the stationary equilibrium p^*, with corresponding supply

$$q^* = \frac{A - x}{N(B + y)}. \tag{4.13}$$

For positive supply, A must be greater than x.

4.2 Overlapping Generations Model of Constant Money Supply

Let us consider another market of a single good, an overlapping generations model. In this model, agents live over two consecutive generations and optimize consumption over their lifetime.

The overlapping generations model considered here is Samuelson's [68] 1958 overlapping generations model of fiat money as given in Arifovic [3]. The economy is populated by agents living over two consecutive time periods, t and $t + 1$, $t \geq 0$. Generations of individual agents are of equal size N and each agent is said to be *young* in the first period of life and *old* in the second. A single perishable consumer good is introduced into the economy in the form of endowment. Individual agents are endowed with $w^{(1)}$ units of good in youth and $w^{(2)}$ units of good in old age. In the first period of life at time t agent i consumes an amount $c_{i,t}^{(1)} \in [0, w^{(1)}]$. The supply of excess goods possessed by an agent i in the first period of life at time t is given by

$$s_{i,t} = w^{(1)} - c_{i,t}^{(1)}. \tag{4.14}$$

Excess goods are supplied by agents only in youth. Nominal per capita money supply is assumed to be constant h each period, with total supply of fiat money Nh initially held by a generation of old agents at $t = 0$. Excess goods are sold on

the market at a price per good p_t determined by the total supply of goods and the total amount of fiat money in the economy

$$p_t = \frac{Nh}{\sum_i s_{i,t}}. \tag{4.15}$$

Individual young agents accumulate monetary savings through the sale of excess goods

$$m_{i,t} = p_t s_{i,t}, \tag{4.16}$$

and use these savings in the second period of life to purchase goods. In old age individual agents consume an amount of goods

$$c^{(2)}_{i,t+1} = w^{(2)} + \frac{m_{i,t}}{p_{t+1}}. \tag{4.17}$$

An individual's utility $U_{i,t}$ is a function of first and second period consumption at t and $t+1$ respectively

$$U_{i,t} = \mu(c^{(1)}_{i,t}, c^{(2)}_{i,t+1}). \tag{4.18}$$

The utility function $\mu(c^{(1)}_{i,t}, c^{(2)}_{i,t+1})$ is assumed to be concave, bounded above, increasing in $c^{(1)}_{i,t}$ and $c^{(2)}_{i,t+1}$, and first order differentiable.

Let r_t be the price ratio

$$r_t = \frac{p_t}{p_{t+1}}. \tag{4.19}$$

Individual agents do not know the prices p_t or p_{t+1} a priori. A model of behaviour is derived by assuming that individual agents formulate an expected value r^e_t for the ratio r_t with perfect foresight. This expectation is identical for all agents. Perfect foresight assumes that the expectation r^e_t and the observed price ratio $r_t = p_t/p_{t+1}$ coincide. Under perfect foresight individuals correctly predict the prices p_t and p_{t+1}. Individual agents $i = 1 \ldots N$ maximize utility over $[0, w^{(1)}]$ at each period $t \geq 0$ with r_t treated as a constant during maximization

$$\begin{array}{lrcl} \max_{0 \leq c^{(1)}_{i,t} \leq w^{(1)}} & U_{i,t} & = & \mu(c^{(1)}_{i,t}, c^{(2)}_{i,t+1}), \\ \text{where} & c^{(2)}_{i,t+1} & = & w^{(2)} + r_t(w^{(1)} - c^{(1)}_{i,t}). \end{array} \tag{4.20}$$

The concavity of the utility function combined with identically formed expectations for r_t implies that all individual agents act identically. It follows that $m_{i,t} = m_t$, $c_{i,t}^{(1)} = c_t^{(1)}$, and by equation (4.15) market price simplifies to

$$p_t = \frac{h}{w^{(1)} - c_t^{(1)}}, \tag{4.21}$$

and so

$$c_t^{(1)} = w^{(1)} - \frac{h}{p_t}, \tag{4.22}$$

for all $t \geq 0$.

Assuming that the initial first period consumption is given for all agents at $t = 0$, then the initial price p_0 can be calculated using equation (4.15). Using equation (4.19) and equation (4.22) we can see that the system (4.20) implicitly describes a difference equation for price

$$p_{t+1} = F(p_t), \ t \geq 0, \tag{4.23}$$

where the function $F : \mathbb{R}^+ \to \mathbb{R}^+$ relates the price at time t and $t + 1$.

The difference equation (4.23) together with p_0 gives a trajectory for price. We call a solution to this recursive system which is constant for all time *stationary*. The fixed point for price p^* corresponding to the solution trajectory $\{p^*\}_0^\infty$ is then called a *stationary rational expectations equilibrium* for price. We define stationary equilibria for first period consumption $c^{(1),*}$ similarly. By equation (4.22), $c^{(1),*} = w^{(1)} - h/p^*$.

4.3 Parameterization and the Utility Function

The utility function as analysed by Arifovic [3] in an overlapping generations model of constant money supply is a function of first and second period consumption for period t given by

$$\mu(c_t^{(1)}, c_{t+1}^{(2)}) = c_t^{(1)} c_{t+1}^{(2)}, \tag{4.24}$$

where

$$c_{t+1}^{(2)} = w^{(2)} + r_t(w^{(1)} - c_t^{(1)}). \tag{4.25}$$

Differentating equation (4.24) with respect to $c_t^{(1)}$, where r_t is treated as a constant, and equating the result to zero gives

$$c_t^{(1)} = \frac{w^{(2)}}{2}\frac{1}{r_t} + \frac{w^{(1)}}{2}. \tag{4.26}$$

Substituting for $c_t^{(1)}$ using equation (4.22) gives the first order linear non-homogeneous difference equation for price

$$p_{t+1} = \frac{w^{(1)}}{w^{(2)}} p_t - \frac{2}{w^{(2)}} h. \tag{4.27}$$

If $w^{(1)} = w^{(2)}$ this becomes

$$p_{t+1} - p_t = -\frac{2}{w^{(2)}} h, \tag{4.28}$$

with solution

$$p_t = p_0 - \frac{2}{w^{(2)}} ht. \tag{4.29}$$

If $w^{(1)} \neq w^{(2)}$ then the homogeneous form of equation (4.27) is

$$p_{t+1}^h - \frac{w^{(1)}}{w^{(2)}} p_t^h = 0, \tag{4.30}$$

with solution

$$p_t^h = \left(\frac{w^{(1)}}{w^{(2)}}\right)^t. \tag{4.31}$$

A particular solution p^* for all t is

$$p^* = \frac{2h}{w^{(1)} - w^{(2)}}. \tag{4.32}$$

The general solution to equation (4.27) when $w^{(1)} \neq w^{(2)}$ is then

$$p_t = A\left(\frac{w^{(1)}}{w^{(2)}}\right)^t + p^*, \tag{4.33}$$

where A is a constant.

Replacing A using the initial conditions, gives

$$p_t = (p_0 - p^*)\left(\frac{w^{(1)}}{w^{(2)}}\right)^t + p^*. \tag{4.34}$$

If we let $t \to \infty$ then, from equation (4.29)

$$p_t \to -\infty \quad \text{as} \quad t \to \infty \quad \text{if} \quad w^{(1)} = w^{(2)}. \tag{4.35}$$

Also, from equation (4.34)

$$\begin{aligned}
&p_t \to p^* && \text{as} \quad t \to \infty \quad \text{if} \quad w^{(1)} < w^{(2)}, \\
&p_t \to \infty && \text{as} \quad t \to \infty \quad \text{if} \quad w^{(1)} > w^{(2)} \quad \text{and} \quad p_0 > p^*, \\
&p_t \to -\infty && \text{as} \quad t \to \infty \quad \text{if} \quad w^{(1)} > w^{(2)} \quad \text{and} \quad p_0 < p^*, \\
&p_t = p^* && \forall t \quad \text{if} \quad w^{(1)} \neq w^{(2)} \quad \text{and} \quad p_0 = p^*.
\end{aligned} \tag{4.36}$$

A realistic interpretation of price requires that $p_t \geq 0$ for all $t \geq 0$. If $w^{(1)} < w^{(2)}$ then $p^* < 0$ which does not make physical sense. Thus, we require that $w^{(1)} > w^{(2)}$ and $p_0 \geq p^*$ for price to be non-negative for all $t \geq 0$.

We substitute our price equation (4.32) into equation (4.22) to obtain the stationary equilibrium for first period consumption. We then use equation (4.14), equation

(4.16), and equation (4.17) to obtain the corresponding second period consuption. That is,

$$c_t^{(1)} = c_t^{(2)} = c^{(1),*}, \ \forall \ t \geq 0, \tag{4.37}$$

are constant where

$$c^{(1),*} = \frac{w^{(1)} + w^{(2)}}{2}, \tag{4.38}$$

with utility

$$\mu(c^{(1),*}, c^{(1),*}) = \left(\frac{w^{(1)} + w^{(2)}}{2}\right)^2. \tag{4.39}$$

The stationary equilibrium for price $p_t = p^*$, for all $t \geq 0$, is unstable because the term $(w^{(1)}/w^{(2)})^t$ in equation (4.34) increases geometrically in time. This means that the stationary equilibrium p^* is attained only when the initial price p_0 happens to be p^* and $w_1 \neq w_2$. Otherwise it must be that $p_0 > p^*$ and $w_1 > w_2$ for $p_t \geq 0$, $\forall \ t \geq 0$. We can use equation (4.22) to see that as $p_t \to \infty$, $c_t^{(1)} \to c^{(1),*}$ and $c_t^{(2)} \to c^{(2),*}$, where

$$c^{(1),*} = w^{(1)}, \tag{4.40}$$

$$c^{(2),*} = w^{(2)}. \tag{4.41}$$

Utility tends to

$$\mu(w^{(1)}, w^{(2)}) = w^{(1)}w^{(2)}. \tag{4.42}$$

As $p_t \to \infty$ fiat money becomes valueless. As $c_t^{(1)} \to w^{(1)}$ from below, a decreasing amount of goods is left for sale at each period. Hence, a decreasing amount of goods are available for purchase in old age, irrespective of the amount of savings which may have been accrued in youth. This self sufficient policy of maximal consumption guarentees a utility of exactly $w^{(1)}w^{(2)}$. This is referred to, ipso facto, as an *autarkic state.*

4.4 Constant Real Deficit Financed Through Seignorage

A constant real deficit d is financed by some external body or government via seignorage. Seignorage is defined to be the revenue raised by money creation. That is, by printing fiat money, see Abel and Bernanke [1]. This deficit can be introduced into the overlapping generations model as given in Arifovic [3]. Per capita money supply becomes time dependent and

$$d = \frac{h_t - h_{t-1}}{p_t}. \tag{4.43}$$

The price function does not automatically assume all individuals act identically. Price at time t is determined by dividing the total value of goods sold at time $t-1$ by the difference in total supply and deficit at time t. It is given by

$$p_t = \frac{\sum_{i=1}^{N} s_{i,t-1} p_{t-1}}{\sum_{i=1}^{N} s_{i,t} - Nd}, \tag{4.44}$$

where

$$\sum_{i=1}^{N} s_{i,t} > Nd, \ \forall\, t \geq 0. \tag{4.45}$$

However, as a result of concavity and utility maximization using a utility function of the form (4.24), it turns out that individuals do act identically with $m_t = h_t$. A substitution of equation (4.19) into equation (4.26) gives

$$p_t c_t^{(1)} = \frac{w^{(1)}}{2} p_t + \frac{w^{(2)}}{2} p_{t+1}. \tag{4.46}$$

Using equation (4.14) and equation (4.16) we have that $m_t = p_t s_t = p_t(w^{(1)} - c_t^{(1)})$. Hence

$$p_t c_t^{(1)} = p_t w^{(1)} - m_t. \tag{4.47}$$

We use equation (4.47) to substitute for $p_t c_t$ in equation (4.46) with $m_t = h_t$. This gives

$$h_t = (p_t - \frac{w^{(2)}}{w^{(1)}} p_{t+1}) \frac{w^{(1)}}{2}. \tag{4.48}$$

Since $h_t > 0$, we require that

$$p_t - \frac{w^{(2)}}{w^{(1)}} p_{t+1} > 0. \tag{4.49}$$

That is

$$\frac{p_{t+1}}{p_t} < \frac{w^{(1)}}{w^{(2)}}. \tag{4.50}$$

Substitution of equation (4.48) into equation (4.43) gives

$$d = \frac{w^{(1)} + w^{(2)}}{2} - \frac{w^{(1)}}{2} \frac{p_{t-1}}{p_t} - \frac{w^{(2)}}{2} \frac{p_{t+1}}{p_t}. \tag{4.51}$$

Let

$$\pi_t = \frac{p_t}{p_{t-1}}, \tag{4.52}$$

be the inflation rate. Then

$$\pi_{t+1} = 1 + \frac{w^{(1)} - 2d}{w^{(2)}} - \frac{w^{(1)}}{w^{(2)} \pi_t}. \tag{4.53}$$

Equation (4.53) defines a non-linear recurrence relation describing the competitive rational expectations equilibrium. The stationary equilibria for inflation π^* are derived by solving the quadratic equation

$$(\pi^*)^2 - (1 + \frac{w^{(1)} - 2d}{w^{(2)}})\pi^* + \frac{w^{(1)}}{w^{(2)}} = 0, \tag{4.54}$$

where price is positive and finite, and $d > 0$.

Examining the discriminant of this quadratic, it is required that

$$d < \frac{w^{(1)} + w^{(2)}}{2} - \sqrt{w^{(1)} w^{(2)}}, \tag{4.55}$$

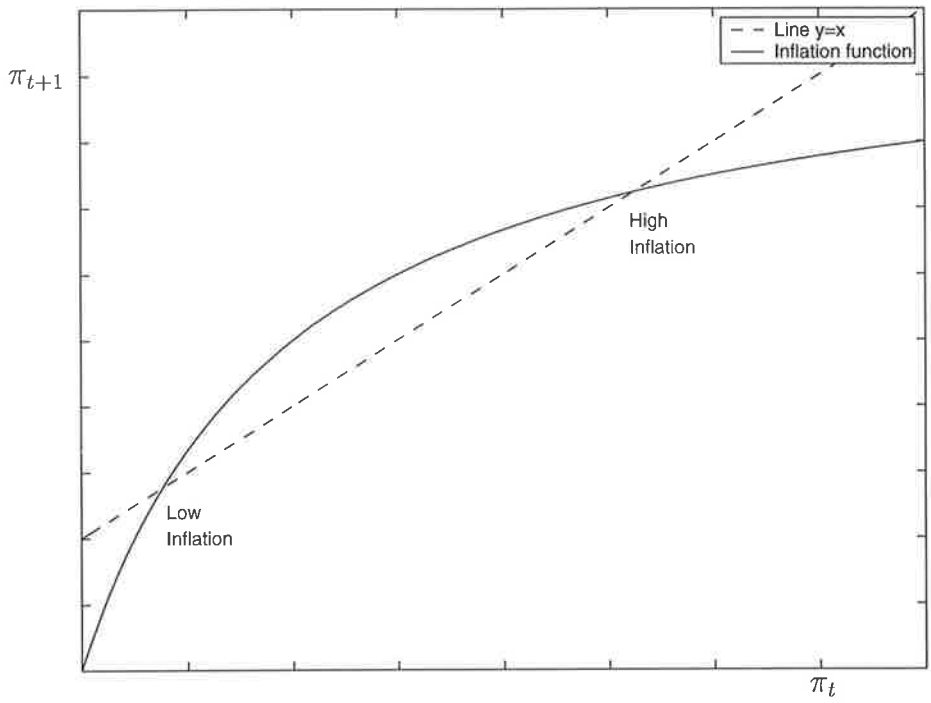


Figure 4.1: Stability diagram for inflation

to give two distinct real roots π_L^* and π_H^* as the respective low and high inflationary values

$$\frac{1}{2}\left(1+\frac{w^{(1)}-2d}{w^{(2)}} \pm \sqrt{(1+\frac{w^{(1)}-2d}{w^{(2)}})^2-4\frac{w^{(1)}}{w^{(2)}}}\right). \tag{4.56}$$

The stability is represented graphically in Figure (4.1) plotting π_{t+1} as a function of π_t against the 45° line. By observation of Figure (4.1) the inflation rate will approach the high inflationary equilibrium π_H^* if the initial inflation rate exceeds the low stationary equilibrium value, $\pi_0 \geq \pi_L^*$. Hence, the stationary equilibrium π_H^* is the stable attractor for all values of $\pi_0 > \pi_L^*$. The low inflation stationary point is obtained only when $\pi_0 = \pi_L^*$. If $\pi_0 < \pi_L^*$, then $\pi_t \to -\infty$ as $t \to \infty$. For inflation to be non-negative for all time it is necessary that $\pi_0 \geq \pi_L^*$. Also, using equation (4.50) and equation (4.52) we require that $\pi_t < w^{(1)}/w^{(2)}$ so that $h_t > 0$ for all $t \geq 0$. Hence, $\pi_0 \in [\pi_L^*, w^{(1)}/w^{(2)})$.

Chapter 5

Stochastic Representation of Economic Models

5.1 Stochastic Modelling and the Markov Process

In Chapter 4 we described the economic theory used to analyse our models. Implicit in our analysis is a formulation of behaviour in which utility maximization is assumed. Under this assumption we derived a dynamical system in which all individuals act identically. In Chapter 3 we described genetic algorithms and how they can be modelled as stochastic processes in discrete time. Implicit in this approach is formulation of behaviour in which individual agents use the genetic operators to generate decisions. As a consequence, agents do not act identically but adapt over time according to a complex evolutionary process. That is, genetic learning is a form of adaptive expectations learning which does not explicitly maximize utility.

A genetic algorithm might be applied to an economic model for several reasons.

First, minimal experimental data is required, if any, for simulations to be run. Second, analytic solutions for equilibrium or non-equilibrium dynamics are not necessary. In cases where experimental data might be costly or difficult to obtain, or where analytic solutions have not or cannot be derived, a genetic algorithm is a useful tool. Alternatively, genetic algorithms can be used to support experimental evidence in situations where rational expectations models are contradictory, or when it is uncertain which of many equilibria should be the outcome of a model over time. We will discuss an exact formulation of the genetic algorithm as applied to individual economic models in later sections.

While the number of applications of genetic algorithms in economics is extensive, see Arifovic [7], few publications attempt to explain the mathematics behind the genetic algorithm. The comprehensive work of Dawid [24] is a notable exception. The lack of detailed analysis of the genetic algorithm itself remains a problem. A number of general convergence results are derived in this chapter by considering a Markov chain formulation for the genetic algorithm. We then apply these results to the cobweb model given in Section 4.1.

The genetic operators presented in Section 3.3 are chosen from a large number of possible variations for each operator. However, what is important in our analysis is the communicating structure of the Markov chains that operators induce. Thus, to analyse a genetic algorithm we shall define conditions on the genetic operators rather than considering the individual operators directly. These conditions are defined below.

Condition 5.1. *There is a strictly non-zero probability that the decision q is mutated to $\tilde{q}$, $\forall\ q \in \Omega$, $\forall\ \tilde{q} \in \Omega$.*

Let us consider the sequence in which the genetic operators are applied. From Section 3.3, the ordering of operations places selection first, then crossover, mutation, and election. We notice that if Condition 5.1 is satisfied, each decision in

Ω is visited infinitely many times by mutation with probability one as $t \to \infty$. It follows that each population decision vector in $\mathcal{S}$ is visited infinitely many times by mutation with probability one as $t \to \infty$. Thus, every population decision vector in $\mathcal{S}$ is generated infinitely often with probability one by the sequence of operations; selection, crossover, and mutation. This does not guarantee that any particular population decision vector is elected. Hence, without loss of generality, we need now only to consider the communication between states during election to determine the communicating structure of the Markov chain.

A convexity condition is usually assumed. It is usual, but not necessary, for the fitness function to be continuously differentiable over the domain $\mathscr{D}$.

Condition 5.2. *The fitness function $f_q(\psi)$ is continuous and convex, and considered as a function of q, $f_q(\psi)$ attains a distinct unique maximum in $\mathscr{D}$ for each respective $\psi \in \mathcal{S}$.*

Let $\psi \in \mathcal{S}$ be some state and $q \in \mathscr{D}$ be some decision in the domain. As a consequence of the convexity and continuity condition given by Condition 5.2 there is some closed and bounded, convex, and non-empty subset of $\mathscr{D}$ which contains all decisions with fitness greater than or equal to q. We call this set of values the election space $\mathscr{E}_q(\psi)$ of the decision q.

Definition 5.1. *The election space $\mathscr{E}_q(\psi)$ of a decision $q \in \mathscr{D}$ conditional on the state $\psi \in \mathcal{S}$ is defined to be the subset of $\mathscr{D}$ consisting of all values $\tilde{q} \in \mathscr{D}$ such that $f_{\tilde{q}}(\psi) \geq f_q(\psi)$.*

Using Condition 5.2, the election space $\mathscr{E}_q(\psi)$, $q \in \Omega$, $\psi \in \mathcal{S}$, takes the form $[a, q]$ or $[q, b]$, for some $a, b \in \mathscr{D}$, and contains the unique maximum $\hat{q} = \arg\max_{\tilde{q} \in \mathscr{D}} f_{\tilde{q}}(\psi)$, irrespective of the value of q.

If an individual has the strictly highest possible fitness over Ω conditional on some state $\psi \in \mathcal{S}$ then the election operator will not allow that individual's decision to

change in the current period. The decision may not have strictly highest fitness in subsequent periods as the population state may change. This suggests that absorbing states of the genetic algorithm can be identified by analysing the election spaces of fittest individuals.

All coded decisions belong to the set Ω and are equally spaced a distance Δ apart. The intersection of an election space and Ω indicates which points are coded by the genetic algorithm in the election space. A population decision vector state (ψ_i), $\psi_i \in \Omega$, $i = 1 \dots N$ corresponding to the state $\psi \in \mathcal{S}$ is absorbing only if $|\mathscr{E}_{\psi_i}(\psi) \cap \Omega| = 1$, $\forall\ i = 1 \dots N$. That is, all individuals have an election space whose intersection with Ω contains only that coded decision which is identifiable with each respective individual.

Theorem 5.1. *Let ψ be a population decision vector of a genetic algorithm with gridspacing Δ. Then*
(a) ψ is uniform if $diam(\mathscr{E}_{\psi_i}(\psi)) < \Delta$, $\forall\ i = 1 \dots N$,
(b) ψ is absorbing iff $diam(\mathscr{E}_{\psi_i}(\psi)) < \Delta$, $\forall\ i = 1 \dots N$.

Proof: We first prove part (b) of out theorem. If a state is absorbing then ψ_i, $i = 1 \dots N$ may never change. That is, no individual may ever change their decision. Hence, every individual must have strictly highest possible fitness. Otherwise, by Condition 5.1, there is a strictly positive probability of mutating to and subsequently electing a different decision in Ω with equal or better fitness. An individual can only have strictly highest possible fitness if its election space contains no point other than itself. That is, for every $i = 1 \dots N$, $|\mathscr{E}_{\psi_i}(\psi) \cap \Omega| = 1$. By Definition 5.1, $\mathscr{E}_{\psi_i}(\psi)$ takes the form $[a, \psi_i]$ or $[\psi_i, b]$, for some $a, b \in \mathscr{D}$. Recall from Section 3.2 that coded decisions in Ω are equally spaced over $\mathscr{D}$ with a gridspacing of Δ. Hence, the condition $|\mathscr{E}_{\psi_i}(\psi) \cap \Omega| = 1$ can be re-stated as $\text{diam}(\mathscr{E}_{\psi_i}(\psi)) < \Delta$, $\forall$ $i = 1 \dots N$.

To show that the reverse also holds, let us consider the individual k. The election

space $\mathscr{E}_{\psi_k}(\psi)$ is defined to contain all decisions in $\mathscr{D}$ with equal or better fitness than ψ_k. Since $\Omega \subset \mathscr{D}$ this election space also contains all decisions in Ω with equal or better fitness than ψ_k. If $\text{diam}(\mathscr{E}_{\psi_k}(\psi)) < \Delta$ then $|\mathscr{E}_{\psi_i}(\psi) \cap \Omega| = 1$ and $\mathscr{E}_{\psi_i}(\psi)$ contains no element of Ω other than ψ_k. Hence, $f_{\psi_k}(\psi) > f_{\varphi}(\psi), \forall\, \varphi \in \Omega \backslash \{\psi_i\}$ and ψ_k has strictly highest fitness from amongst all decisions in Ω. Thus, no individual will ever change their decision and ψ is absorbing. This establishes part (b) of our theorem.

To get part (a) of our theorem, note that if $f_{\psi_k}(\psi) > f_{\varphi}(\psi), \forall\, \varphi \in \Omega \setminus \{\psi_i\}$, then $\psi_i = \psi_k, \forall\, i = 1 \ldots N$. That is, ψ is uniform. ◇

Corollary 5.1. *No non-uniform state is absorbing.*

If at any time a genetic algorithm enters an absorbing state u_k it remains in u_k ad infinitum. Theorem 5.1 gives a necessary and sufficient condition for a state to be absorbing. This does not guarantee that there will be an absorbing state or that that absorbing state will be entered. Hence, to analyse the convergence of our Markov chain we must consider both the existence and accessibility of absorbing states.

Theorem 5.2. *Consider a genetic algorithm with gridspacing Δ and let $P = [p_{i,j}]$ be its state transition matrix. Then the algorithm converges to the state $u_k \in \mathcal{S}$, $k \in \Omega$, with probability one iff $diam(\mathscr{E}_k(u_k)) < \Delta$ and for every $i \in \mathcal{S}$, $\exists\ m_i \in \mathbb{Z}^+$, such that $p_{i,u_k}^{(m_i)} > 0$.*

Proof: If $\text{diam}(\mathscr{E}_k(u_k)) < \Delta$, then by Theorem 5.1, u_k is an absorbing state. If for every $i \in \mathcal{S}$, $\exists\ m_i \in \mathbb{Z}^+$, such that $p_{i,u_k}^{(m_i)} > 0$, then u_k is accessible from every state i. Since $\mathcal{S}$ is finite, then by Theorem 2.4, the event $X(t) = u_k$ occurs with probability one in the limit $t \to \infty$. Hence, the genetic algorithm converges to the state u_k with probability one. Likewise, if the genetic algorithm converges to the state u_k with probability one it must be that $\text{diam}(\mathscr{E}_k(u_k)) < \Delta$ and for every

$i \in \mathcal{S}$, $\exists\ m_i \in \mathbb{Z}^+$, such that $p_{i,u_k}^{(m_i)} > 0$. $\diamond$

In general it is difficult to show that a state $u_k \in \mathcal{S}$, $k \in \Omega$, will be entered with probability one. From Theorem 5.2, we know that if a state is accessible from the entire state space in some finite number of steps then that state is entered with probability one. When this state is absorbing our Markov chain converges to that state with probability one. We will consider the accessibility of the state u_k from all initial states in exactly two steps to derive a sufficient, but not necessary, condition for convergence to the state u_k with probability one.

Theorem 5.3. *A Markov chain model for the genetic algorithm converges to the state $u_k \in \mathcal{S}$, $k \in \Omega$, with probability one if $diam(\mathscr{E}_k(u_k)) < \Delta$, and $k \in \mathscr{E}_j(u_j)$, $\forall$ $j \in \Omega$.*

Proof: Let $\hat{q}(\varphi) = \arg\max_{\tilde{q} \in \Omega} f_{\tilde{q}}(\varphi)$, $\varphi \in \mathcal{S}$. Conditions 5.1 and 5.2 imply that $\mathbb{P}(\psi(t) = u_{\hat{q}(\varphi)} | \psi(t-1) = \varphi) > 0$, $\forall\ t \geq 1$. If $\mathbb{P}(\psi(t+1) = u_k | \psi(t) = u_{\hat{q}(\varphi)}) > 0$ then $p_{\varphi,u_k}^{(2)} > 0$, $\forall\ \varphi \in \mathcal{S}$. This condition is met when $k \in \mathscr{E}_j(u_j)$, $\forall\ j \in \Omega$. Since $\text{diam}(\mathscr{E}_k(u_k)) < \Delta$, we know that u_k is absorbing by Theorem 5.1. Hence, the genetic algorithm converges to the state $u_k \in \mathcal{S}$, $k \in \Omega$, with probability one. $\diamond$

We can generalize Theorem 5.3 to the case where there is not necessarily an absorbing state but where there is a recurrent class that is accessible from all other states.

Theorem 5.4. *The recurrent class $\mathscr{C}$ is entered with probability one if $\exists\ u_k \in \mathscr{C}$ s.t. $k \in \mathscr{E}_j(u_j)$, for every $j \in \Omega$.*

Proof: The result follows from the proof of Theorem 5.3 mutatis mutandis. $\diamond$

5.2 Cobweb Model

The first application of a genetic algorithm in a cobweb model was given in Arifovic [3]. She demonstrated that genetic algorithms can be used to replicate the behaviour recorded in Wellford's [73] experiments and compared the genetic algorithm simulation results to the price expectation models of Ezekiel [36] who formulated the model using naive expectations, Marcet and Sargent [58] who formulated the model using least squares learning, and Carlson [20] who formulated the model using a sample average of past prices. Simulations showed that a genetic algorithm captured three features observed in Wellford's experimental data. First, the genetic algorithm converged to a stationary equilibrium solution. This convergence occured even when input parameters were consistent with unstable behaviour according to the analysis presented in Section 4.1. That is, the inequality (4.12) was not satisfied. Second, the genetic algorithm fluctuated about the equilibrium. Third, after thirty generations of the genetic algorithm, the price variance was greater across randomly selected seeds when the inequality (4.12) was not satisfied.

The early work of Arifovic provides a foundation for the exploration of the cobweb model using genetic algorithms. Dawid and Kopel [28] consider a formulation of a cobweb model with market exit and entry decisions. The impact of coding on simulation results is explained using an analysis of uniform states as given in Dawid [24], see also Vose and Liepins [72] and Nix and Vose [62]. While Arifovic concentrates on illustrating the workings of the genetic algorithm, Dawid formally establishes results for the genetic algorithm using stochastic theory for Markov chains.

In the cobweb model, the supply and demand curves affect how individuals respond to price. As derived in Chapter 4, the closed form solution for price given by

equation (4.10) was $p_t = (p_0 - p^*)(-B/y)^t + p^*$, where B appeared in the demand equation setting price, given by equation (4.2) and y appeared in the cost function, given by equation (4.1). The linear supply equation as a function of price was given by equation (4.6).

In Arifovic's simulations of the cobweb model using genetic algorithms, population decision vectors contain component values which represent the supply decisions $q_{i,t}$, $i = 1 \ldots N$, $t \geq 0$, corresponding to each member of a population at time t. The genetic operators generate these population decision vectors. The fitness function is identified with the utility function as is standard with simulations using genetic algorithms. In her application of the genetic algorithm, Arifovic uses the equation for profit given by equation (4.4) as utility. Agents are assumed not to know the price at time t a priori. Instead, a price forecast, as given by equation (4.7), uses the price which clears the market in the previous time period. Profit is calculated and the average supply determines the market price, as given by equation (4.2).

We apply a similar genetic algorithm as used by Arifovic. At each time period $t \geq 0$, our genetic algorithm encodes a population decision vector, representing N values for supply $q_{i,t} \in [0, A/(NB)]$. The initial values $q_{i,0}$, $i = 1 \ldots N$, are randomly generated. Supply decisions are decoded to determine the market price $p_0 = A - B\sum_{i=1}^{N} q_{i,0}$. We generate vectors of values for supply during successive periods $t > 0$ by applying the genetic operators. At $t = 1$, we apply the genetic selection operation to construct a mating pool consisting of N supply decisions. We apply genetic operators, crossover and mutation, to this mating pool as described in Section 3.3. We decode each new decision $\tilde{q}_{i,1}$ in the mating pool and calculate their fitness as $f(p_0, \tilde{q}_{i,1}) = p_0\tilde{q}_{i,1} - x\tilde{q}_{i,1} - \frac{1}{2}yN\tilde{q}_{i,1}^2$. During election, we compare the fitness of each individual i in the pool with the fitness value $f(p_0, q_{i,0})$. The value $f(p_0, q_{i,0})$ represents the actual profit individual i obtained at time $t = 0$. The value $f(p_0, \tilde{q}_{i,1})$ represents a prediction for utility at time t using the price forecast

$p_1^e = p_0$. If the fitness corresponding to individual i in the mating pool exceeds the profit obtained in the previous period then $q_{i,1} = \tilde{q}_{i,1}$, otherwise we retain the past supply decision and $q_{i,1} = q_{i,0}$. We use the values $q_{i,1}$ to construct our population decision vector at $t = 1$. We then calculate the total supply $\sum_{i=1}^{N} q_{i,1}$ and determine the corresponding price $p_1 = A - B\sum_{i=1}^{N} q_{i,1}$. In the consecutive periods $t = 2, 3, \ldots$, we continue this process, terminating after a set number of iterations.

	Set 1	Set 2	Set 3	Set 4	Set 5
A	2.184	2.296	2.296	2.296	2.296
B	0.0152	0.0168	0.0168	0.0168	0.0168
x	0	0	0	0	0
y	0.016	0.016	0.0084	0.0016	0.0008
N	30	30	30	30	30
B/y	0.95	1.05	2.0	10.5	21.0
q^*	2.3333	2.3333	2.9896	4.1594	4.3484
p^*	1.12	1.12	0.7653	0.1997	0.1044

Table 5.1: Cobweb model, parameter sets 1 to 5

Two parameter sets are used in Wellford's experiments [73] and adopted by Arifovic in her simulations. These two sets correspond to the first two parameter sets given in Table 5.1. In Arifovic's simulations a bit length of thirty is used and two hundred iterations conducted. Arifovic [3] reports values close to the stationary equilibrium price p^* are obtained by the genetic algorithm and she statistically analyses the variations in price. Simulations show that convergence to an equilibrium depends on both coding mechanism and parameter values. Problems associated with Hamming cliffs in binary coding are noted by Arifovic and she shows that Gray coding can be used to overcome these problems. Recall from Section 3.2 that the Hamming cliff problem is a consequence of the ordering of bits

within a string and a correspondence with the relative weight each bit contributes to the value of the string. Problems with these cliffs have previously been observed in studies involving genetic algorithms, see Caruana and Schaffner [21], Goldberg [44].

We conducted ten thousand iterations of the genetic algorithm under Gray coding with a bit length of ten. The parameter sets we examined are given in Table 5.1. Results for Sets 1, 2, 3, 4, and 5 are tabulated in Tables 5.2, 5.3, 5.4, 5.5, 5.6 respectively. These results represent the population decision vectors we observed upon termination of our simulations. Supply decisions in Ω made by agents are displayed as real numbers to four decimal places.

Simulations for the stable parameters given by Set 1 in Table 5.1 converged after less than one hundred iterations. Simulations for Sets 2 and 3 did not converge. However, at every iteration, a change in decision was recorded between population decision vectors whose components took one of the two values that appear in Table 5.3 and 5.4 respectively. No other values apart from these were observed after around one hundred iterations. Agents in simulations for Set 4 and Set 5 maintained a diverse range of values throughout all iterations.

To explain why such results are observed, let us consider the local asympotic stability conditions for the application of genetic algorithms derived by Dawid [26] pp. 83–91. Dawid analyses a dynamical system in discrete time with deterministic state calculations. This system represents the expected value of a Markov process modelling genetic algorithm dynamics at time $t+1$ conditional on the state of the process at time t. He remarks that the trajectories arising from this system

	Vector
Set 1	$u_{(2.3339)}$

Table 5.2: Cobweb model, population decision vector, set 1

Agents' decisions				
2.3313	2.3313	2.3313	2.3357	2.3313
2.3313	2.3313	2.3313	2.3313	2.3357
2.3357	2.3313	2.3357	2.3313	2.3357
2.3313	2.3313	2.3313	2.3357	2.3313
2.3313	2.3357	2.3357	2.3357	2.3313
2.3357	2.3313	2.3357	2.3313	2.3313

Table 5.3: Cobweb model, population decision vector, set 2

Agents' decisions				
3.0395	3.0351	3.0351	3.0351	3.0351
3.0351	3.0351	3.0395	3.0395	3.0351
3.0351	3.0395	3.0351	3.0395	3.0351
3.0351	3.0351	3.0395	3.0351	3.0351
3.0351	3.0351	3.0395	3.0395	3.0351
3.0395	3.0395	3.0351	3.0351	3.0351

Table 5.4: Cobweb model, population decision vector, set 3

Agents' decisions				
4.0293	4.2096	4.1260	4.2008	4.0909
4.1568	4.1260	4.1260	4.1876	4.2096
4.1700	4.1260	4.1656	4.1788	4.1656
4.1524	4.1524	4.1392	4.1129	4.2052
4.1788	4.1129	4.2052	4.1788	3.9633
4.1480	4.1964	4.1876	4.0733	4.1524

Table 5.5: Cobweb model, population decision vector, set 4

Agents' decisions				
4.1173	4.3900	4.3020	4.3724	4.3592
4.3240	3.5322	4.3768	4.3856	4.3240
4.1964	4.4120	4.2756	4.2844	4.4076
4.3944	4.1876	4.3152	4.3548	4.3812
4.2932	4.3592	4.3416	4.3636	4.3765
4.3240	4.3768	4.3548	4.3328	4.2932

Table 5.6: Cobweb model, population decision vector, set 5

approximate the behaviour of a genetic algorithm when the population size is sufficiently large and mutation is applied with sufficiently small probability. A binary encoded population $u_k \in \mathcal{S}$, $k \in \Omega$, with a mutation probability of zero and one-point crossover with probability $\chi \in (0, 1]$ is a locally asympotically stable state of this system if

$$\frac{\lambda(j,k)}{l-1} > \frac{1}{\chi}\left(1 - \frac{f_k(u_k)}{f_j(u_k)}\right), \tag{5.1}$$

for all binary coded strings $j \in \Omega$, $j \neq k$, see Dawid [26] p. 87. The value $l > 0$ in the inequality (5.1) represents the number of bits used in encoding decisions and $\lambda(j,k)$ represents distance between the two outmost bits of j and k which differ in value and zero when j and k differ in value at less than two distinct bit positions. Likewise, a binary encoded population $u_k \in \mathcal{S}$, $k \in \Omega$, with a mutation probability of zero and one-point crossover with probability $\chi \in (0, 1]$ which is locally asymptotically stable satisfies the inequality (5.1).

It is noted by Dawid [26] pp. 161–165 that states other than stationary equilibria may be locally asymptotically stable. Let us consider exempli gratia Set 1 of Table 5.1 with a bit length of $l = 10$ as used in Dawid's work. There are two distinct binary strings which are locally asympotically stable. The first represents the stationary equilibrium q^*, the second a value to four decimals of 2.3523 which is encoded as the string $k = (1000000000)$. To display these results graphically, Dawid takes the value of χ in the inequality (5.1) to be one and rearranges this inequality to obtain

$$\frac{l-1}{l-1-\lambda(j,k)} > \frac{f_j(u_k)}{f_k(u_k)}, \tag{5.2}$$

To determine if the inequality (5.2) holds for our specified value of k, one need only to check that the points $(\lambda(k,j), f_j(u_k)/f_k(u_k))$, $j \in \Omega \setminus \{k\}$, lie below the curve $g(\lambda) = (l-1)/(l-1-\lambda)$.

For $k = (1000000000)$, Figure 5.1 plots the function $g(\lambda)$ as a solid curve and

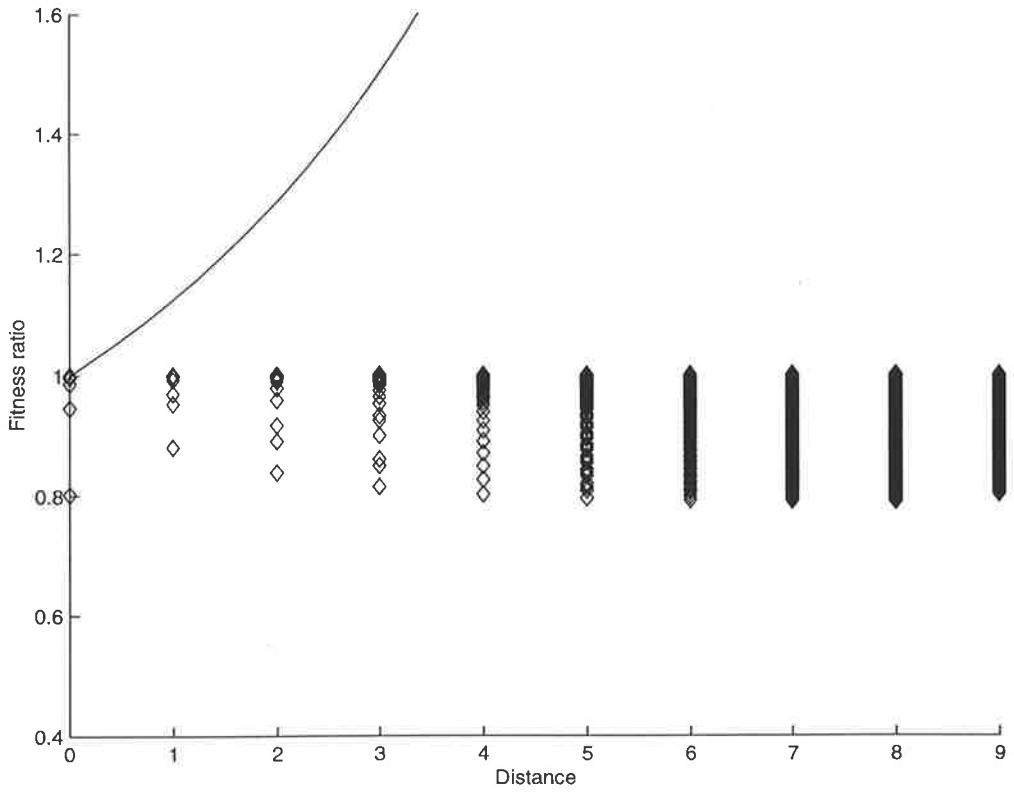


Figure 5.1: Local asymptotic stability diagram

points $(\lambda(k,s), f_j(u_k)/f_k(u_k))$, $j \in \Omega \setminus \{k\}$, as diamonds. As each diamond lies below $g(\lambda)$ the uniform state u_k is locally asymptotically stable even when the crossover probability is taken to be one.

We can use the inequality (5.1) to deduce properties of a genetic algorithm with Gray coding, see also Dawid [26] p. 90. Observe that if there exists a coded decision $j \in \Omega$ differing from another coded decision $k \in \Omega$ in a single bit position, then $\lambda(j,k) = 0$ and the left hand size of the inequality (5.1) becomes zero. Moreover, if $f_j(u_k) > f_k(u_k)$ then the right hand side of the inequality (5.1) is strictly positive. Thus, the inequality (5.1) is not satisfied and so u_k cannot be locally asymptotically stable.

Now, recall from Section 3.2 that a Gray code $k \in \Omega \setminus \{0\ldots0, 1\ldots1\}$ is constructed

such that if k has neighbours $i, j \in \Omega$ corresponding to the gridpoints immediately to the left and right of k respectively, then $\lambda(i,k) = \lambda(j,k) = 0$. This implies that in models with a suitable fitness fuctions, continuous and concave over the domain, no uniform state other than those corresponding to stationary equilibria may be locally asympotically stable if Gray coding is used.

The analysis of locally asymptotic stable states given by Dawid does not explain certain key observations about the model. As we have shown, the string (1000000000) is locally asymptotically stable according to the definition of Dawid. However, this string did not arise as a convergent point of the genetic algorithm in Set 1 of our data. In addition, the criterion for local asymptotic stability in uniform states does not explain the persistence of a system of two distinct supply decisions observed in Sets 2 and Set 3 of our data. To explain such findings an alternative approach must be found. Finally, local asymptotic stability does not help us to understand how in Set 4 and Set 5 of our data supply decisions deviate from the equilibrium to such an extent. That is, asymptotic convergence conditions tell us little about the behaviour of the model if no uniform state is stable. To answer these questions we adopt an approach based on Markov chains to analyse the properties of election spaces in our model.

The gridspacing corresponding to a l bit binary encoding over $[0, M]$ is $\Delta = M/(2^l - 1)$, where M represents the maximum value encoded by the genetic algorithm. The exact value q^* is approximated to some desired precision by specifying l. Denote by $q_- \leq q^*$ and $q_+ \geq q^*$ the two closest points to q^* as encoded by the genetic algorithm on either side of q^*, possibly the point q^* itself. Assume that the average supply at time t is less than q^*. Thus, we can write

$$\frac{1}{N}\sum_{i=1}^{N} q_{i,t} = q^* - \delta_t, \tag{5.3}$$

for some $\delta_t > 0$. Hence,

$$\delta_t = q^* - \frac{1}{N}\sum_{i=1}^{N} q_{i,t}, \tag{5.4}$$

represents the difference between the equilibrium supply q^* given by equation (4.13) and the average supply at each time period. By equation (4.2), the corresponding price at time t is given by

$$p_t = A - BN(q^* - \delta_t). \tag{5.5}$$

Let $\psi(t) \in \mathcal{S}$ be our population decision vector at time $t \geq 0$. From the set of all possible supply decisions Ω at time $t+1$, we use equation (4.4) to calculate the election space $\mathscr{E}_{q^*-\delta_t}(\psi(t))$ representing those values with equal or higher fitness to the supply strategy $q^* - \delta_t$. Our election space $\mathscr{E}_{q^*-\delta_t}(\psi(t))$ is convex and non-empty, taking the form $[a, b]$, $a, b \in \mathscr{D}$. The endpoints a and b are the two real roots of the quadratic equation in ϕ

$$p\phi - x\phi - \frac{1}{2}yN\phi^2 = p(q^* - \delta_t) - x(q^* - \delta_t) - \frac{1}{2}yN(q^* - \delta_t)^2. \tag{5.6}$$

This has solution $\phi = q^* - \delta_t$ and

$$\begin{aligned} \phi &= 2\frac{p-x}{yN} - (q^* - \delta_t), \\ &= 2\frac{A-x}{yN} - (q^* - \delta_t)(1 + 2\frac{B}{y}). \end{aligned} \tag{5.7}$$

From Section 4.1 we deduce that

$$\frac{A-x}{yN} - (q^* - \delta_t)(1 + 2\frac{B}{y}) \geq q^* - \delta_t, \tag{5.8}$$

so that

$$\mathscr{E}_{q^*-\delta_t}(\psi(t)) = [q^* - \delta_t, 2\frac{A-x}{yN} - (q^* - \delta_t)(1 + 2\frac{B}{y})]. \tag{5.9}$$

When $\delta_t = 0$, $\text{diam}(\mathscr{E}_{q^*-\delta_t}(\psi(t))) = 0$. An analysis of the the first derivative of $\text{diam}(\mathscr{E}_{q^*-\delta_t}(\psi(t)))$ with respect to δ_t reveals that as $\delta_t \to 0^+$ the subset (5.9) decreases in diameter to zero.

Assume that $\delta_t = \delta$ is a constant up to time t and that $q_{i,t} = q^* - \delta$, $\forall\, i = 1 \ldots N$. Hence, $\psi(t) = \psi = u_{q^*-\delta}$. Any point in $\mathscr{E}_{q^*-\delta}(u_{q^*-\delta})$ encoded by the genetic algorithm may be elected at $t+1$ and the number of points encoded by the genetic algorithm within $\mathscr{E}_{q^*-\delta}(u_{q^*-\delta})$ can be determined. For there to be no more than one point encoded by the genetic algorithm in $\mathscr{E}_{q^*-\delta}(u_{q^*-\delta})$, $\text{diam}(\mathscr{E}_{q^*-\delta}(u_{q^*-\delta})) < \Delta$. That is, $u_{q^*-\delta}$ is an absorbing state according to Theorem 5.1. This is equivalent to

$$2\frac{A-x}{yN} - (q^* - \delta)(1 + 2\frac{B}{y}) - (q^* - \delta) < \Delta. \tag{5.10}$$

This simplifies to

$$\frac{A-x}{yN} - (q^* - \delta)(1 + \frac{B}{y}) < \frac{\Delta}{2}. \tag{5.11}$$

Given Δ, we rearrange this inequality to give a bound on δ of

$$\delta < q^* + (\frac{\Delta}{2} - \frac{A-x}{yN}) \left(1 + \frac{B}{y}\right)^{-1}. \tag{5.12}$$

Substitution for q^* using equation (4.13) gives

$$\delta < \frac{\Delta}{2}\frac{y}{B+y}. \tag{5.13}$$

Analogous situations arise when the average supply given by equation (5.3) is assumed to be $q^* + \delta_t$, $\delta_t > 0$.

We give an example which displays the election space $\mathscr{E}_{q_-}(u_{q_-})$ in Figure 5.2. In this example the inequality (5.13) is satisfied. Hence, the state u_{q_-} is absorbing.

Theorem 5.5. *Let q^* be the stationary equilibrium supply for the cobweb model given by equation (4.13). If there is a supply decision $q \in \Omega$ in a genetic algorithm*

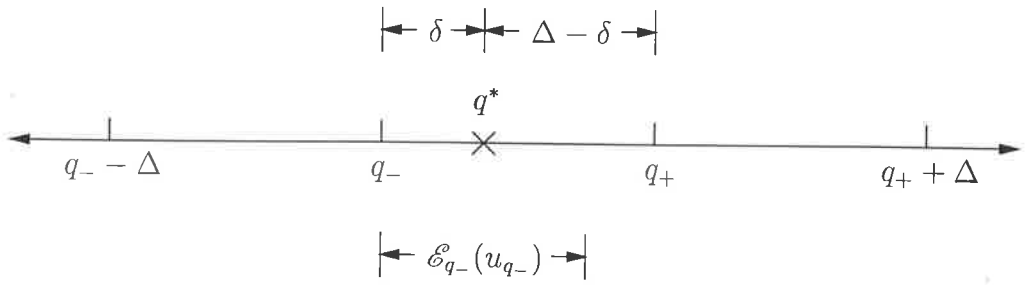


Figure 5.2: Gridspacing and set 1

simulation of the cobweb model with gridspacing Δ *such that*

$$|q^* - q| < \frac{\Delta}{2}\frac{y}{B+y}, \tag{5.14}$$

then the genetic algorithm is convergent to the uniform state u_q *with probability one. Conversely, if the genetic algorithm is convergent to the uniform state* u_q, $q \in \Omega$, *with probability one, then the inequality (5.14) is satisfied.*

Proof: To use Theorem 5.3 two requirements must be met. The first requirement is that $\text{diam}(\mathscr{E}_q(u_q)) < \Delta$, the second that $u_q \in \mathscr{E}_j(u_j), \forall\, j \in \Omega$. From the inequalities (5.10)–(5.13), we know that $\text{diam}(\mathscr{E}_q(u_q)) < \Delta$, if the inequality (5.14) is satisfied.

Let us consider the second requirement. By Condition 5.2 the election space $\mathscr{E}_k(u_k)$ of the population decision vector $u_k \in \mathcal{S}$, $k \in \Omega$, contains the value $\hat{q} = \arg\max_{\tilde{q}\in\mathscr{D}} f_{\tilde{q}}(u_k)$. Using our analysis of equation (5.9), $\text{diam}(\mathscr{E}_k(u_k))$ is a strictly decreasing function as k approaches q^*. Notice that utility, hence fitness, is a quadratic function of supply with a negative coefficient of the squared term. By definition, an election space conditional on the decision k and the state u_k contains all values for supply in $\mathscr{D}$ of equal or better fitness than k. Two possibilities arise. First, both endpoints of $\mathscr{E}_k(u_k))$ have the same fitness. Second, the endpoints of $\mathscr{E}_k(u_k))$ have different fitness. In the first case, notice that since fitness is a quadratic function of supply, $\hat{q}$ occurs at the midpoint of $\mathscr{E}_k(u_k)$ so that $\text{diam}(\mathscr{E}_k(u_k)) = 2|\hat{q} - k|$. We deduce from Section 4.1 that if $k < q^*$ then $\hat{q} > q^*$ and if $k > q^*$ then $\hat{q} < q^*$. This means that $2|\hat{q} - k| \geq 2|q^* - k|$ and

$\text{diam}(\mathscr{E}_k(u_k)) \geq 2|q^* - k|$. Hence, the point q occurs in every election space $\mathscr{E}_k(u_k)$ for which $|q^* - k| \geq |q^* - q|$.

In the second case, fitness is still a quadratic function of supply. However, $\mathscr{E}_k(u_k)$ is confined within $\mathscr{D}$ causing the two endpoints of $\mathscr{E}_k(u_k)$ to have different fitness. Trivially, $\mathscr{E}_k(u_k)$ contains every decision q where $|q^* - k| \geq |q^* - q|$. Now, $y < B + y$, as both B and y are non-negative. Hence, if $|q^* - q| \geq \Delta/2$ then the inequality (5.14) can never be satisfied. Hence, q is the unique point in Ω strictly closest to q^*. This point occurs in every election space $\mathscr{E}_k(u_k)$, $k \in \Omega$ because every $k \in \Omega \setminus \{q\}$ is distanced further from q^* then q. Theorem 5.3 now completes the first part of our proof.

It also follows that if the genetic algorithm converges to the state u_q with probability one, then u_q must be an absorbing state. If the state u_q is absorbing, then $\text{diam}(\mathscr{E}_q(u_q)) < \Delta$). Hence, the inequality (5.14) must be satisfied. This completes our proof. ◇

Note that the parameters for Set 1, in which convergence to a single point is observed, satisfy the inequality (5.13).

We have derived a test for a genetic algorithm to converge to the state $q \in \Omega$ with probability one. However, convergence to the state q with probability one is not the only type of behaviour we have observed in our simulations.

To explain the second and third sets of simulation results from Table 5.3 and Table 5.4 respectively, we again consider our election space given by equation (5.9). We again assume that $\delta_t = \delta$ is a constant up to time t and that $q_{i,t} = q^* - \delta$, $\forall$ $i = 1 \ldots N$. Hence, $\psi(t) = \psi = u_{q^*-\delta}$. Suppose we relax the constraint (5.10) and instead look at the election space when the right hand side is 2Δ

$$2\frac{A-x}{yN} - (q^* - \delta)(1 + 2\frac{B}{y}) - (q^* - \delta) < 2\Delta, \tag{5.15}$$

then there are at most two distinct points encoded by the genetic algorithm in

$\mathscr{E}_{q^*-\delta}(u_{q^*-\delta})$.

The inequality (5.15) simplifies to

$$\delta < \Delta \frac{y}{B+y}, \tag{5.16}$$

Again, analogous situations arise when the average supply given by equation (5.3) is assumed to be $q^* + \delta_t$, $\delta_t > 0$.

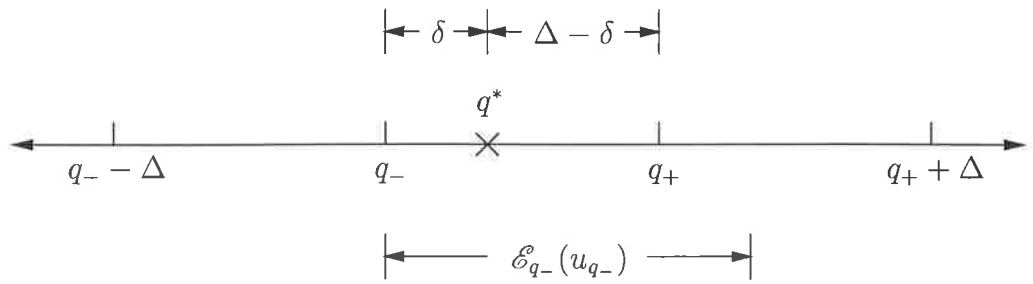


Figure 5.3: Gridspacing and set 2

We give an example which displays the election space $\mathscr{E}_{q_-}(u_{q_-})$ in Figure 5.3. In this example the state u_{q_-} is not absorbing. However, the inequality (5.16) is satisfied and $\mathscr{E}_{q_-}(u_{q_-})$ contains only the two coded decisions q_- and q_+.

It may be that the inequality (5.16) is satisfied by the states u_{q_-} and u_{q_+}. Let us consider the set $\Lambda \subseteq \mathcal{S}$ composed of all of population decision vectors $\psi = (\psi_i)$, $\psi_i \in \{q_-, q_+\}$, $i = 1 \ldots N$. Now, the value $\max_{\psi\in\Lambda, q\in\psi} \operatorname{diam}(\mathscr{E}_q(\psi))$ occurs either at $\psi = u_{q_-}$, $q = q_-$ or at $\psi = u_{q_+}$, $q = q_+$. To verify that $\mathscr{E}_q(\psi)$, contains no more than two distinct points in Ω for all $\psi \in \Lambda$, $q \in \{q_-, q_+\}$, it is sufficient that inequality (5.16) be satisfied for u_{q_-} and u_{q_+}.

Theorem 5.6. *Let q^* be the stationary equilibrium supply for the cobweb model given by equation (4.13). If there is a supply decision $q \in \Omega$ in a genetic algorithm simulation of the cobweb model with gridspacing Δ such that*

$$|q^* - q| < \Delta \frac{y}{B+y}, \tag{5.17}$$

$\forall\ q \in \{q_-, q_+\}$, *and* $\nexists\ \varphi \in \mathcal{S}$ *such that* φ *is absorbing, then the set* $\Lambda = \{(k_i)|k_i \in \{q_-, q_+\}, i = 1 \ldots N\}$ *forms a positive recurrent class. Conversely, if the set* Λ *forms a positive recurrent class, then the inequality (5.17) is satisfied,* $\forall\ q \in \{q_-, q_+\}$, *and* $\nexists\ \varphi \in \mathcal{S}$ *such that* φ *is absorbing.*

Proof: The inequality (5.17) follows from inequality (5.16). If no globally asymptotically stable state exists, and if q_- and q_+ are such that the inequality (5.17) is satified, then $\mathscr{E}_{q_-}(\varphi) \cap \Omega$ and $\mathscr{E}_{q_+}(\varphi) \cap \Omega$ contain exactly the two points q_- and q_+. That is, Λ is a recurrent class because all states in Λ communicate but no state outside Λ is accessible. Following an argument similar to that used in the proof of Theorem 5.5, both of q_- or q_+ occur in every election space $\mathscr{E}_k(u_k)$, $k \in \Omega$ and we can use Theorem 5.4 to show that Λ is entered with probability one. This completes the first part of our proof. If Λ forms a positive recurrent class then it must be that the inequality (5.17) is satified, $\forall\ q \in \{q_-, q_+\}$, or a state outside Λ is accessible. Also, no absorbing state may exist so that all states in Λ communicate. This completes our proof. $\diamond$

In Set 2 of the input parameters given in Table 5.1, $q_- = 2.3317$, $q^* = 2.3333$, and $q_+ = 2.3357$ with a gridspacing of approximately 0.0044. The election space $\mathscr{E}_{(2.3317)}(u_{(2.3317)})$ is approximately $[2.3317, 2.3382]$. There are two points encoded by the genetic algorithm within this region, namely q_- and q_+. A similar result holds for $\mathscr{E}_{(2.3357)}(u_{(2.3357)})$.

If the diameter of an election space is sufficiently large, more than two distinct points from Ω lie within the election space. When $\text{diam}(\mathscr{E}_q(u_q)) \geq \Delta y/(B + y)$ for at least one of $q \in \{q_-, q_+\}$, we might expect that all states in $\mathcal{S}$ are recurrent. However, the domain $\mathscr{D} = [0, M]$ is bounded and these boundaries cause some of the states to be transient.

Theorem 5.7. *The set* $\mathscr{C} = \{(k_i)|k_i \in \mathscr{E}_0(u_0) \cap \mathscr{E}_M(u_M) \cap \Omega, i = 1 \ldots N\}$ *forms a positive recurrent class iff the inequality (5.17) is not satisfied for at least one of*

the coded decisions $\{q_-, q_+\}$ *and* $\nexists \varphi \in \mathcal{S}$ *such that* φ *is absorbing.*

Proof: Recall from section 3.2 that $\mathscr{D} = [0, M]$ so that M denotes the largest value in Ω. Now, any coded decision $r \in \mathscr{E}_q(u_q)$ can be reached by each individual in the genetic algorithm. Hence, any state with components from $\mathscr{E}_q(u_q)$ is accessible from u_q.

Recall our comments on the set (5.9). For $q \in \mathscr{D}$, the diameter of $\mathscr{E}_q(u_q) \to 0$ linearly as $q \to q^*$. Now, $\mathscr{E}_0(u_0)$ denotes the election space with largest possible diameter from those coded decisions $q < q^*$. Likewise, $\mathscr{E}_M(u_M)$ denotes the election space with largest possible diameter from those coded decisions $q > q^*$. These election spaces overlap. Hence, the state of the Markov chain has a positive probability of leaving $\mathcal{S} \setminus \mathscr{C}$ never to return and all states $\mathcal{S} \setminus \mathscr{C}$ are transient.

We now show that all states in $\mathscr{C}$ are accessible. From this accessibility we can deduce that all states in $\mathscr{C}$ communicate. Let m denote the slope of the straight line segment $\text{diam}(\mathscr{E}_q(u_q))$, $q > q^*$ so that $-m$ denotes the slope of the straight line segment $\text{diam}(\mathscr{E}_q(u_q))$, $q < q^*$. As the inequality (5.17) is not satisfied for at least one of u_{q_-}, u_{q_+}, at least one of u_{q_-}, u_{q_+} has an election space of diameter at least 2Δ. However, both q_- and q_+ are at a distance strictly less than Δ from q^*. That is, $m > 2$. Hence, if the state of the Markov chain were to move from q_- to $q_+ + \Delta$ or from q_+ to $q_- - \Delta$ the diameter of the election space will increase from at least 2Δ to at least $2\Delta + m\Delta$. To reach a coded decision $r \in \Omega$ outside $\mathscr{E}_q(u_q)$ in the next time period, let the state of the Markov chain move to a uniform state u_r, $r \in \mathscr{E}_q(u_q)$, such that $|q^* - r| > |q^* - q|$. Such a state u_r must exist and $\text{diam}(\mathscr{E}_r(u_r))$ will exceed $\text{diam}(\mathscr{E}_q(u_q))$ by at least $m\Delta$. Repeating this process, every uniform state in $\mathscr{C}$ may be reached. From these uniform states it is possible to reach any other state in $\mathscr{C}$. No state in $\mathscr{C}$ communicates with a state in $\mathcal{S} \setminus \mathscr{C}$. Hence, the set of states $\mathscr{C}$ is recurrent.

It follows from Theorem 5.5 and Theorem 5.6 that if $\mathscr{C}$ forms a positive recurrent

class then the inequality (5.17) may not satisfied for at least one of the coded decisions $\{q_-, q_+\}$. ◇

In the simulation results in Table 5.5 and Table 5.6 for Sets 4 and 5 respectively, a broad range of values is recorded. Hence, we do not expect the inequality (5.16) to be satisfied. For Set 4 of our input parameters from Table 5.1, $q_- = 4.1568$, $q^* = 4.1594$, and $q_+ = 4.1612$ with a gridspacing of approximately 0.0044. The election space $\mathscr{E}_{(4.1568)}(u_{(4.1568)})$ is approximately $[4.1568, 4.2170]$. There are fourteen points encoded by the genetic algorithm within this region. Similar results are obtained when one considered the election space $\mathscr{E}_{(4.1612)}(u_{(4.1612)})$. Populations retain a diverse range of values ad infinitum. Analogous results apply for Set 5 of our input parameters.

Theorem 5.8. *Let $\mathscr{C}$ be a recurrent class in a genetic algorithm simulation of the cobweb model with gridspacing Δ. Either*

(a) $\mathscr{C}$ is a single absorbing state, either q_- or q_+, or

(b) $\mathscr{C}$ is composed of all states $\{(k_i)|k_i \in \{q_-, q_+\}, i = 1 \ldots N\}$, or

(c) $\mathscr{C}$ is composed of all states $\{(k_i)|k_i \in \mathscr{E}_0(u_0) \cap \mathscr{E}_M(u_M) \cap \Omega, i = 1 \ldots N\}$.

Let us assume that q_+ lies closer to q^* than q_-. In Theorem 5.8 above, for $\mathscr{C}$ to be composed of all states $\{(k_i)|k_i \in \mathscr{E}_0(u_0) \cap \mathscr{E}_M(u_M) \cap \Omega, i = 1 \ldots N\}$ two cases must apply. First, the inequality (5.14) is violated for q_+ and $q_+ - q_- \geq \Delta y/(2B + 2y)$. Second, the inequality (5.17) is violated at least for q_- and $q_+ - q_- \geq \Delta y/(B + y)$. Hence,

$$\Delta = q_+ - q_- \geq \frac{3\Delta}{2} \frac{y}{B + y}. \tag{5.18}$$

This simplifies to

$$B \geq \frac{y}{2}. \tag{5.19}$$

In the cobweb model described in Section 4.1, convergence to q^* is obtained for stable parameter values $B < y$. This means that case (c) of Theorem 5.8 might

occur even when q^* is locally asymptotically stable with respect to these naive expectations cobweb dynamics. Hence, to ensure that case (c) of Theorem 5.8 does not occur, $B < y/2$.*

In simulations using the genetic algorithm, convergence to the single decision closest to q^* can occur only when the inequality (5.13) is satisfied. The gridspacing Δ decreases as the number of bits used in encoding solutions increases. Let δ_{min} as the minimum of $\{q^* - q_-, q_+ - q^*\}$. The value δ_{min} lies in the range $(0, \Delta/2]$ so that the largest possible value of δ_{min} decreases as Δ decreases. The ratio of the largest value of δ_{min} to Δ is constant. Recall that the inequality (5.13) exhibits a dependence on both Δ and $\delta \geq \delta_{min}$. This means that if a simulation does not converge under some particular bit length, it cannot a priori be forced to converge by arbitrarily increasing the accuracy of the coding. Furthermore, a simulation which is convergent using one particular bit length may not be convergent with another. Either the inequality (5.13) is satisfied in a new parameterization or not. The probability that any particular bit length chosen at random satisfies the inequality (5.13) is a constant for any set of parameters in the economic model.

To illustrate this result, we simulate the stable parameters given by Set 1 with a bit length of thirteen rather than ten. Results of this simulation are given in Table 5.7. It can be verified that the condition for convergence given by the inequality (5.13) is not satisfied. The inequality (5.16) is satisfied and the two values q_- and q_+ are observed.

The number of decisions encoded by a genetic algorithm within any election space determines the maximum number of different decisions which may be adopted by individual agents. Arifovic conducted a statistical test using twenty random seed values and thirty generations of a genetic algorithm. She concluded that a cobweb model containing an unstable stationary equilibrium has a greater price

*I am indebted to one of the examiners for this observation.

Agents' decisions				
2.3336	2.3336	2.3330	2.3336	2.3330
2.3336	2.3336	2.3336	2.3330	2.3336
2.3330	2.3330	2.3330	2.3336	2.3330
2.3330	2.3330	2.3336	2.3330	2.3330
2.3330	2.3330	2.3330	2.3336	2.3330
2.3336	2.3336	2.3330	2.3336	2.3336

Table 5.7: Cobweb model, second simulation of set 1

variance than a model containing a stable stationary equilibrium. This is consistent with the analysis of the election space and the inequality (5.13). An increased relative likelihood that multiple points in election spaces are encoded by the genetic algorithm causes a greater volatility in decisions and a greater price variance in simulations.

Let us consider Arifovic's [6] notion of a stable stationary equilibrium under a genetic-algorithm adaptation. For some $k \in \Omega$, consider the uniform state u_k. Let the genetic algorithm be in state u_k at some time t and replace the j^{th} component $(u_k)_j$ by $\varphi_j \neq k$, $\varphi_j \in \Omega$. Arifovic defines the state u_k as stable under a genetic-algorithm adaptation if the genetic algorithm returns to u_k. We now contrast this approach with our stability analysis and that of Dawid. We have shown that the inequality (5.13) is a necessary and sufficient condition for convergence. By our remarks on Dawid's local asymptotic stability condition, if Gray coding is used the inequality (5.1) and the inequality (5.13) are either both satisfied or neither. Thus, under Gray coding a uniform state is locally asympotically stable according to Dawid when it is stable under a genetic-algorithm adaptation. However, if Gray coding is used, the inequality (5.1) can be satified when the inequality (5.13) is not. A uniform state which is locally asympotically stable state according to Dawid does not necessarily imply stability under a genetic-algorithm adaptation unless Gray coding is applied. Each of the three convergence conditions are satisfied when the inequality (5.13) is satisfied.

Applying concepts for stability is limited in practice by a number of factors. Arifovic's stability under a genetic-algorithm adaptation may be difficult or impossible to determine analytically. Dawid's inequality (5.1) for local asympotic stability is derived under a number of assumptions including an infinitely large population size, low mutation rates, and infinitely many time steps. His inequality (5.1) also specifically applies to single point crossover and binary code. When multiple stable states exist, stability conditions do not provide information about the relative likelihood that any one particular stable uniform state will be attained. Recall from Theorem 5.1 that only uniform states may be absorbing. Hence, we can tell which states are absorbing by examining the inequality (5.13) for each uniform state $u_k \in \mathcal{S}$. This does not in itself guarantee convergence when satisfied. That is, the inequality (5.13) is not sufficient to show that a genetic algorithm converges. However, our approach can be universally applied across all genetic algorithms with a coding mechanism of the type described in Section 3.2. Furthermore, we have made few assumptions about the genetic operators, see Section 5.1. Our Markov model for the genetic algorithm also provides insights into the role input parameters play in simulations. The inequality (5.13) links the parameters of the economic system and parameters of the genetic algorithm in a single condition. For the cobweb model, we have used the inequality (5.13) to derive Theorem 5.8. This result does address the convergence of the genetic algorithm. Similar results must be derived for each economic model studied. It follows that no one particular concept for the stability of uniform states is useful across all models.

5.3 Stationary Distribution and Convergence Rate

The convergence condition given by inequality (5.13) and the analysis of recurrent classes summerized by Theorem 5.8 make definitive statements about the asymptotic behaviour of the genetic algorithm. However, these do not give any insight into the rate at which a stationary distribution is approached. When inequality (5.13) is not satisfied, particularly with an irreducible transition matrix, it is desirable to have some measure of the proportion of time spent in states near the equilibrium. Inequality (5.13) and Theorem 5.8 do not address this.

Using Perron-Frobenius theory given in Section 2.1 it is possible to calculate the stationary distribution and rate of decay numerically for sufficiently small state spaces.

We use input parameters which correspond to those used by Arifovic [3] where the value of A has been modified for single agent experiments with $N = 1$. Our results are then used to support observations made by Arifovic and to explain the simulation output we present in Section 5.2.

Set 1		Set 2	
A	$\frac{2.184}{30}$	A	$\frac{2.296}{30}$
B	0.0152	B	0.0168
x	0	x	0
y	0.016	y	0.016
N	1	N	1

Table 5.8: Comparison of decay rates, parameter sets 1 and 2

Let us consider a single agent experiment, with $nBITS = 3$, and a gridspacing of $\Delta = 4.5/7$ for binary code with input parameters given by set 2 of Table 5.8.

We give all stochastic matrices to three decimal places and all eigenvalues and stationary distributions to four decimal places. The matrix M given by (5.21) represents the bitwise mutation operator. The matrix P given by (5.22) represents the state transition matrix after election and is an irreducible matrix. The selection and crossover matrices are given by the 8×8 identity matrix I.

We calculate the stationary distribution $\pi = (\pi_1, \ldots, \pi_8)$ using Theorem 2.7 as the normalized positive Perron-Frobenius left eigenvector of P

$$\pi = \Big(\; 0.0000, \;\; 0.0005, \;\; 0.0000, \;\; 0.0469, \;\; 0.9515, \;\; 0.0010, \;\; 0.0000, \;\; 0.0000 \; \Big) . \tag{5.20}$$

This stationary distribution gives an example of behaviour common to binary codes under bitwise mutation. A comparatively large proportion of time is spent in the second state, $\pi_2 = 0.0005$. Almost no time is spent in the third state even though this state has a higher fitness than the second. It is evident from the matrix (5.22) that the probability of entry to the third state from the second in a single transition is much lower than the probability of entry to the fourth and sixth state.

The value of $\pi_5 = 0.9515$ is interesting because it is a direct consequence of the Hamming cliff problem, see Section 3.2. Observing the transition matrix (5.22), we can see that once the fifth state is entered it is difficult to leave. The actual value $p_{5,5}$ is one using three decimal places. Once at the fifth state, the fourth state has strictly higher fitness than the fifth. A Hamming cliff arises because there is a comparatively low probability of leaving the current state even though a state with higher fitness exists. State five is binary coded to the string 100 and state four is binary coded to the string 011. The probability of mutating from state four to state five is $(pMUTATE)^3 \approx 3.6 \times 10^{-5}$, displayed as 0.000 in the matrix (5.21).

The rate of decay is given by Theorem 2.8 as the magnitude of the second largest

$$M = \begin{bmatrix} 0.904 & 0.031 & 0.031 & 0.001 & 0.031 & 0.001 & 0.001 & 0.000 \\ 0.031 & 0.904 & 0.001 & 0.031 & 0.001 & 0.031 & 0.000 & 0.001 \\ 0.031 & 0.001 & 0.904 & 0.031 & 0.001 & 0.000 & 0.031 & 0.001 \\ 0.001 & 0.031 & 0.031 & 0.904 & 0.000 & 0.001 & 0.001 & 0.031 \\ 0.031 & 0.001 & 0.001 & 0.000 & 0.904 & 0.031 & 0.031 & 0.001 \\ 0.001 & 0.031 & 0.000 & 0.001 & 0.031 & 0.904 & 0.001 & 0.031 \\ 0.001 & 0.000 & 0.031 & 0.001 & 0.031 & 0.001 & 0.904 & 0.031 \\ 0.000 & 0.001 & 0.001 & 0.031 & 0.001 & 0.031 & 0.031 & 0.904 \end{bmatrix}. \quad (5.21)$$

$$P = \begin{bmatrix} 0.904 & 0.031 & 0.031 & 0.001 & 0.031 & 0.001 & 0.001 & 0.000 \\ 0.000 & 0.935 & 0.001 & 0.031 & 0.001 & 0.031 & 0.000 & 0.001 \\ 0.000 & 0.000 & 0.936 & 0.031 & 0.001 & 0.000 & 0.031 & 0.001 \\ 0.000 & 0.000 & 0.000 & 0.999 & 0.000 & 0.001 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1.000 & 0.000 & 0.000 & 0.000 \\ 0.001 & 0.031 & 0.000 & 0.001 & 0.031 & 0.936 & 0.000 & 0.000 \\ 0.001 & 0.000 & 0.031 & 0.001 & 0.031 & 0.001 & 0.935 & 0.000 \\ 0.000 & 0.001 & 0.001 & 0.031 & 0.001 & 0.031 & 0.031 & 0.904 \end{bmatrix}. \quad (5.22)$$

eigenvalue λ_2 of the transition matrix P

$$\lambda_2 = 0.9992. \tag{5.23}$$

The value of λ_2 indicates a slow decay to the stationary distribution because it is close to one.

We contrast our results for binary coding with those obtained using Gray coding. With Gray code

$$\lambda_2 = 0.9725. \tag{5.24}$$

We expect a simulation using Gray coding to reach a stationary distribution much faster than its binary counterpart.

The stationary distribution under Gray coding is

$$\pi = \Big(\; 0.0000, \;\; 0.0003, \;\; 0.0053, \;\; 0.4863, \;\; 0.4972, \;\; 0.0107, \;\; 0.0002, \;\; 0.0000 \; \Big). \tag{5.25}$$

Gray code does not form Hamming cliffs to the same degree as binary code. It is reasonable to expect in this model that the proportion of time spent in states about the stationary equilibrium to be more equally divided than binary code. This is independent of the actual Euclidean distance between the stationary equilibrium and the value of states four and five so long as neither state is absorbing. Hence, with Gray coding it is unlikely that any state in the stationary distribution have a value of greater than 0.5. This effect explains why convergence of simulations using binary codes may be observed to exhibit a false convergence to unexpected values as noted by Arifovic.

We display the results of similar experiments using a bit length of three to eight bits in Table 5.9 and Table 5.10. Bit lengths of more than eight bits are computationally difficult to analyse due to the exponential increase in the size of the state space

		Binary			Gray		
$nBITS$	Δ	λ_2	$\% < \Delta$	$\% < 2\Delta$	λ_2	$\% < \Delta$	$\% < 2\Delta$
3	0.6714	0.9992	0.9984	0.9994	0.9725	0.9835	0.9995
4	0.3133	1.0000	1.0000	1.0000	0.9750	0.9835	0.9995
5	0.1516	1.0000	1.0000	1.0000	0.9765	0.9835	0.9995
6	0.0770	1.0000	1.0000	1.0000	0.9775	0.9735	0.9995
7	0.0374	1.0000	0.5701	1.0000	0.9778	0.9790	0.9921
8	0.0184	1.0000	1.0000	1.0000	0.9787	0.9838	0.9996

Table 5.9: Results for set 1 of input

		Binary			Gray		
$nBITS$	Δ	λ_2	$\% < \Delta$	$\% < 2\Delta$	λ_2	$\% < \Delta$	$\% < 2\Delta$
3	0.6429	0.9992	0.9984	0.9994	0.9725	0.9835	0.9995
4	0.3033	1.0000	1.0000	1.0000	0.9750	0.9835	0.9995
5	0.1435	1.0000	0.9840	1.0000	0.9746	0.5635	0.9825
6	0.0738	1.0000	0.5831	1.0000	0.9770	0.9790	0.9921
7	0.0350	1.0000	0.8785	0.9080	0.9827	0.9838	0.9995
8	0.0175	1.0000	0.8868	0.9105	0.9850	0.9805	0.9953

Table 5.10: Results for set 2 of input

and numerical errors in P when positive entries become extremely close to zero. The fourth and seventh columns represent the asymptotic proportion of time spent in states less than one gridspace from the stationary equilibrium. The fifth and eighth column represent the asymptotic proportion of time spent in states less than two gridspaces from the stationary equilibrium. Note that the value $vMAX$ is at most $A/(NB)$ which is approximately 4.7 for set 1 and 4.5 for set 2. We chose values for $vMAX$ are such that the inequality (5.16) was not satisfied.

Tables 5.9 and 5.10 indicate that Gray code decays at a faster rate than binary. These results can also be used to support the observation that the rate of decay decreases as the number of states increases. In this experiment increasing the number of bits used to encode values decreases the decay rate. The asymptotic proportion of time spent in states near the equilibrium is not directly related to the size of the state space. This observation is also consistent with simulation results. Asymptotically, the genetic algorithm spends a large proportion of the time within one or two gridspaces of the stationary equilibrium, as previously observed in Table 5.5 and Table 5.6. The stable parameters given by Set 1 have a faster decay than the unstable parameter Set 2. Such behaviour has previously been observed by Arifovic.

Chapter 6

Two Population Models

6.1 An Extension of our Markovian Model to Two Populations

Two population models, such as the overlapping generations model described in Chapter 2, are not formulated as a Markov process in the same way as previously explained. In two population models, individual agents live over two consecutive periods. First period consumption decisions are made by agents in their youth. No consumption decisions are made in old age. Second period consumption of individuals is dependent on both the amount of monetary savings accrued by these individuals in their youth and the price per unit good set by the individuals that are young during these individuals' old age. In such a model, utility is a function of first period consumption decisions made by two populations of agents over two generations. Hence, the state of the Markov process which describes this behaviour involves two populations rather than one.

To differentiate between one and two population models, we denote the state of the Markov process by $[\psi, \varphi]$, where ψ represents a population of first period consump-

tion decisions made by N old agents in their youth and φ represents a population of first period consumption decisions made by N young agents. The set of all coded decisions Ω and the domain $\mathscr{D}$ are unchanged. The set of all possible states $\mathcal{S}$ in a two population model represents all possible combinations of $2N$ ordered decisions from Ω. For some $q \in \mathscr{D}$, $[\psi, \varphi] \in \mathcal{S}$, the fitness function is given by $f_q([\psi, \varphi])$, and the election space by $\mathscr{E}_q([\psi, \varphi])$.

In two population models, the selection operator uses decisions made by old agents in their youth to generate decisions for new young agents. Hence, decisions made by young agents at time $t-2$, who are old at time $t-1$, are used to generate new decisions at time t. This means that if the Markov chain is in state $[\psi, \varphi] \in \mathcal{S}$ at some time t then at time $t+1$ the Markov chain will be in a state $[\varphi, \phi]$, $\phi \in \Omega^N$.

Theorem 6.1. *Let $([\psi_i, \varphi_i])$, $\psi_i, \varphi_i \in \Omega$, $i = 1 \ldots N$, be a population decision vector of the genetic algorithm with gridspacing Δ corresponding to the state $[\psi, \varphi] \in \mathcal{S}$. If $diam(\mathscr{E}_{\psi_i}([\psi, \varphi])) < \Delta$ and $diam(\mathscr{E}_{\varphi_i}([\varphi, \psi])) < \Delta$, $\forall\ i = 1 \ldots N$, then*

(a) the state of the genetic algorithm alternates between $[\psi, \varphi]$ and $[\varphi, \psi]$ in each successive time period, and

(b) ψ and φ are uniform.

Proof: In an argument similar to that used in Theorem 5.1, let us consider the individual k. The election spaces $\mathscr{E}_{\psi_k}([\psi, \varphi])$ and $\mathscr{E}_{\varphi_k}([\varphi, \psi])$ contain all decisions $x, y \in \Omega$ such that $f_x([\psi, \varphi]) \geq f_{\psi_k}([\psi, \varphi])$ and $f_y([\varphi, \psi]) \geq f_{\varphi_k}([\varphi, \psi])$ respectively. If $\text{diam}(\mathscr{E}_{\psi_k}([\psi, \varphi])) < \Delta$ and $\text{diam}(\mathscr{E}_{\varphi_k}([\varphi, \psi])) < \Delta$ then ψ_k and φ_k both have strictly highest possible fitness from amongst all decisions in Ω. For this to hold for all individuals $i = 1 \ldots N$, it must be that $\psi_i = \psi_k$ and $\varphi_i = \varphi_k$, $\forall\ i = 1 \ldots N$. That is both ψ and φ are uniform. As both populations have strictly highest possible fitness, no individual at time t, $t+2$, $t+4$, ..., or at time $t+1$, $t+3$, $t+5$, ..., is permitted to change decision. Hence, the state of the genetic algorithm alternates between $[\psi, \varphi]$ and $[\varphi, \psi]$. ◇

Note that in Theorem 6.1 above, ψ and φ need not necessarily be distinct. We observe that if $\psi = \varphi$ satisfies Theorem 6.1 then the state of the genetic algorithm will never change. We now address the criteria for absorbing states.

Theorem 6.2. *All absorbing states are of the form* $[u_k, u_k] \in \mathcal{S}$, $k \in \Omega$.

Proof: Let the state of the genetic algorithm be $[\psi, \varphi] \in \mathcal{S}$, at some time $t \geq 0$. The state of the genetic algorithm at $t+1$ must be of the form $[\varphi, \phi]$, $\phi \in \Omega^N$. If $\psi \neq \varphi$, the state $[\psi, \varphi]$ is not absorbing. By Theorem 6.1, both ψ and φ must be uniform since all individuals in the two populations must have stictly highest fitness and satisfy $\text{diam}(\mathscr{E}_{\psi_i}([\psi, \varphi])) < \Delta$ and $\text{diam}(\mathscr{E}_{\varphi_i}([\varphi, \psi])) < \Delta$, $\forall\, i = 1 \ldots N$. Hence, all absorbing states are of the form $[u_k, u_k]$, $k \in \Omega$. $\diamond$

Corollary 6.1. *The state* $[u_k, u_k]$, $k \in \Omega$, *is absorbing iff* $diam(\mathscr{E}_k([u_k, u_k])) < \Delta$.

6.2 Constant Money Supply

Arifovic [3] first demonstrated that a genetic algorithm can be applied in an economy of two populations. She examined two overlapping generations models, a model of constant money supply and a model of constant real deficit. Apart from the additional complexity a two population model introduces in comparison to a single population model, a feature of Arifovic's parameterization for the overlapping generations model is multiplicity of equilibria. Arifovic compared her simulation results for the genetic algorithm in these models to rational expectations, experimental evidence [56, 61], and adaptive learning schemes such as price averaging [57] and least squares learning [58]. Differences in the stability of equilibria and convergence were noted between models. In particular, it was found that equilibria might be stable in one formulation for the model and unstable in another. In Arifovic's model of constant money supply, the rational expectations analysis as

given in Chapter 2 predicts that the autarkic state should be the stable attractor for equilibrium price paths $p_0 \neq p^*$. Experimentally the unstable stationary equilibrium p^* was observed. It was also obtained by Arifovic in simulations using the genetic algorithm.

We describe Arifovic's formulation for the genetic algorithm as applied to the overlapping generations model of constant money supply as follows. Initially two populations representing the first period consumption of N individuals at $t = 0$ and N individuals at $t = 1$ are randomly generated over $[0, w^{(1)}]$ by the genetic algorithm. These consumption decisions are decoded according to Section 3.2 to give the values for individual first period consumption $c_{i,0}^{(1)}$ and $c_{i,1}^{(1)}$, $i = 1 \ldots N$. Excess goods are sold by young individuals to old individuals at each period to accumulate monetary savings. At $t = 0, 1$, a young individual i accumulates a saving of

$$s_{i,0} = w^{(1)} - c_{i,0}^{(1)}, \tag{6.1}$$

$$s_{i,1} = w^{(1)} - c_{i,1}^{(1)}. \tag{6.2}$$

With a constant money supply the price of each unit of good at $t = 0, 1$ is

$$p_0 = \frac{Nh}{\sum_{i=1}^{N} s_{i,0}}, \tag{6.3}$$

$$p_1 = \frac{Nh}{\sum_{i=1}^{N} s_{i,1}}, \tag{6.4}$$

where h is the per capita money supply. A young individual i at $t = 0$ and $t = 1$ has $m_{i,0}$ and $m_{i,1}$ units of fiat money respectively to purchase goods in old age

$$m_{i,0} = s_{i,0} p_0, \tag{6.5}$$

$$m_{i,1} = s_{i,1} p_1. \tag{6.6}$$

Second period consumption for an old individual i at $t = 1$ is given by

$$c_{i,1}^{(2)} = w^{(2)} + \frac{m_{i,0}}{p_1}. \tag{6.7}$$

We calculate the utility corresponding to an old individual i with first period consumption at $t = 0$ and second period consumption at $t = 1$ as

$$\mu_{i,0} = c_{i,0}^{(1)} c_{i,1}^{(2)}. \tag{6.8}$$

The first application of genetic operators occurs at period $t = 2$. We use the utility received by agents born at $t = 0$ during selection at $t = 2$ to generate a mating pool of N individuals. Let these individuals have corresponding decisions $\tilde{c}_{i,2}^{(1)}$, $i = 1 \ldots N$. We then apply crossover and mutation to this mating pool. The utility received by each agent living over periods $t = 0$ and $t = 1$ is given by equation (6.8). However, an individual agent at time $t \geq 2$ formulating a first period consumption decision has no method a priori of determining either the price at time t or the price at time $t+1$. Hence, agents at $t = 2$ estimate their potential utility using the price at $t = 0$ as a forecast for the price at $t = 2$ and the price at $t = 1$ as a forecast for the price at $t = 3$. Thus

$$p_t^e = p_{t-2}, \ \forall \, t \geq 2. \tag{6.9}$$

The potential fitness of each individual in the mating pool is calculated as follows. Determine the savings each individual accumulates,

$$\tilde{s}_{i,2} = w^{(1)} - \tilde{c}_{i,2}^{(1)}, \tag{6.10}$$

the corresponding potential holdings of fiat money

$$\begin{aligned} \tilde{m}_{i,2} &= \tilde{s}_{i,2} p_2^e, & (6.11) \\ &= \tilde{s}_{i,2} p_0, & (6.12) \end{aligned}$$

and the corresponding potential second period consumption

$$\begin{aligned} \tilde{c}_{i,3}^{(2)} &= w^{(2)} + \frac{\tilde{m}_{i,2}}{p_3^e}, & (6.13) \\ &= w^{(2)} + \frac{\tilde{m}_{i,2}}{p_1}. & (6.14) \end{aligned}$$

An agent's fitness is then

$$\tilde{\mu}_{i,2} = \tilde{c}_{i,2}^{(1)} \tilde{c}_{i,3}^{(2)}. \tag{6.15}$$

We use the election operator to compare the mating pool and the population born at $t = 0$. The resultant population becomes the representative population of agents at $t = 2$. Once all consumption decisions at $t = 2$ are known, we can calculate p_2 and the utility of all agents who live over periods $t = 1$ and $t = 2$. We continue this process at $t = 3$ and all subsequent iterations of the genetic algorithm, terminating once a specified number of generations have been produced.

Rational expectations analysis of the overlapping generations model as given in Chapter 2 and the price expectation formulation for the genetic algorithm given by equation (6.9) are not consistent. There is no reason a priori why the equilibria resulting from adaptive price expectations would retain the same properties of stability and existence as those equilibria found under a rational expectation analysis. To verify the existence of stationary equilibria under such an adaptive rule, and to determine the stability of these equilibria, let us consider a formulation of the overlapping generations model using adaptive price expectations as given by equation (6.9).

Assuming that all agents maximize utility, then all agents act identically. Utility is given by

$$\mu(c_t^{(1)}, w^{(2)} + r_t^e(w^{(1)} - c_t^{(1)})), \ \forall\, t \geq 2, \tag{6.16}$$

where

$$r_t^e = \frac{p_t^e}{p_{t+1}^e}, \tag{6.17}$$

represents the ratio of forecast values for price at t and $t + 1$. We will assume a product form for utility as given by equation (4.24).

The first order conditions give the optimal first period consumption strategy as a

function of r_t^e

$$c_t^{(1)} = \frac{w^{(2)}}{2}\frac{1}{r_t^e} + \frac{w^{(1)}}{2}. \tag{6.18}$$

Using equation (4.19), equation (6.9), and equation (6.17), we have

$$r_t^e = r_{t-2}, \ \forall \ t \geq 2. \tag{6.19}$$

We substitute this into equation (6.18) to give

$$c_t^{(1)} = \frac{w^{(2)}}{2}\frac{1}{r_{t-2}} + \frac{w^{(1)}}{2}. \tag{6.20}$$

By equation (4.19) and equation (4.21)

$$c_t^{(1)} = \frac{w^{(2)}}{2}\frac{w^{(1)} - c_{t-2}^{(1)}}{w^{(1)} - c_{t-1}^{(1)}} + \frac{w^{(1)}}{2}. \tag{6.21}$$

We obtain a stationary solution to this equation for first period consumption by a substitution of

$$c_t^{(1)} = c_{t-1}^{(1)} = \ldots = c^{(1),*}, \tag{6.22}$$

to give

$$c^{(1),*} = \frac{w^{(1)} + w^{(2)}}{2}. \tag{6.23}$$

This is consistent with equation (4.38) derived under perfect foresight.

To determine the stability of the stationary point (6.23), we write the second order non-linear recurrence relationship (6.21) as a coupled system of first order equations in x and y

$$\begin{aligned} x_{t+1} &= \frac{w^{(2)}}{2}\frac{w^{(1)} - y_t}{w^{(1)} - x_t} + \frac{w^{(1)}}{2}, \\ y_{t+1} &= x_t. \end{aligned} \tag{6.24}$$

Let

$$\begin{aligned} x_{t+1} &= \alpha_{t+1} + \frac{w^{(1)} + w^{(2)}}{2}, \\ y_{t+1} &= \beta_{t+1} + \frac{w^{(1)} + w^{(2)}}{2}. \end{aligned} \tag{6.25}$$

Then the system (6.24) becomes

$$\begin{aligned} \alpha_{t+1} &= \frac{w^{(2)}}{2}\frac{w^{(1)}-w^{(2)}-2\beta_t}{w^{(1)}-w^{(2)}-2\alpha_t}-\frac{w^{(2)}}{2}, \\ \beta_{t+1} &= \alpha_t. \end{aligned} \tag{6.26}$$

A Taylor series expansion about the point $(0,0)$ yields the linearized system

$$\begin{aligned} \alpha_{t+1} &= \frac{w^{(2)}}{w^{(1)}-w^{(2)}}(\alpha_t-\beta_t), \\ \beta_{t+1} &= \alpha_t. \end{aligned} \tag{6.27}$$

That is

$$\begin{bmatrix} \alpha_{t+1} \\ \beta_{t+1} \end{bmatrix} = \begin{bmatrix} \frac{w^{(2)}}{w^{(1)}-w^{(2)}} & \frac{-w^{(2)}}{w^{(1)}-w^{(2)}} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha_t \\ \beta_t \end{bmatrix}. \tag{6.28}$$

The matrix

$$\begin{bmatrix} \frac{w^{(2)}}{w^{(1)}-w^{(2)}} & \frac{-w^{(2)}}{w^{(1)}-w^{(2)}} \\ 1 & 0 \end{bmatrix}, \tag{6.29}$$

has eigenvalues which satisfy the characteristic equation for λ

$$\lambda^2-\frac{w^{(2)}}{w^{(1)}-w^{(2)}}\lambda+\frac{w^{(2)}}{w^{(1)}-w^{(2)}}=0. \tag{6.30}$$

This characteristic equation has roots in λ of

$$\frac{1}{2}\frac{w^{(2)}}{w^{(1)}-w^{(2)}} \pm \frac{1}{2}\sqrt{(\frac{w^{(2)}}{w^{(1)}-w^{(2)}})^2-4\frac{w^{(2)}}{w^{(1)}-w^{(2)}}}. \tag{6.31}$$

If $w^{(1)} > \frac{5}{4}w^{(2)}$ then the characteristic equation (6.30) has imaginary roots with

$$\Re\mathrm{e}(\lambda) = \frac{1}{2}\frac{w^{(2)}}{w^{(1)}-w^{(2)}}, \tag{6.32}$$

$$\Im\mathrm{m}(\lambda) = \frac{1}{2}\sqrt{4\frac{w^{(2)}}{w^{(1)}-w^{(2)}}-(\frac{w^{(2)}}{w^{(1)}-w^{(2)}})^2}. \tag{6.33}$$

If $w^{(1)} = \frac{5}{4}w^{(2)}$ then the characteristic equation (6.30) has two repeated roots

$$\lambda = 2. \tag{6.34}$$

If $w^{(1)} < \frac{5}{4}w^{(2)}$ then the matrix (6.29) has at least one eigenvalue with a magnitude exceeding 2.

This analysis concludes that each eigenvalue of the matrix (6.29) has a magnitude less than one only when the characteristic equation (6.30) has imaginary roots which lie within a ball of radius one in the complex plane. That is

$$\sqrt{\Re\mathrm{e}(\lambda)^2 + \Im\mathrm{m}(\lambda)^2} < 1. \tag{6.35}$$

This inequality becomes

$$\sqrt{\frac{w^{(2)}}{w^{(1)} - w^{(2)}}} < 1, \tag{6.36}$$

or

$$w^{(1)} > 2w^{(2)}. \tag{6.37}$$

Hence, the fixed point $((w^{(1)} + w^{(2)})/2, (w^{(1)} + w^{(2)})/2)$ is a locally asymptotically stable point of the system (6.27) only if $w^{(1)} > 2w^{(2)}$. By Theorem 2.9, the stationary equilibrium $c^{(1),*} = (w^{(1)} + w^{(2)})/2$ of the non-linear system (6.24) is locally asymptotically stable only if $w^{(1)} > 2w^{(2)}$.

The autarkic state $c_t^{(1)} = w^{(1)}$ is a situation for which equation (6.21) is not valid. Once consumption is at a maximum, it will remain so thereafter. Let us consider equation (6.18) displayed in Figure 6.1 which gives the optimal consumption strategy as a function of r_t^e, $t \geq 2$, for all individual agents under the adaptive price expectation (6.9). A price ratio $r_t^e \leq w^{(2)}/w^{(1)}$ results in an optimal consumption strategy $\hat{c}_t^{(1)} > w^{(1)}$. That is, individuals predict that best returns will be obtained if all first period endowment is consumed. Hence, the autarkic state is locally asymptotically stable in a neighbourhood $r_0 \leq w^{(2)}/w^{(1)}$.

	Set 1	Set 2	Set 3
$w^{(1)}$	150	120	100
$w^{(2)}$	10	20	90
h	1000	500	1000
N	30	30	30
$c^{(1),*}$	80	70	95
$c^{(2),*}$	80	70	95
p^*	14.2857	10	200

Table 6.1: Model of constant money supply, parameter sets 1 to 3

	Vector
Set 1	$u_{(80.0587)}$
Set 2	$u_{(70.0293)}$

Table 6.2: Constant money supply, population decision vector, sets 1 and 2

Agents' decisions				
94.7214	96.0899	95.4056	94.0371	94.2326
95.2101	94.6236	94.8191	94.4281	95.5034
95.4056	94.9169	95.6989	95.0146	95.4056
95.8944	93.8416	95.7966	95.2101	94.7214
94.9169	96.0899	94.8191	94.7214	94.6236
94.9169	95.3079	95.2101	94.6236	95.1124

Table 6.3: Constant money supply, population decision vector, set 3

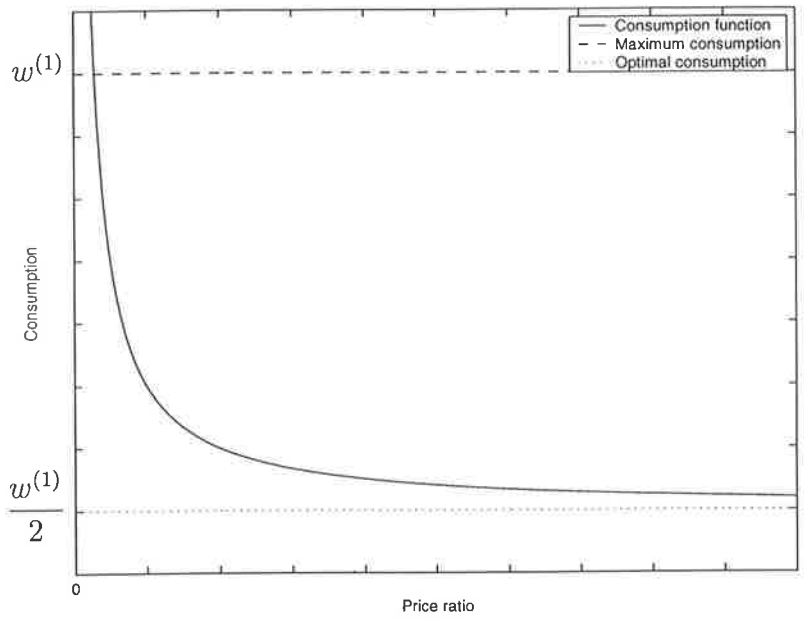


Figure 6.1: First period consumption, $c_t^{(1)}$ against r_t^e

Let us consider the parameters sets given in Table 6.1. These three sets correspond to the first two sets of input studied by Arifovic [3]. Set 1 and Set 2 satisfy the inequality (6.37) so that we expect the fixed point $((w^{(1)} + w^{(2)})/2, (w^{(1)} + w^{(2)})/2)$ given by equation (6.23) to be locally asymptotically stable. We conducted ten thousand iterations of the genetic algorithm under Gray coding with a bit length of ten. In all simulations of Set 1 and Set 2, each of the thirty agents adopted the same decision. Convergence was attained in the majority of simulations within the first two hundred iterations of our algorithm. No change was recorded after this convergence. Set 3 of Table 6.1 represents a set of parameters which do not satisfy the inequality (6.37) for local asymptotic stability. That is, the fixed point given by equation (6.23) is not locally asymptotically stable. A diverse range of values was observed throughout the simulation. Results for the first two parameter sets are tabulated in Table 6.2. The value of the population decision vector at the end of the simulation for parameter Set 3 is recorded in Table 6.3. First period consumption decisions made by agents are displayed as real numbers to four decimal places.

The findings of the genetic algorithm for Set 1 and Set 2 of Table 6.1 are consistent with our stability analysis, except that the autarkic state was not observed during any simulation. We know from our stability analysis that trajectories for first period consumption are attracted to the autarkic state in the region $r_0 < w^{(2)}/w^{(1)}$. Hence, one might expect a simulation of the model to behave similarly. This is not the case. We display in Figure 6.2 and Figure 6.3 a numerical simulation of the trajectories for first period consumption given by equation (6.21), with inital values at $t = 0$ and $t = 1$ equally spaced ten units apart over $[0, w^{(1)}]$. The hashed area marked $\times$ represents the region for which consumption at $t = 2$ is maximal. All consumption paths with initial conditions sufficiently close to equilibrium are convergent to $c^{(1),*}$.

To explore the behaviour of the genetic algorithm initialized close to the autarkic state, we repeated our simulations one hundred times and enforced the initial condition $r_0 < w^{(2)}/w^{(1)}$. That is, the genetic algorithm was initialized for parameter sets 1 and 2 within the hashed region displayed in Figure 6.2 and Figure 6.3 respectively. The genetic algorithm was not observed to converge to the autarkic state in any of these simulations. That is, our results were of a similar kind to those given in Table 6.2.

In simulations of the first two parameter sets given in Table 6.1 it is found that although all individuals at $t = 2$ receive highest fitness for the first period consumption decision $w^{(1)}$, few individuals actually find this decision. This is explained by observing that there is only ever a relatively small probability that the genetic operators will generate such a decision. Hence, this decision will generally not be adopted. Instead, an individual might maintain their previous consumption decision from $t = 0$, or choose some decision between this value and $w^{(1)}$. Now notice that there is a strictly positive probability that a population of decisions at $t = 2$ remains unchanged by the genetic operators from that at $t = 0$. That is all

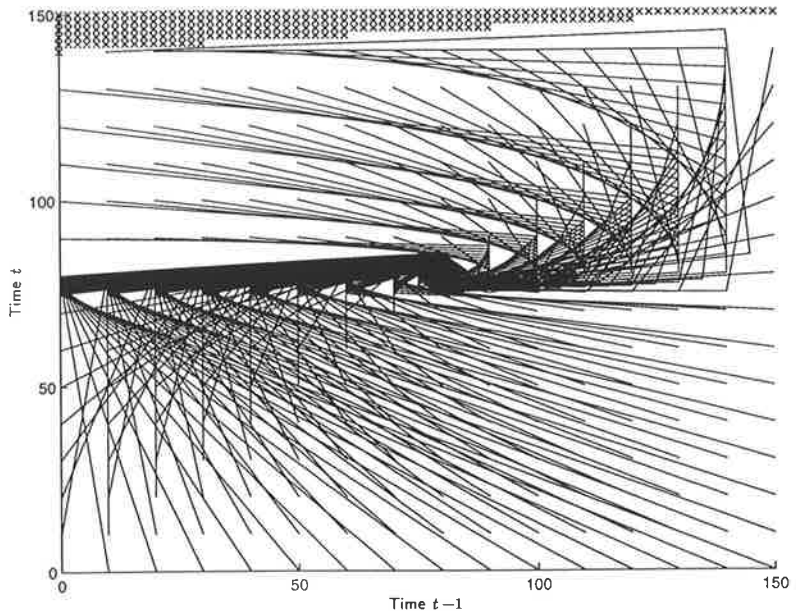


Figure 6.2: Model of constant money supply, consumption paths $(c_{t-1}^{(1)}, c_t^{(1)})$, set 1

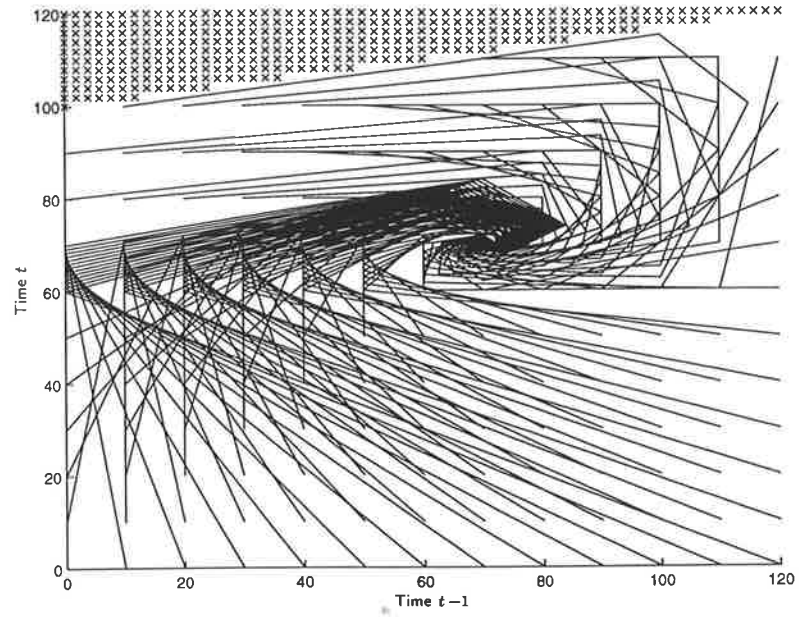


Figure 6.3: Model of constant money supply, consumption paths $(c_{t-1}^{(1)}, c_t^{(1)})$, set 2

individuals maintain their previous first period consumption decision from $t = 0$. From such an observation it is easy to conclude that $r_1 > w^{(1)}/w^{(2)} > w^{(2)}/w^{(1)}$ if $r_0 < w^{(2)}/w^{(1)}$. Hence, the genetic algorithm has a positive probability of leaving the region marked $\times$ in Figure 6.2 and Figure 6.3 so long as neither of the two populations are identically $u_{w^{(1)}}$. Hence, convergence to the stationary equilibrium occurs. The chance that the autarkic state is observed in randomly generated populations is infinitesimal.

The genetic algorithm has given insight into the behaviour of economic systems. In particular, the reasons why individuals are attracted to one particular equilibrium over another is important. As noted above, local asymptotic stability conditions are unlikely to be the sole factor in the convergence of simulations using a genetic algorithm. In Set 3 of Table 6.1 we might have expected convergence to the autarkic state. This was not the case, as illustrated in Table 6.3. We repeated our simulations using one hundred thousand iterations. Convergence was not observed. The convergence of the genetic algorithm gives an insight into the economic model. As remarked by Arifovic [3], convergence in the genetic algorithm and empirical results are in agreement. Hence, genetic algorithms more accurately represents the behaviour of this economic model than do other learning algorithms. In our analysis of the cobweb model, it was meaningful to consider gridspacing in the genetic algorithm to analyse convergence. We also apply this approach to this model, as given below.

We encode the exact value $c^{(1),*}$ to some fixed precision in the genetic algorithm by specifying the gridspacing of Δ. Denote $c_-^{(1)} \leq c^{(1),*}$ and $c_+^{(1)} \geq c^{(1),*}$ the two closest points to $c^{(1),*}$ encoded by the genetic algorithm on either side of $c^{(1),*}$, possibly the point $c^{(1),*}$ itself. Assume that the average first period consumption is less than $c^{(1),*}$. Thus, we can write

$$\frac{1}{N}\sum_{i=1}^{N} c_{i,t}^{(1)} = c^{(1),*} - \delta_t, \tag{6.38}$$

for some $\delta_t > 0$. Hence,

$$\delta_t = c^{(1),*} - \frac{1}{N}\sum_{i=1}^{N} c_{i,t}^{(1)}, \tag{6.39}$$

represents the difference between the equilibrium first period consumption $c^{(1),*}$ given by equation (4.38) and the average first period consumption at each time period. By equation (4.15) and equation (4.14), the corresponding price at time t is given by

$$p_t = \frac{h}{(w^{(1)} - c^{(1),*} + \delta)}. \tag{6.40}$$

Let $(c_{i,t}^{(1)})$ be our population decision vector at time $t \geq 0$ corresponding to the state $\psi(t) \in \mathcal{S}$. From the set of all possible first period consumption decisions at time $t+1$, we use equation (4.24) to calculate the election space $\mathscr{E}_{c^{(1),*}-\delta_t}(\psi(t))$ representing those values with equal or higher fitness to the first period consumption strategy $c^{(1),*} - \delta_t$. Our election space $\mathscr{E}_{c^{(1),*}-\delta_t}(\psi(t))$ is convex and non-empty, taking the form $[a, b]$, $a, b \in \mathscr{D}$. The endpoints a and b are the two real roots of the quadratic equation in ϕ

$$\phi(w^{(2)} + r_{t+1}^e(w^{(1)} - \phi)) = (c^{(1),*} - \delta_t)(w^{(2)} + r_{t+1}^e(w^{(1)} - c^{(1),*} + \delta_t)). \tag{6.41}$$

This has solution $\phi = c^{(1),*} - \delta_t$ and

$$\phi = w^{(1)} + \frac{w^{(2)}}{r_{t+1}^e} - c^{(1),*} + \delta_t. \tag{6.42}$$

Assume that $\delta_t = \delta$ is a constant up to time t and that $c_{i,t}^{(1)} = c^{(1),*} - \delta$, $\forall\ i = 1 \ldots N$. Hence, $\psi(t) = \psi = u_{c^{(1),*}-\delta}$, $r_{t+1}^e = 1$, and our election space is given by $[c^{(1),*} - \delta, w^{(1)} + w^{(2)} - c^{(1),*} + \delta]$. For the state $[u_{c^{(1),*}-\delta}, u_{c^{(1),*}-\delta}]$ to be absorbing no more than one point may be encoded by the genetic algorithm in the election space $\mathscr{E}_{c^{(1),*}-\delta}([u_{c^{(1),*}-\delta}, u_{c^{(1),*}-\delta}])$. That is, $u_{q^*-\delta}$ is an absorbing state according to Theorem 6.1 and Theorem 6.2. We require that

$$w^{(1)} + w^{(2)} - c^{(1),*} + \delta - c^{(1),*} + \delta < \Delta. \tag{6.43}$$

This simplifies to

$$\frac{w^{(1)} + w^{(2)}}{2} - c^{(1),*} + \delta < \frac{\Delta}{2}. \tag{6.44}$$

Substitution of $c^{(1),*} = (w^{(1)} + w^{(2)})/2$ gives

$$\delta < \frac{\Delta}{2}. \tag{6.45}$$

Analogous situations arise when average supply given by equation (6.38) is assumed to be $c^{(1),*} + \delta_t$, $\delta_t > 0$.

We now note that the inequality (6.45) can never be satisfied if both $c_-^{(1)}$ and $c_+^{(1)}$ are equi-distant from $c^{(1),*}$. That is, $c^{(1),*}$ occurs at the exact midpoint of two coded decisions in Ω. Otherwise, we can assume that one of either $c_-^{(1)}$ or $c_+^{(1)}$ lies closer to $c^{(1),*}$ than the other. Denoting δ_{min} as the minimum of $\{c^{(1),*} - c_-^{(1)}, c_+^{(1)} - c^{(1),*}\}$ it follows that the inequality (6.45) is always satisfied by the one unique point either $c_-^{(1)}$ or $c_+^{(1)}$ strictly closest to $c^{(1),*}$.

The situation in which all agents consume all available goods must be treated as a special case during simulations of a genetic algorithm to avoid run time errors when price tends to infinity. That is, we check for such a situation during execution of our algorithm and terminate the process appropriately should the need arise. This autarkic state is always encoded by the genetic algorithm.

Theorem 6.3. *A Markov chain model for genetic algorithm simulation of the overlapping generations model of constant money supply has*

(a) an absorbing state $u_{w^{(1)}}$,

(b) an absorbing state $u_{c_{min}^{(1)}}$, *iff* $c_{min}^{(1)} = arg\ min_{c^{(1)} \in \Omega} |c^{(1),*} - c^{(1)}|$ *is unique, where* $c^{(1),*}$ *is given by equation (6.23)*

Proof: Proof follows from previous observations and Corollary 6.1. ◇

In Set 1 and Set 2 of Table 6.1 convergence to the absorbing state $c_{min}^{(1)}$ given by Theorem 6.3 can reasonably be expected. In Set 3 of Table 6.1 our analysis

indicates convergence should be expected. However, this was not observed in practice. This suggests the rate of decay to the stationary distribution of the chain is slow. That is, although we expect one of the two absorbing states given by Theorem 6.3 to be attained by our Markov chain, the epoch of arrival was not observed. The probability that either of the two distinct absorbing states of the Markov chain are attained cannot in general be calculated without an analysis of the stationary distribution of the process. It is not practical to calculate this distribution for this application of the genetic algorithm due to the size of the state space.

6.3 Volatility

In Section 6.2 we commented that it is not practical to calculate the rate of decay for our Markov process to its stationary distribution. However, we can measure the volatility of the market over time in a manner similar to that used by Arifovic [4]. In our analysis we are interested in variation between coded decisions rather than price. Hence, we define volatility V_T in terms of individual first period consumption $c_{i,t}^{(1)}$, $i = 1 \ldots N$, over the periods $t = 0 \ldots T$

$$V_T = \frac{1}{N(T+1)-1} \sum_{i=1}^{N} \sum_{t=0}^{T} (c_{i,t}^{(1)} - c^{(1),*})^2. \tag{6.46}$$

If we perform R independent simulations for some fixed set of input parameters and fixed value for T, the sample mean volatility $\bar{V}_T$ over these R replications is

$$\bar{V}_T = \frac{1}{R} \sum_{k=1}^{R} V_T^k, \tag{6.47}$$

where V_T^k denotes the volatility corresponding to the k^{th} numbered trial.

The sample variance s_T^2 in the volatility over R replications is

$$s_T^2 = \frac{1}{R-1}\sum_{k=1}^{R}(V_T^k - \bar{V}_T). \tag{6.48}$$

We conducted $R = 100$ independent simulations with $T = 100$. We selected a bit length of ten for binary and Gray coded simulations and used Set 1 of Table 6.1 as input parameters. Simulation statistics are tabulated in Table 6.4. Figure 6.4 displays the sample mean volatility as a function of time for binary, Gray, and real coded simulations. We give some example results for a single simulation run of our genetic algorithm in Appendix B.

	Binary	Gray	Real
$\bar{V}_{100}$	57.2113	53.9031	89.5923
s_{100}^2	13.2858	11.9545	9.1866

Table 6.4: Volatility, output statistics

Let $\bar{V}_B$, $\bar{V}_G$, and $\bar{V}_R$, represent the binary coded, Gray coded, and real coded sample mean volatilities respectively as given in Table 6.4. Let s_B^2, s_G^2, and s_R^2, represent the binary coded, Gray coded, and real coded sample variance in the volatility respectively as given in Table 6.4. Let μ_B, μ_G, and μ_R, represent the binary coded, Gray coded, and real coded volatilities respectively. To statistically test the hypothesis $H_0 : \mu_B = \mu_G$ with alternative hypothesis $H_a : \mu_B > \mu_G$, we use a two sample t-test with $R-1$ degrees of freedom. A t-value statistic of 1.8510 is calculated. This means that our null hypothesis H_0 is rejected in favor of the alternative hypothesis H_a at a significance level of $\alpha = 0.05$ but is retained at a significance level of $\alpha = 0.025$. Hence, in rejecting the null hypothesis H_0 at a significance level of $\alpha = 0.05$ we are 95% confident that H_0 is false and that the true binary coded volatility exceeds the true Gray coded volatility. We compare the real coded sample means with the binary and Gray sample means in a similar

fashion. In the first instance, binary and real coded samples, we propose the null hypothesis $H_1 : \mu_B = \mu_R$, and the alternative hypothesis $H_b : \mu_R > \mu_B$. In the second instance, Gray and real coded samples, we propose the null hypothesis $H_2 : \mu_G = \mu_R$, and the alternative hypothesis $H_c : \mu_R > \mu_G$. Both the null hypotheses H_1 and H_2 are rejected in favor of the alternative hypotheses H_b and H_c respectively with at a significance level of approximately 0%. That is, it is almost certain that H_1 and H_2 are false.

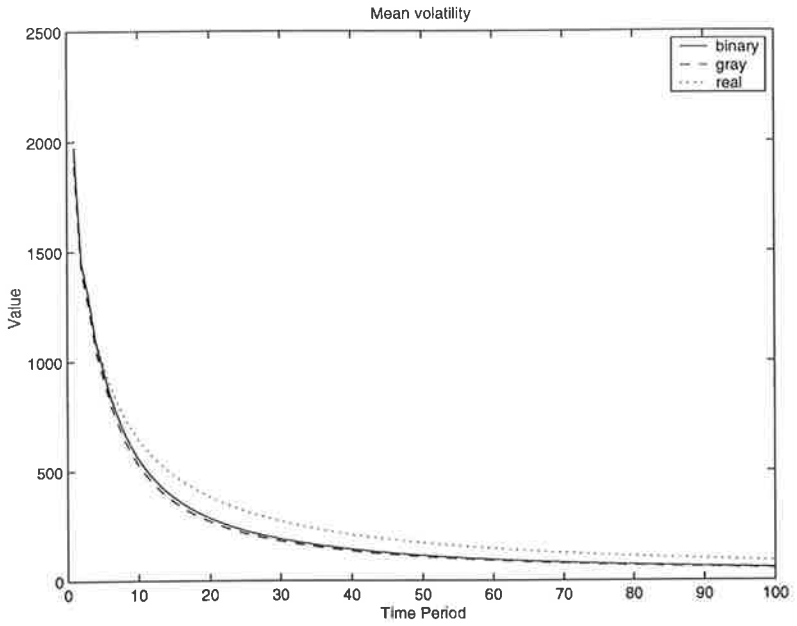


Figure 6.4: Mean volatility $\bar{V}_t$

This experiment demonstrates the importance of coding mechanisms on simulation results. A complete analysis of the stationary distribution of our Markov chain model is not possible for the overlapping generations model. However, these findings are conclusive evidence, for this particular set of parameters, of the inherent differences between genetic algorithms which employ binary, Gray, and real coding mechanisms. Similar results are obtained when alternative parameter sets are considered.

6.4 Constant Real Deficit

The overlapping generations model of constant real deficit has been shown to admit two stationary equilibria for consumption in Chapter 2 for which money is valued. The stationary low inflationary equilibrium is the attracting point for existing genetic algorithms. However, the low inflationary equilibrium is reached under rational expectations dynamics only if first period consumption decisions within the population are initialized at that equilibrium $c_L^{(1),*}$ with $\pi_0 = \pi_L^*$. Simulations performed by Arifovic [5] converge to the low inflationary equilibrium. The high inflationary equilibrium is the stable equilibrium point under rational expectations for all $\pi_0 > \pi_L^*$. One might reasonably ask how these two conflicting statements are reconciled. To address these issues let us start by considering a genetic algorithm and the interactions brought about by its fitness function.

Our implementation of the genetic algorithm for this model closely follows that given in Section 6.2 for the overlapping generations model of constant money supply. The difference in a model of constant deficit is that market price is calculated using equation (4.44).

If price forecasts are made according to $p_t^e = p_{t-2}$ as given in equation (6.9) and utility is given by equation (4.24) then the relationship for consumption is

$$c_t^{(1)} = \frac{w^{(2)}}{2}\frac{1}{r_t^e} + \frac{w^{(1)}}{2}, \tag{6.49}$$

which is identical to equation (6.18). By equation (4.19), equation (4.44), and using $s_t = w^{(1)} - c_t^{(1)}$

$$c_t^{(1)} = \frac{w^{(2)}}{2}\frac{w^{(1)} - c_{t-2}^{(1)}}{w^{(1)} - c_{t-1}^{(1)} - d} + \frac{w^{(1)}}{2}. \tag{6.50}$$

Stationary equilibrium values for first period consumption are given as the two real

roots of the quadratic equation in ϕ

$$2\phi^2 - (3w^{(1)} + w^{(2)} - 2d)\phi + w^{(1)}(w^{(1)} + w^{(2)} - d) = 0. \tag{6.51}$$

This has roots $c_L^{(1),*}$ and $c_H^{(1),*}$ representing the low and high stationary inflation equilibria

$$\frac{1}{4}(3w^{(1)} + w^{(2)} - 2d) \mp \frac{1}{4}\sqrt{(w^{(1)} - w^{(2)})^2 + 4d(d - w^{(1)} - w^{(2)})}, \tag{6.52}$$

respectively. These two equilibria are consistent with those derived under rational expectations.

To determine the stability of the stationary point (6.52), we write the second order non-linear recurrence relationship (6.50) as a coupled system of first order equations in x and y

$$\begin{aligned} x_{t+1} &= \frac{w^{(2)}}{2}\frac{w^{(1)} - y_t}{w^{(1)} - x_t - d} + \frac{w^{(1)}}{2}, \\ y_{t+1} &= x_t. \end{aligned} \tag{6.53}$$

Let

$$\begin{aligned} x_{t+1} &= \alpha_{t+1} + c^{(1),*}, \\ y_{t+1} &= \beta_{t+1} + c^{(1),*}, \end{aligned} \tag{6.54}$$

where $c^{(1),*}$ represents a solution to equation (6.50), either $c_L^{(1),*}$ or $c_H^{(1),*}$.

Then the system (6.53) becomes

$$\begin{aligned} \alpha_{t+1} &= \frac{w^{(2)}}{2}\frac{w^{(1)} - \beta_t - c^{(1),*}}{w^{(1)} - \alpha_t - c^{(1),*} - d} + \frac{w^{(1)}}{2} - c^{(1),*}, \\ \beta_{t+1} &= \alpha_t. \end{aligned} \tag{6.55}$$

A Taylor series expansion about the point $(0, 0)$ yields the linearized system

$$\begin{aligned} \alpha_{t+1} &= \frac{w^{(2)}}{2}\left(\frac{\alpha_t(w^{(1)} - c^{(1),*})}{(w^{(1)} - c^{(1),*} - d)^2} - \frac{\beta_t}{w^{(1)} - c^{(1),*} - d}\right), \\ \beta_{t+1} &= \alpha_t. \end{aligned} \tag{6.56}$$

That is

$$\begin{bmatrix} \alpha_{t+1} \\ \beta_{t+1} \end{bmatrix} = \begin{bmatrix} \frac{w^{(2)}}{2}\frac{w^{(1)}-c^{(1),*}}{(w^{(1)}-c^{(1),*}-d)^2} & -\frac{w^{(2)}}{2}\frac{1}{w^{(1)}-c^{(1),*}-d} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha_t \\ \beta_t \end{bmatrix}. \tag{6.57}$$

The matrix

$$\begin{bmatrix} \frac{w^{(2)}}{2}\frac{w^{(1)}-c^{(1),*}}{(w^{(1)}-c^{(1),*}-d)^2} & -\frac{w^{(2)}}{2}\frac{1}{w^{(1)}-c^{(1),*}-d} \\ 1 & 0 \end{bmatrix}, \tag{6.58}$$

has eigenvalues which satisfy the characteristic equation for λ

$$\lambda^2 - \frac{w^{(2)}}{2}\frac{w^{(1)}-c^{(1),*}}{(w^{(1)}-c^{(1),*}-d)^2}\lambda + \frac{w^{(2)}}{2}\frac{1}{w^{(1)}-c^{(1),*}-d} = 0. \tag{6.59}$$

This characteristic equation has roots in λ of

$$\frac{1}{2}\frac{w^{(2)}}{2}\frac{w^{(1)}-c^{(1),*}}{(w^{(1)}-c^{(1),*}-d)^2} \pm \frac{1}{2}\sqrt{\left(\frac{w^{(2)}}{2}\frac{w^{(1)}-c^{(1),*}}{(w^{(1)}-c^{(1),*}-d)^2}\right)^2 - 4\frac{w^{(2)}}{2}\frac{1}{w^{(1)}-c^{(1),*}-d}}. \tag{6.60}$$

For the linearised system (6.57) to be locally asymptotically stable according to the stability criteria given in Theorem 2.9, it is required that the absolute value of both roots of the characteristic equation be less than one.

All first period consumption decisions $c_t^{(1)} \geq w^{(1)} - d$ represent a situation for which a break down in the economy is observed. All such decisions are locally asymptotically stable.

Let us consider four parameter sets used by Arifovic [5] as given in Table 6.5. We conducted ten thousand iterations of the genetic algorithm under Gray coding with a bit length of ten. In all simulations for Set 1 and Set 2 of our input, each of the thirty individuals adopted the same decision. Convergence was attained in the majority of simulations within the first two hundred iterations of our algorithm.

	Set 1	Set 2	Set 3	Set 4
$w^{(1)}$	150	10	10	2
$w^{(2)}$	30	4	4	1.8
d	15	0.001	0.67544	0.0024
π_L^*	1.38197	1.00033	1.03769	1.57922
π_H^*	3.61803	2.49917	1.07076	1.58306
$c_L^{1,*}$	95.7294	7.00067	8.15843	1.93392
$c_H^{1,*}$	129.270	9.99833	8.16613	1.96368

Table 6.5: Model of constant real deficit, parameter sets 1 to 4

No change was recorded after convergence was attained. Note that the high inflationary stationary equilibrium is an unstable fixed point for all parameter sets. The simulation output corresponding to Set 3 and Set 4 of Table 6.5 varies. We report a typical simulation for which a break down in the economy was observed. This break down occured in both simulations within one hundred iterations of our algorithm. The values of the population decision vectors observed at the end of simulations for the first two parameter sets are tabulated in Table 6.6. Population decision vectors for the last two parameter sets are tabulated in Table 6.7 and Table 6.8. First period consumption decisions made by agents are displayed as real numbers to four decimal places.

Arifovic reports convergence to the low stationary equilibrium for all published sets of parameter values. We obtain convergence in all simulations only for Set 1 and Set 2 of Table 6.5. In both of these sets, the low inflationary stationary equilibrium is locally asymptotically stable and the high inflationary stationary equilibrium is unstable. Graphical stability diagrams for equation (6.50) corresponding to Set 1 and Set 2 are given in in Figure 6.5 and Figure 6.6 respectively. These plot the trajectories for consumption with inital values at $t = 0$ and $t = 1$ equally spaced ten units apart over $[0, w^{(1)}]$. In both figures, the hashed region marked

	Vector
Set 1	$u_{(95.7478)}$
Set 2	$u_{(6.9990)}$

Table 6.6: Constant real deficit, population decision vector, sets 1 and 2

Agents' decisions				
8.9638	8.9736	8.9442	8.8074	10.0000
8.9736	8.9736	10.0000	8.7878	8.8660
9.8826	9.9902	8.8563	8.8563	8.9247
8.9051	8.7781	8.9540	9.9413	9.0127
8.8660	9.3646	8.8856	9.8142	10.0000
8.8856	10.0000	8.8269	9.0518	8.7487

Table 6.7: Constant real deficit, population decision vector, set 3

Agents' decisions				
2.0000	2.0000	2.0000	2.0000	1.9980
2.0000	2.0000	1.9804	2.0000	1.9960
2.0000	2.0000	2.0000	1.9921	2.0000
1.9941	2.0000	2.0000	2.0000	2.0000
2.0000	2.0000	2.0000	1.9882	2.0000
2.0000	1.9863	2.0000	2.0000	2.0000

Table 6.8: Constant real deficit, population decision vector, set 4

$\times$ represents initial conditions such that a breakdown of the economy is observed. That is, price becomes infinite or negative. Random initialization of first period consumption in Set 1 and Set 2 is unlikely to generate initial values within the unstable region. Convergence to the low inflationary equilibrium can then be expected with reasonable certainty. In parameter Set 3 and Set 4 of Table 6.5, the low and high inflationary stationary equilibria are unstable. We demonstrate in Table 6.7 and Table 6.8 that a breakdown in the economy may occur during simulations.

We can examine the stationary equilibria in greater detail by considering our Markov chain model of the genetic algorithm. Through such an analysis we explain why the high inflationary stationary equilibrium is not observed in simulations. The value $c_H^{(1),*}$ is encoded by the genetic algorithm to some precision by specifying the gridspacing Δ. Denote $c_{L-}^{(1)} \leq c_L^{(1),*} \leq c_{L+}^{(1)}$ and $c_{H-}^{(1)} \leq c_H^{(1),*} \leq c_{H+}^{(1)}$ the two closest points to $c_L^{(1),*}$ and $c_H^{(1),*}$ encoded by the genetic algorithm on either side of $c_L^{(1),*}$ and $c_H^{(1),*}$ respectively. Assume that the average first period consumption at time t is less than $c_H^{(1),*}$. Thus, we can write

$$\frac{1}{N}\sum_{i=1}^{N} c_{i,t}^{(1)} = c_H^{(1),*} - \delta_t, \tag{6.61}$$

for some $\delta_t > 0$. Hence,

$$\delta_t = c_H^{(1),*} - \frac{1}{N}\sum_{i=1}^{N} c_{i,t}^{(1)}, \tag{6.62}$$

represents the difference between the high inflationary equilibrium $c^{(1),*}$ given by equation (6.52) and the average first period consumption at each time period.

Let $(c_{i,t}^{(1)})$ be our population decision vector at time $t \geq 0$ corresponding to the state $\psi(t) \in \mathcal{S}$. From the set of all possible first period consumption decisions at time $t+1$, we calculate the subset of decisions representing those values with equal or higher fitness to the first period consumption strategy $c_H^{(1),*} - \delta_t$. Our election

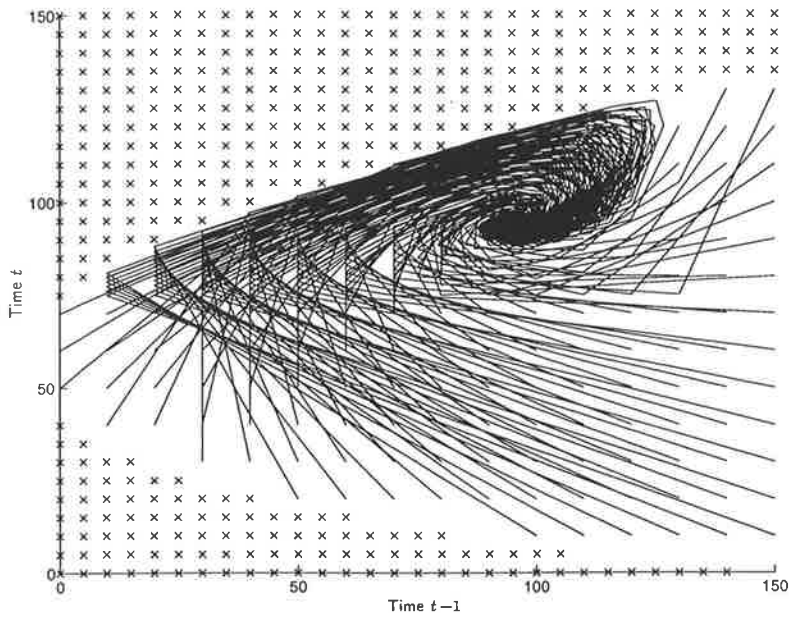


Figure 6.5: Model of constant real deficit, consumption paths $(c_{t-1}^{(1)}, c_t^{(1)})$, set 1

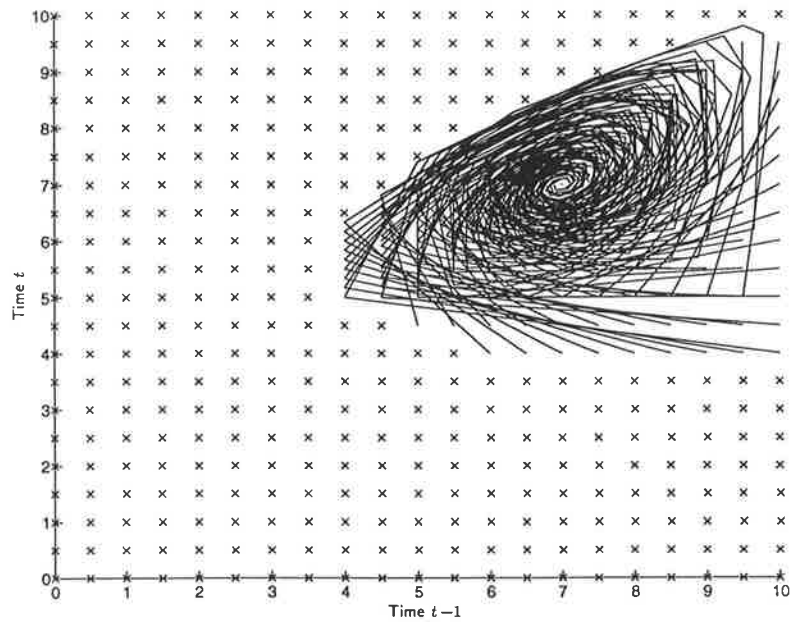


Figure 6.6: Model of constant real deficit, consumption paths $(c_{t-1}^{(1)}, c_t^{(1)})$, set 2

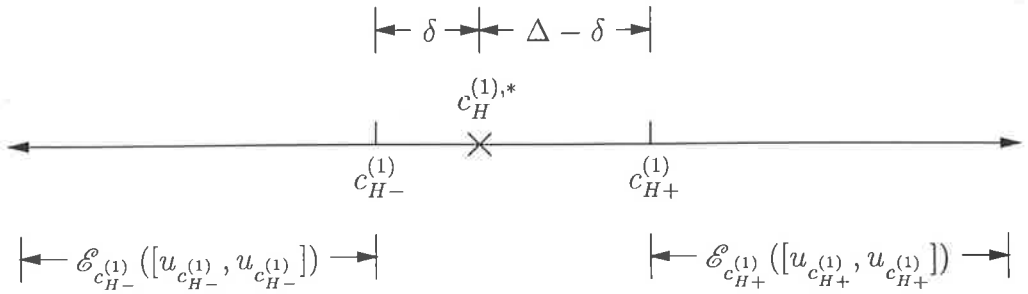


Figure 6.7: Gridspacing and the high inflationary equilibrium

space $\mathscr{E}_{c_H^{(1),*}-\delta_t}(\psi(t))$ is convex and non-empty, taking the form $[b, a]$, $a, b \in \mathscr{D}$. The endpoints a and b are the two real roots of the quadratic equation in ϕ

$$\phi(w^{(2)} + r_{t+1}^e(w^{(1)} - \phi)) = (c^{(1),*} - \delta_t)(w^{(2)} + r_{t+1}^e(w^{(1)} - c^{(1),*} + \delta_t)). \tag{6.63}$$

This has solution $\phi = c_H^{(1),*} - \delta_t$ and

$$\phi = w^{(1)} + \frac{w^{(2)}}{r_{t+1}^e} - c_H^{(1),*} + \delta_t. \tag{6.64}$$

By equation (4.19) and equation (6.19), we have

$$r_{t+1}^e = \frac{p_{t-2}}{p_{t-1}}. \tag{6.65}$$

We substitute for the value of p_{t-1} using equation (4.44) and use equation (4.14) to give

$$r_{t+1}^e = \frac{w^{(1)} - c_{t-1}^{(1)} - d}{w^{(2)} - c_{t-2}^{(1)}} \tag{6.66}$$

Assume that $\delta_t = \delta$ is a constant up to time t and that $c_{i,t}^{(1)} = c_H^{(1),*} - \delta$, $\forall\, i = 1 \ldots N$. Hence, $\psi(t) = \psi = u_{c_H^{(1),*}-\delta}$. Then

$$r_t^e = \frac{w^{(1)} - c_H^{(1),*} + \delta - d}{w^{(2)} - c_H^{(1),*} + \delta}, \tag{6.67}$$

is a constant, r_0.

Now, notice that irrespective of how close average first period consumption is to the high inflationary equilibrium, the dynamics of the system favors values which are further displaced from the stationary equilibrium. The equilibrium value does not appear in the election space. This is also true when average first period consumption given by equation (6.61) is assumed to be $c_H^{(1),*} + \delta_t$, $\delta_t > 0$. Figure 6.7 gives an example in which the election spaces $\mathscr{E}_{c_{H-}^{(1)}}([u_{c_{H-}^{(1)}}, u_{c_{H-}^{(1)}}])$ and $\mathscr{E}_{c_{H+}^{(1)}}([u_{c_{H+}^{(1)}}, u_{c_{H+}^{(1)}}])$ are displayed.

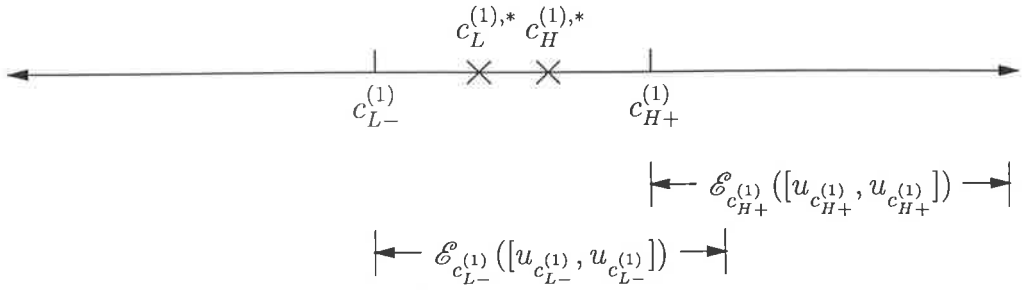


Figure 6.8: Gridspacing and divergence

The election spaces corresponding to states about the low inflationary equilibrium include the equilibrium point for all sufficiently small deviations from the equilibrium value. However, changes in stability occur whenever $c_L^{(1),*}$ approaches $c_H^{(1),*}$. Figure 6.8 gives an example in which convergence is not possible where $\mathscr{E}_{c_{L-}^{(1)}}([u_{c_{L-}^{(1)}}, u_{c_{L-}^{(1)}}])$ and $\mathscr{E}_{c_{H+}^{(1)}}([u_{c_{H+}^{(1)}}, u_{c_{H+}^{(1)}}])$ represent the regions obtained when first period consumption is $c_{L-}^{(1)}$ and $c_{H+}^{(1)}$ respectively. From equation (6.52) the low and high inflationary equilibria approach a single value as $(w^{(1)} - w^{(2)})^{(2)} + 4d(d - w^{(1)} - w^{(2)}) \to 0$. Low and high inflationary equilibrium values may not necessarily be within one gridspace as given in Figure 6.8. In simulations, fluctuations about the low inflationary equilibrium occur. When fluctuations are not sufficiently small, divergence becomes inevitable.

Absorbing states of the Markov process are calculated by considering the diameter of the election space $\mathscr{E}_{c^{(1)}}([u_{c^{(1)}}, u_{c^{(1)}}])$, for all $c^{(1)} \in \Omega$. The diameter of this election

space is

$$\text{diam}(\mathscr{E}_{c^{(1)}}([u_{c^{(1)}}, u_{c^{(1)}}])) = 2|c^{(1)} - \hat{c}^{(1)}|, \tag{6.68}$$

where

$$\hat{c}^{(1)} = \frac{w^{(2)}}{2}\frac{w^{(1)} - c^{(1)}}{w^{(1)} - c^{(1)} - d} + \frac{w^{(1)}}{2}. \tag{6.69}$$

An analysis of the value of the derivative of $\text{diam}(\mathscr{E}_{c^{(1)}}([u_{c^{(1)}}, u_{c^{(1)}}]))$ reveals that

$$\frac{d(\text{diam}(\mathscr{E}_{c^{(1)}}([u_{c^{(1)}}, u_{c^{(1)}}])))}{dc^{(1)}} \to 0, \tag{6.70}$$

as $c^{(1)} \to c_L^{(1),*}$ or as $c^{(1)} \to c_H^{(1),*}$. The diameter of this election space is an increasing function as $c^{(1)} \to 0$ from the right and as $c^{(1)} \to w^{(1)}$ from the left.

Let $c^{(1),*}$ represent a solution of equation (6.51). That is, either $c_L^{(1),*}$ or $c_H^{(1),*}$. Let $\delta > 0$, then a state $[u_{c^{(1),*}+\delta}, u_{c^{(1),*}+\delta}]$ or $[u_{c^{(1),*}-\delta}, u_{c^{(1),*}-\delta}]$ respectively is an absorbing state if

$$2\left|(c^{(1),*} \pm \delta) - \frac{w^{(2)}}{2}\frac{w^{(1)} - (c^{(1),*} \pm \delta)}{w^{(1)} - (c^{(1),*} \pm \delta) - d} - \frac{w^{(1)}}{2}\right| < \Delta. \tag{6.71}$$

This inequality is simplified by a substitution for the first occurrence of the term $c^{(1),*}$

$$\left|w^{(2)}\frac{w^{(1)} - c^{(1),*}}{w^{(1)} - c^{(1),*} - d} - w^{(2)}\frac{w^{(1)} - (c^{(1),*} \pm \delta)}{w^{(1)} - (c^{(1),*} \pm \delta) - d} \pm 2\delta\right| < \Delta, \tag{6.72}$$

or

$$\left|\mp\frac{\delta w^{(2)} d}{(w^{(1)} - c^{(1),*} - d)(w^{(1)} - (c^{(1),*} \pm \delta) - d)} \pm 2\delta\right| < \Delta. \tag{6.73}$$

Four cases arise when we wish to determine the diameter of an election space at the points $c_L^{(1),*} \pm \delta$, $c_H^{(1),*} \pm \delta$.

Below we provide a full derivation for the case corresponding to the point $c_L^{(1),*} + \delta$.

The inequality (6.73) becomes

$$\frac{-\delta w^{(2)} d}{(w^{(1)} - c_L^{(1),*} - d)(w^{(1)} - (c_L^{(1),*} + \delta) - d)} + 2\delta < \Delta. \tag{6.74}$$

This can be rearranged to

$$-\delta w^{(2)} d + (2\delta - \Delta)(w^{(1)} - c_L^{(1),*} - d)(w^{(1)} - (c_L^{(1),*} + \delta) - d) < 0. \tag{6.75}$$

The left side of this inequality describes a quadratic function in δ

$$-2\gamma\delta^2 + (2\gamma^2 + \gamma\Delta - w^{(2)}d)\delta - \gamma^2\Delta, \tag{6.76}$$

where

$$\gamma = w^{(1)} - c_L^{(1),*} - d. \tag{6.77}$$

Note that this quadratic has a negative coefficient of δ^2.

The quadratic equation (6.76) has two real and positive roots if

$$(2\gamma^2 + \gamma\Delta - w^{(2)}d)^2 \geq 8\gamma^3\Delta. \tag{6.78}$$

If the inequality (6.78) is not satisfied, then the quadratic equation (6.76) lies entirely below the horizontal axis. Hence, the inequality (6.75) is satisfied for all values of δ. The situations for which $\delta < 0$ and $\delta > c_H^{(1),*} - c_L^{(1),*}$ are not covered by our initial assumptions. Hence, $\delta \in [0, c_H^{(1),*} - c_L^{(1),*}]$.

If the inequality (6.78) is satisfied, denote the two real roots of equation (6.76) by ρ_1 and ρ_2, $\rho_1 \leq \rho_2$. The inequality (6.75) is satisfied in the regions $(-\infty, \rho_1)$ and $(\rho_2, +\infty)$. The sub-sets $[0, \rho_1)$ and $(\rho_2, c_H^{(1),*} - c_L^{(1),*}]$ represent practical ranges for δ.

The quadratic equation $g : \mathbb{R} \to \mathbb{R}$ given by equation (6.76) is displayed in Figure 6.9 for the case $\rho_1 \neq \rho_2$, $\rho_1, \rho_2 \in \mathbb{R}$. The two intervals of interest for δ are denoted by $\mathcal{I}_1$ and $\mathcal{I}_2$ respectively. The low and high inflationary equilibria occur at $\delta = 0$ and $\delta = c_H^{(1),*} - c_L^{(1),*}$ respectively.

The values of ρ_1 and ρ_2 are given by

$$\frac{1}{4\gamma}(2\gamma^2 + \gamma\Delta - w^{(2)}d) \mp \frac{1}{4\gamma}\sqrt{(2\gamma^2 + \gamma\Delta - w^{(2)}d)^2 - 8\gamma^3\Delta}, \tag{6.79}$$

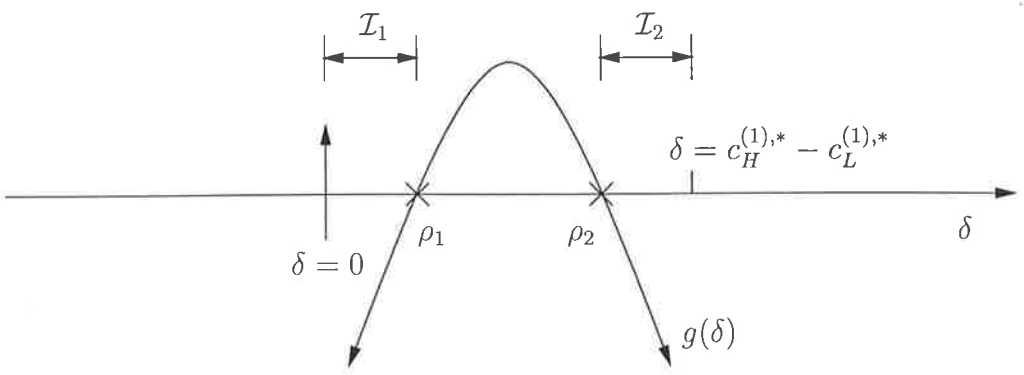


Figure 6.9: Graphical representation of roots and the quadratic $g(\delta)$

respectively.

We have derived a condition on the diameter of an election space by considering the point $c_L^{(1),*} + \delta$ and deriving two intervals of interest, $\mathcal{I}_1$ and $\mathcal{I}_2$. Notice that we may just as easily have considered a point $c_H^{(1),*} - \tilde{\delta}$ and obtained equivalent intervals. Observing the symmetry inherent between these two points, the intervals of interest for $\tilde{\delta}$ are $[0, c_H^{(1),*} - c_L^{(1),*}]$, if equation (6.76) has imaginary roots, and $[0, c_H^{(1),*} - c_L^{(1),*} - \rho_2] \cup [c_H^{(1),*} - c_L^{(1),*} - \rho_1, c_H^{(1),*} - c_L^{(1),*}]$ otherwise. Now, of our four original cases $c_L^{(1),*} \pm \delta$, $c_H^{(1),*} \pm \delta$, we can derive a condition on the diameter of an election space for the remaining two similarly by considering either of $c_L^{(1),*} - \delta$, $c_H^{(1),*} + \delta$. Let us consider the case $c_L^{(1),*} - \delta$.

The inequality (6.73) becomes

$$\frac{-\delta w^{(2)} d}{(w^{(1)} - c_L^{(1),*} - d)(w^{(1)} - (c_L^{(1),*} - \delta) - d)} + 2\delta < \Delta. \tag{6.80}$$

This is re-arranged to

$$-\delta w^{(2)} d + (2\delta - \Delta)(w^{(1)} - c_L^{(1),*} - d)(w^{(1)} - (c_L^{(1),*} - \delta) - d) < 0. \tag{6.81}$$

The left side of this inequality describes a quadratic function in δ

$$2\gamma\delta^2 - (w^{(2)} d + \gamma\Delta - 2\gamma^2)\delta - \gamma^2\Delta, \tag{6.82}$$

where γ is given by equation (6.77).

Note that this quadratic has a positive coefficient of δ^2.

The quadratic equation (6.82) always has two real roots, one positive and one negative. Denote these two real roots by ϱ_1 and ϱ_2, $\varrho_1 \leq \varrho_2$. The inequality (6.81) is satisfied in the region (ϱ_1, ϱ_2). The sub-sets $[0, \varrho_2)$ and $(\varrho_1, -(c_H^{(1),*} - c_L^{(1),*})]$ represent our intervals of interest for δ.

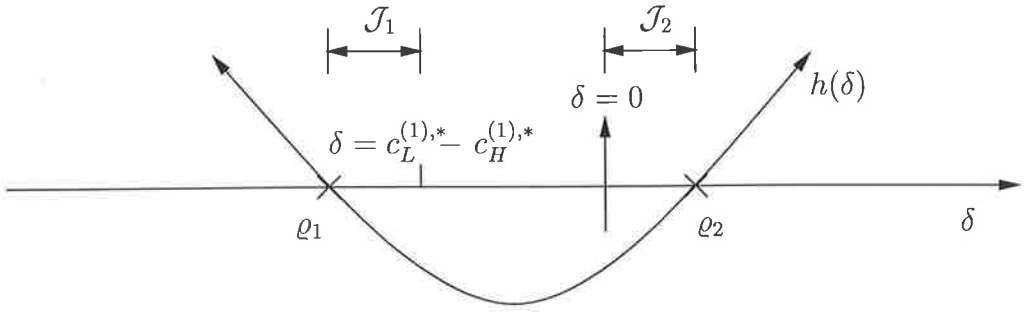


Figure 6.10: Graphical representation of roots and the quadratic $h(\delta)$

The quadratic equation $h : \mathbb{R} \to \mathbb{R}$ given by equation (6.82) is displayed in Figure 6.9. The two intervals of interest for δ are denoted by $\mathcal{J}_1$ and $\mathcal{J}_2$ respectively. The low and high inflationary equilibria occurs at $\delta = -(c_H^{(1),*} - c_L^{(1),*})$ and $\delta = 0$ respectively.

The value of ϱ_1 and ϱ_2 is given by

$$\frac{1}{4\gamma}(w^{(2)}d + \gamma\Delta - 2\gamma^2) \mp \frac{1}{4\gamma}\sqrt{(w^{(2)}d + \gamma\Delta - 2\gamma^2)^2 + 8\gamma^3\Delta}, \tag{6.83}$$

respectively.

A break down in the economy occurs once average first period consumption reaches or exceeds $w^{(1)} - d$. Hence, all states with average first period consumption greater than or equal to $w^{(1)} - d$ are absorbing. This establishes the following theorem.

Theorem 6.4. *A Markov chain model for a genetic algorithm simulation of the overlapping generations model of constant real deficit has*

(a) at most four absorbing states $u_{c_{L-}^{(1)}}$, $u_{c_{L+}^{(1)}}$, $u_{c_{H-}^{(1)}}$, *and* $u_{c_{H+}^{(1)}}$, *for which fiat money is valued, and which satisfy the inequality (6.73),*

(b) an absorbing state for every state $\psi \in \mathcal{S}$ *which represents a break down in the economy,*

(c) at least one absorbing state which represents a break down in the economy.

Proof: Proof follows from previous observations and Corollary 6.1. ◇

For our parameter sets, the state $u_{c_{L-}^{(1)}}$ is an absorbing state for Set 1 and Set 2, each of $u_{c_{L-}^{(1)}}$, $u_{c_{L+}^{(1)}}$, $u_{c_{H-}^{(1)}}$, and $u_{c_{H+}^{(1)}}$ are absorbing states of Set 3, and each of $u_{c_{L-}^{(1)}}$, $u_{c_{L+}^{(1)}}$, and $u_{c_{H-}^{(1)}}$ are absorbing states of Set 4.

In light of these results we can re-consider the output of our genetic algorithm given in Tables 6.6, 6.7, 6.8, the trajectory diagrams given in Figures 6.5, 6.6, and the local asymptotic stability condition analysis given by our discussion of the inequality (6.60). For parameter sets 1 and 2, we have shown that the low stationary inflationary equilibrium is locally asymptotically stable. The state $u_{c_{L-}^{(1)}}$ is the only absorbing state near this equilibrium and was obtained by our genetic algorithm. In parameter sets 3 and 4 we have shown that there are no locally asymptotically stable states apart from those which represent a break down in the economy. While there exist absorbing states which represent both the low and high inflationary equilibrium states, these were not obtained by our genetic algorithm. Instead, as predicted by the stability analysis, a break down in the economy occurred.

Each of the approaches, genetic algorithms, local asymptotic stability, and Markov chains, give unique insight into the behaviour of an economic system. Hence, when applying a genetic algorithm, one should consider all with equal merit as each describes some aspect of the model.

6.5 Comparison to Least Squares Learning

In an overlapping generations model of constant real deficit, Arifovic [5] compared the results of her simulations using a genetic algorithm to Marcet and Sargant's [58] least squares learning algorithm. Where least squares learning predicted divergence of the model, Arifovic found that an application of the genetic algorithm with a price forecast given by equation (6.9) was convergent to the low inflationary equilibrium π_L^*.

In this section, we replace the price forecast (6.9) by the least squares mechanism suggested by Marcet and Sargant. That is, unknown values for price are forecast by extrapolating a line of best fit which interpolates known values for price. With such a mechanism in place, we demonstrate how simulations of the genetic algorithm in an overlapping generations model of constant real deficit can replicate the expected dynamics of least squares learning.

Least squares learning is the forecast rule

$$p_{t+1}^e = \beta_t p_t, \tag{6.84}$$

where

$$\beta_t = \left[\sum_{i=1}^{t-1}(p_{i-1})^2\right]^{-1}\left[\sum_{i=1}^{t-1} p_i p_{i-1}\right]. \tag{6.85}$$

In this experiment, β_t exhibits a functional dependence on all past prices. Our Markov chain model as given in Section 6.1 for the genetic algorithm is inappropriate under such a dependence. Hence, we report simulation results only for this model.

In the implementation of least squares learning, we calculate market prices according to equation (4.44). The price forecast mechanism is changed from equation (6.9) to equation (6.84). To initialize the economy, we generate three populations rather than two as per usual. Hence, we start our simulation at $t = 3$.

	Vector
Set 1	$u_{(95.7478)}$
Set 2	$u_{(6.9990)}$

Table 6.9: Least squares learning, population decision vector, sets 1 and 2

Agents' decisions				
9.7458	9.7751	7.1945	9.7556	9.0029
8.7878	9.9804	9.9706	9.9413	9.7165
9.7849	9.7947	9.6871	9.8044	9.6089
8.7878	8.2306	9.7947	9.7360	9.7458
9.2473	9.8826	9.8240	9.6871	9.9217
9.9511	10.0000	9.8435	9.8826	9.6480

Table 6.10: Least squares learning, population decision vector, set 3

Agents' decisions				
1.9960	2.0000	2.0000	1.9941	1.9765
2.0000	2.0000	2.0000	2.0000	1.9980
2.0000	1.9921	1.9902	2.0000	2.0000
2.0000	2.0000	1.9706	2.0000	1.9608
2.0000	2.0000	2.0000	2.0000	2.0000
1.9980	2.0000	2.0000	2.0000	1.9980

Table 6.11: Least squares learning, population decision vector, set 4

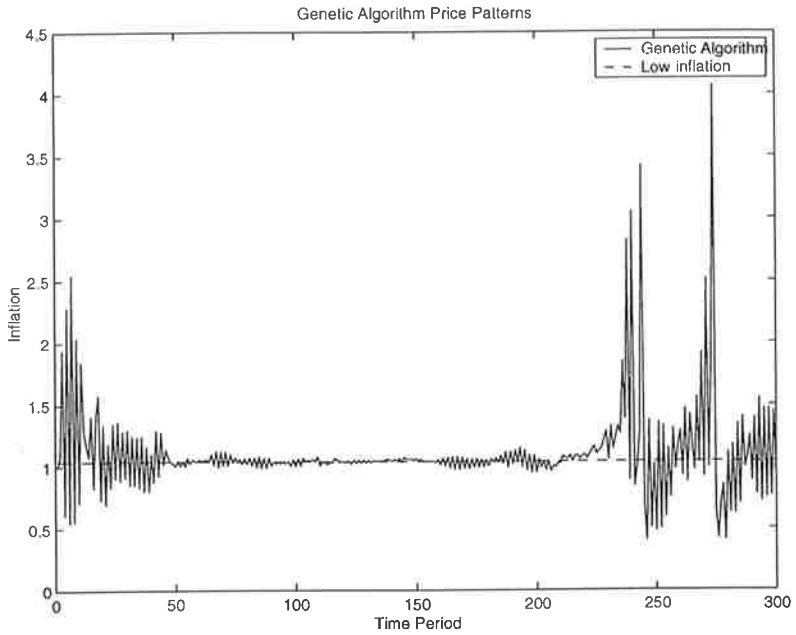


Figure 6.11: Simulated output to set 3 of least squares learning, $\beta_0 = 1.0376$

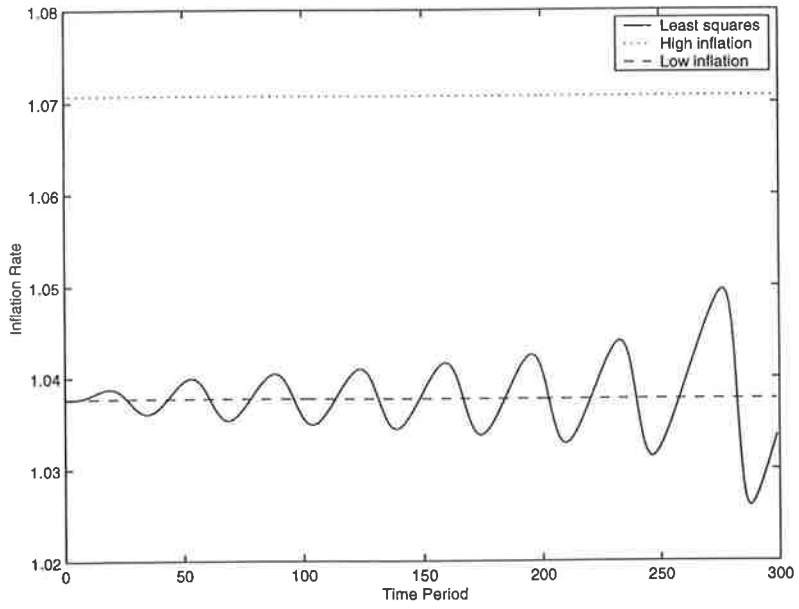


Figure 6.12: Analytic solution to set 3 of least squares learning

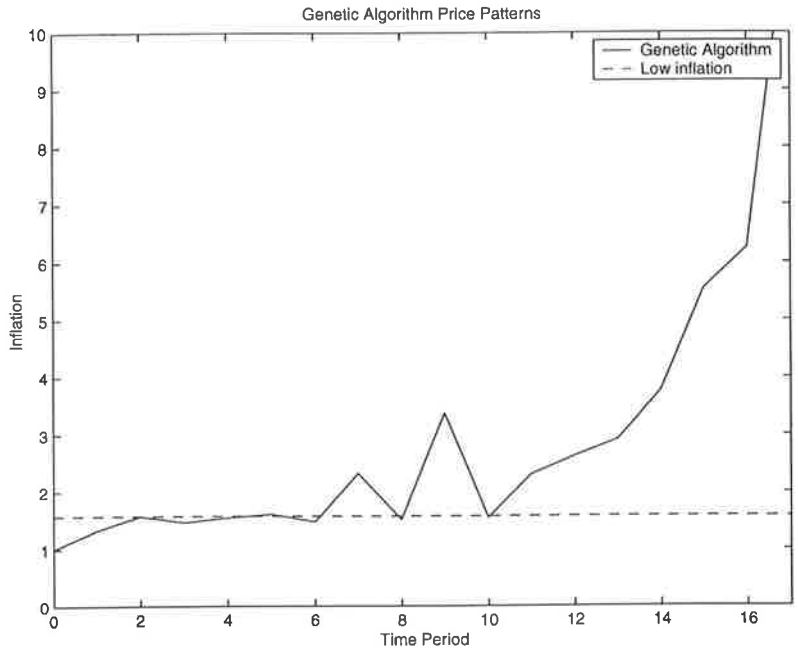


Figure 6.13: Simulated output to set 4 of least squares learning, $\beta_0 = 1.25$

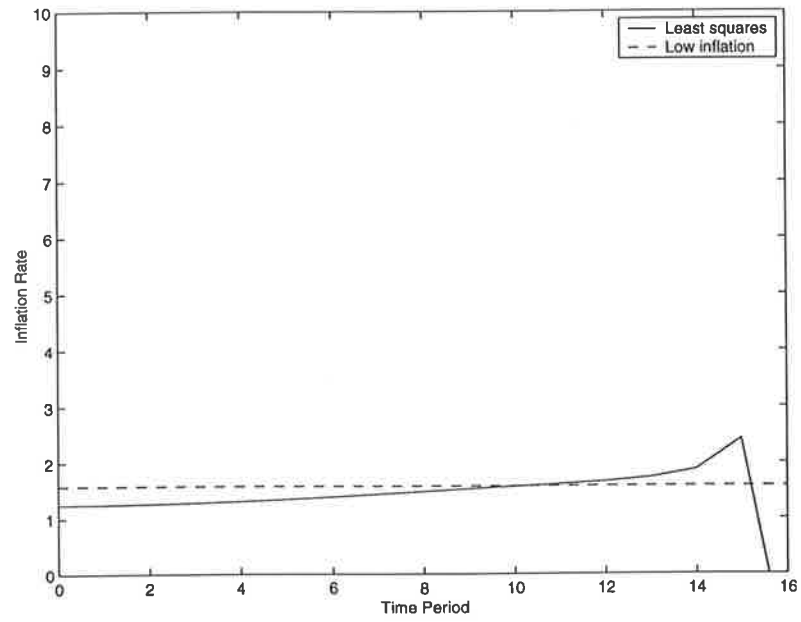


Figure 6.14: Analytic solution to set 4 of least squares learning

We conducted ten thousand iterations of the genetic algorithm under Gray coding with a bit length of ten and recorded simulation results for all parameter sets given in Table 6.5. All agents in simulations for the stable parameters given by Set 1 and Set 2 converged to the low inflationary stationary equilibrium after less than one hundred iterations. This convergence is consistent with the dynamics of least squares learning. Simulations for Sets 3 and 4 diverged and the simulations were stopped with a break down in the economy under less than a thousand iterations. The population decision vectors obtained by our genetic algorithm are recorded for Set 1 and Set 2 in Table 6.9 and for Set 3 and Set 4 in Table 6.10 and Table 6.11 respectively. Agents' decisions in Ω are displayed as real numbers to four decimal places.

Figure 6.11 demonstrates, after a period of adjustment, the ocillatory behaviour of least squares learning in a single simulation of our genetic algorithm for Set 3 of Table 6.5. This can be compared with the analytic solution trajectories given in Figure 6.12. Between iterations fifty and two hundred simulation results closely match qualitative features exhibited by the analytic model. After iteration two hundred the amplitude of oscillations is greater in simulations. A break down in the economy occured shortly after iteration three hundred. Figure 6.13 demonstrates a single simulation of our genetic algorithm for Set 4 of Table 6.5. This simulated behaviour can be compared with the analytic solution trajectories given in Figure 6.14. Again the qualitative features exhibited by the analytic model can be distinguished in simulations. The spikes in inflation occuring in iterations seven and nine are typical to simulations of two populations. Inflation after iteration twelve increased until it became negative in iteration seventeen and a break down in the economy occured.

Note that the convergence of the analytic model for least square learning depends on initial values for β_0. For more on the convergence of this model refer to Marcet

and Sargent [59] and Arifovic [5]. For the purpose of demonstration we report results only for those parameter sets we consider.

	Vector
Set 4	$u_{(8.1036)}$

Table 6.12: Least squares learning, second simulation of set 4

Not all simulations of Set 4 exhibit the same behaviour as we have reported, even given identical initial values of β_0. Convergence to a uniform state can occur, even when this is not consistent with analytic results. To illustrate this we report another set of simulation results Set 4 in Table 6.12 where $\beta_0 = 1.25$. We also notice that the state we report does not consistently appear. That is, other uniform state are observed and these states are typically closer to the low inflationary stationary equilibrium than the state reported in Table 6.12. Even given this, a break down in the economy is the most likely outcome a priori of any simulation for this parameter set.

These experiments show how the behaviour of the least squares learning algorithm can be modelled by a genetic algorithm. Simulations also show that the outcome of any two simulations are not identical. Convergence or a break down in the economy may be observed. Furthermore, this experiment demonstrates the inherent differences between simulations modelling a least squares learning algorithm as opposed to those conducted in Section 6.4. That is, the particular implementation of price forecast mechanism will effect simulation results. Hence, the choice of particular forecast mechanism can not be made arbitrarily.

Chapter 7

Overview

7.1 Conclusions

Economic systems are commonly studied by considering simulation results and deterministic models of behaviour. Genetic have also been proposed as models for the behaviour of economic systems. These models give additional insight into the behaviour of the respective economic system. However, genetic algorithms are inherently stochastic. When using genetic algorithms it becomes necessary to take into account the stochastic effects such algorithms introduce. Hence, it makes sense to consider modelling genetic algorithms using stochastic processes.

To build a mathematical foundation for an analysis of genetic algorithms in economic models, we modelled the representative discrete time stochastic process as a homogeneous Markov chain. With this approach, we defined a state transition matrix corresponding to this Markov chain and investigated the properties the transition matrix exhibited. The form of this transition matrix depended on each particular model, the genetic operators, and other parameters. However, upon consideration of this matrix we established two key conditions, one on the convexity

of the utility function, the other on the genetic mutation operator, which generalized our approach. Using these conditions, we partitioned states of our Markov chain into disjoint communicating classes and established the solidarity properties of recurrence and transience for all states. An analysis of the comminication and hitting probabilities on given states of our chain gave us insight into the long term behaviour of our chain. We used these concepts from Markov theory to derive corresponding conditions for the existence and uniqueness of equilibria, and for convergence and stability in an economic system.

We illustrated how our theoretical results could be applied in practice in context of a cobweb model investigated by Arifovic [4]. For this model, we derived a condition for the existence of absorbing states of our Markov chain and for convergence with probability one to such a state. Furthermore, we discussed the behaviour of chain when no absorbing states existed. We proved that for this model three distinct types of behaviour are possible. First, convergence to a unique absorbing state with probability one was possible. Second, entry to a small recurrent class in a neighbourhood of the equilibrium with probability one was possible when no absorbing states exist. Otherwise, entry to a large recurrent class, possibly consisting of all states, was possible. We compared our results to the local asymptotic stability conditions derived by Dawid [24] and to the stability criteria suggested by Arifovic [6]. To discuss the rate of convergence and the asymptotic properties of the model, we calculated the stationary distribution of our chain for sufficiently small state spaces. These results were consistent with those experimental results provided by Arifovic.

We also considered two overlapping generations economies investigated by Arifovic [5]. First, a model of constant money supply. Second, a model of constant real deficit. Again, we applied a Markov chain analysis to these two population models to derive conditions for the existence of absorbing states. In analysing the

long term behaviour of the chain, we showed that, due to multiplicity of absorbing states, convergence to any particular absorbing state with probability one is not possible. We linearized the corresponding non-linear deterministic models to analyse the local asymptotic stability properties of the two economies. We compared our simulation results with those results derived though our Markov chain analysis and our analysis of the deterministic models to give valuable insight into the behaviour of genetic algorithms. In two population models, it was not practical to calculate a stationary distribution of our Markov chain. However, we did statistically analyse market volatility in our simulations. This statistical analysis showed that market volatility and hence simulation results differ between implementations of our genetic algorithm. In the model of constant real deficit, we discussed an implementation of Marcet and Sargent's [58] least squares learning algorithm. Differences between this implementation and a standard implementation were apparent. Hence, choice of implementation could not be made arbitrarily without affecting the behaviour of our genetic algorithm.

7.2 Closing Remarks

Despite the acceptance of genetic algorithms in economic applications, a mathematical analysis of their behaviour in such applications is rarely attempted. Dawid's [24] work in this field is a notable exception. We have used Markov chains to provide a mathematical foundation for genetic algorithms in economic applications. Using this Markov chain approach, we derived a number of general convergence results and applied these results to a cobweb model and an overlapping generations model. While we have considered only these two economic models studied by Arifovic [3], the potential number of applications for these results is extensive. We hope that these results can be used to dispel some of the common misconceptions

held about the nature of genetic algorithms and to give insight into their behaviour in economic applications.

Appendix A

Parameter Values

Operator probabilities		Input parameters		Interval size	
$pCROSS$	0.6	$nPOP$	30	$vMAX$	$\frac{A}{NB}$
$pMUTATE$	varies	$nGEN$	10,000	$vMIN$	0.0
		$nBITS$	10, 13		

Table A.1: Genetic algorithm parameter values, cobweb model

Operator probabilities		Input parameters		Interval size	
$pCROSS$	0.6	$nPOP$	30	$vMAX$	$w^{(1)}$
$pMUTATE$	varies	$nGEN$	10,000	$vMIN$	0.0
		$nBITS$	10		

Table A.2: Genetic algorithm parameter values, overlapping generations model

A value of $pMUTATE = 0.033$ is used in binary and Gray coded simulations. A value of $pMUTATE = 0.33$ is used in real coded simulations. The number of bits $nBITS$ is immaterial in real coded simulations.

Appendix B

Simulation Plots

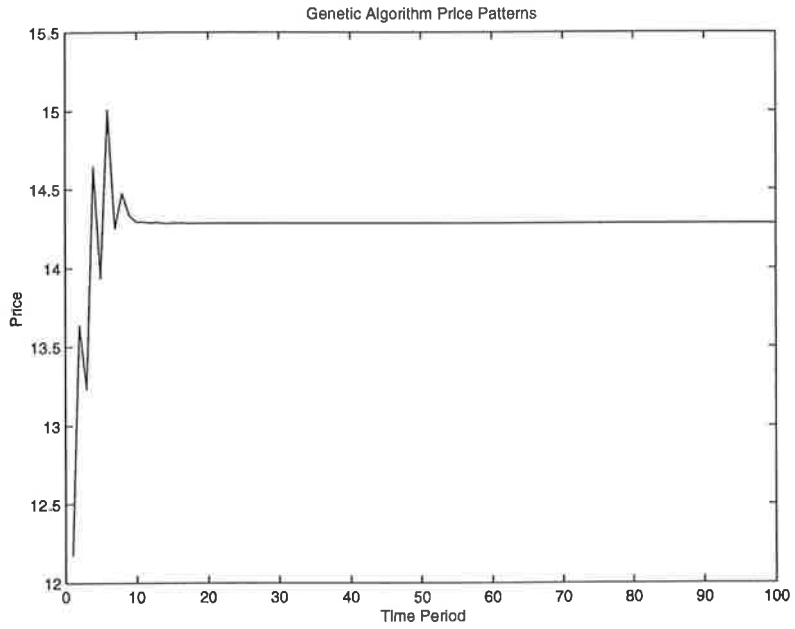


Figure B.1: Example real coded output, single simulation

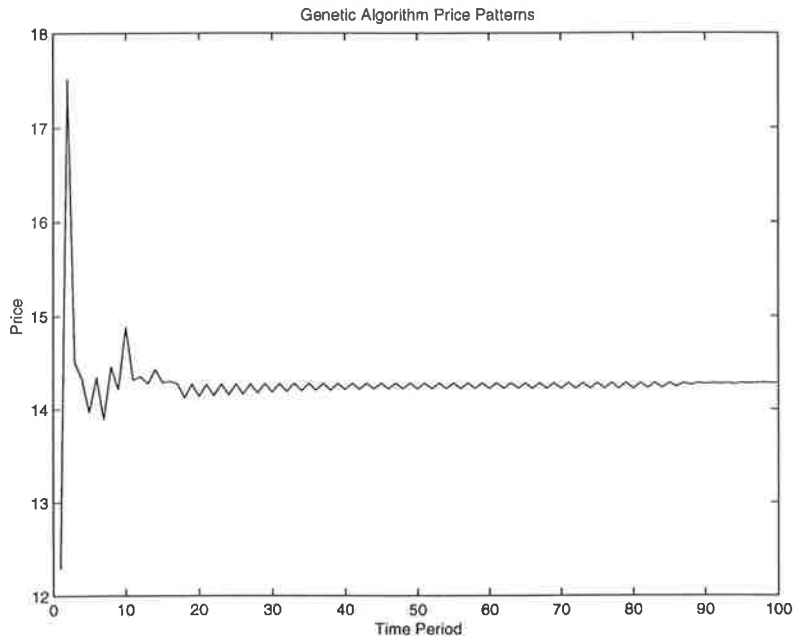


Figure B.2: Example binary coded output, single simulation

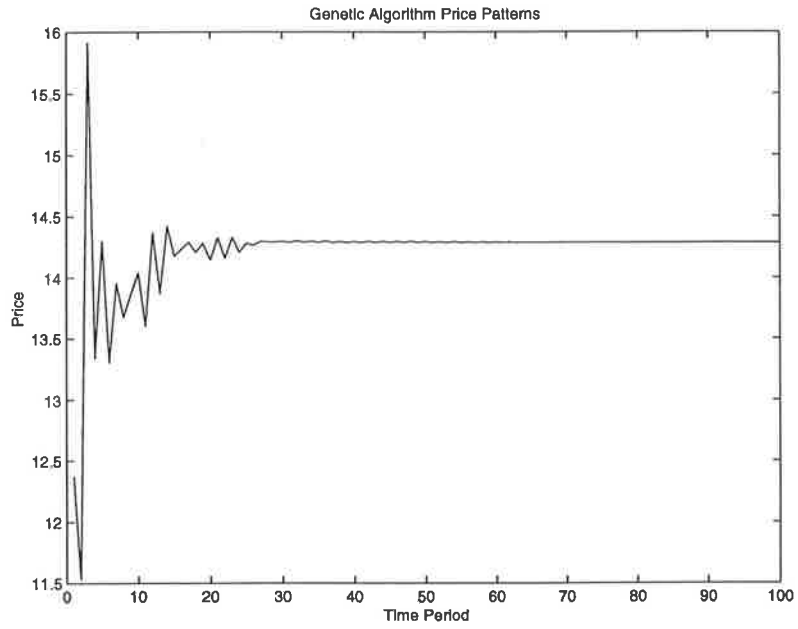


Figure B.3: Example Gray coded output, single simulation

Bibliography

[1] A. B. Abel and B. S. Bernanke. *Macroeconomics.* Addison-Wesley, Reading, Massachusetts, 2nd edition, 1995.

[2] J. T. Alander. Indexed bibliography of genetic algorithms in economics. Report 94-1-ECO, University of Vaasa, Department of Information Technology and Production Economics, 1995. (ftp: ftp.uwasa.fi, cs/report94-1/gaECObib.ps.Z).

[3] J. Arifovic. *Learning by Genetic Algorithms in Economic Environments.* PhD thesis, University of Chicago, Chicago, 1991.

[4] J. Arifovic. Genetic algorithm learning and the cobweb model. *Journal of Economic Dynamics and Control*, 18:3–28, 1994.

[5] J. Arifovic. Genetic algorithms and inflationary economics. *Journal of Monetary Economics*, 36:219–243, 1995.

[6] J. Arifovic. Stability of equilibria under genetic-algorithm adaptation : An analysis. *Macroeconomic Dynamics*, 2:1–21, 1998.

[7] J. Arifovic. Evolutionary algorithms in macroeconomic models. *Macroeconomic dynamics*, 4(3):373–414, 2000.

[8] J. Arifovic and R. Gencay. Statistical properties of genetic learning in a model of exchange rate. *Journal of Economic Dynamics and Control*, 24:981–1005,

2000.

[9] C. Azariadis and R. Guesnerie. Sunspots and cycles. *Review of Economic Studies*, LIII:725–737, 1986.

[10] F. Beckenbach. Learning by genetic algorithms in economics. In T. Brenner, editor, *Computational Techniques for Modelling Learning in Economics*, volume 11 of *Advances in Computational Economics*, pages 73–100. Kluwer Academic, Boston, 1999.

[11] P. Billingsley. *Convergence of Probability Measures*. Wiley series in probability and mathematical statistics. Wiley, New York, 1968.

[12] C. Birchenhall, N. Kastrinos, and S. Metcalfe. Genetic algorithms in evolutionary modelling. *Journal of Evolutionary Economics*, 7:375–393, 1997.

[13] T. A. Bodnovich and B. K. Wong. A bibliography of genetic algorithm business application research : 1988-june 1996. *Expert Systems*, 15(2):75–82, 1998.

[14] W. A. Brock and C. H. Hommes. A rational route to randomness. *Econometrica*, 65(5):1059–1095, 1997.

[15] J. Bullard. Learning equilibria. *Journal of Economic Theory*, 64:468–485, 1994.

[16] J. Bullard and J. Duffy. Learning and the stability of cycles. *Macroeconomic Dynamics*, 2:22–48, 1998.

[17] J. Bullard and J. Duffy. A model of learning and emulation with artificial adaptive agents. *Journal of Economic Dynamics and Control*, 22:179–207, 1998.

[18] J. Bullard and J. Duffy. Using genetic algorithms to model the evolution of heterogeneous beliefs. *Computational Economics*, 13:41–60, 1999.

[19] J. Carlson. The stability of an experimental market with a supply-response lag. *Southern Economic Journal*, 33:305–321, 1967.

[20] J. Carlson. An invariably stable cobweb model. *Review of economic studies*, 69:360–362, 1969.

[21] R.A. Caruana and J.D. Schaffer. Representation and hidden bias: Gray versus binary coding for genetic algorithms. In *Proceedings of the Fifth International Conference on Machine Learning*, pages 153–162, 1988.

[22] C. Chiarella. *The Elements of a Nonlinear Theory of Economic Dynamics*, volume 343 of *Lecture Notes in Economics and Mathematical Systems*. Springer-Verlag, New York, 1990.

[23] C. Darwin. *On the Origin of Species by Means of Natural Selection*. John Murry, London, 1859.

[24] H. Dawid. A markov chain analysis of genetic algorithms with a state dependent fitness function. *Complex Systems*, 8:407–417, 1994.

[25] H. Dawid. Learning of cycles and sunspot equilibria by genetic algorithms. *Journal of Evolutionary Economics*, 6:361–373, 1996.

[26] H. Dawid. *Adaptive Learning by Genetic Algorithms. Analytical Results and Applications to Economic Models*. Springer, Berlin, Heidelberg, 2nd edition, 1999.

[27] H. Dawid and K. Hornik. The dynamics of genetic algorithms in interactive environments. *Journal of Network and Computer Applications*, 19:5–19, 1996.

[28] H. Dawid and M. Kopel. On economic application of the genetic algorithm : A model of the cobweb type. *Journal of Evolutionary Economics*, 8:297–315, 1998.

[29] K. A. De Jong. Genetic algorithms are not function optimizers. In L. D. Whitley, editor, *Foundations of Genetic Algorithms*, volume 2, pages 5–28. Morgan Kauffmann, San Matteo, 1993.

[30] J. Duffy. On learning and the nonuniqueness of equilibrium in an overlapping generations model with fiat money. *Journal of Economic Theory*, 64:541–553, 1994.

[31] B. Edmonds. Modelling socially intelligent agents. *Applied Artifical Intelligence*, 12:677–699, 1996.

[32] B. Edmonds and S. Moss. Modelling bounded rationality using evolutionary techniques. *Lecture Notes in Computer Science*, 1305:31–42, 1997.

[33] S. N. Elaydi. *An Introduction to Difference Equations.* Undergraduate texts in mathematics. Springer, New York, 1996.

[34] G. W. Evans and S. Honkapohja. On the local stability of sunspot equilibria under adaptive learning rules. *Journal of Economic Theory*, 64:142–161, 1994.

[35] G. W. Evans and S. Honkapohja. Local convergence of recursive learning to steady states and cycles in stochastic nonlinear models. *Econometrica*, 63(1):195–206, 1995.

[36] M. Ezekiel. The cobweb theorem. *Quarterly Journal of Economics*, 52:255–280, 1938.

[37] W. Feller. *An Introduction to Probability Theory and its Applications*, volume 1 of *Wiley series in probability and mathematical statistics.* Wiley, New York, 3rd edition, 1968.

[38] S. Forest. Genetic algorithms : Principles of natural selection applied to computation. *Science*, 261:872–878, 1993.

[39] R. Franke. Coevolution and stable adjustment in the cobweb model. *Journal of Evolutionary Economics*, 8:383–406, 1998.

[40] J. Gaffney, K. Parrott, C. Pearce, and F. Salzborn. The cobweb model and a modified genetic algorithm. In W. A. Barnett, C. Chiarella, S. Keen, and H. Schnabl, editors, *Commerce, Complexity, and Evolution. Topics in Economics, Finance, Marketing, and Management : Proceedings of the Twelfth International Symposium in Economic Theory and Econometrics*, pages 268–276, Cambridge, 1996. Cambridge University Press.

[41] G. Giancarlo. *Economic dynamics.* Springer, Berlin, 3rd edition, 1996.

[42] J. K. Goeree and C. H. Hommes. Heterogeneous beliefs and the non-linear cobweb model. *Journal of Economic Dynamics and Control*, 24:761–798, 2000.

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

[44] D.E. Goldberg. Real-coded genetic algorithms, virtual alphabets, and blocking. Report IlliGAL 90001, Illinois Genetic Algorithms Laboratory, Dept. of General Enginerring, University of Illinois, Urbana, Illinois, 1990. (http://citeseer.nj.nec.com/24642.html).

[45] G. R. Grimmett and D. R. Stirzaker. *Probability and Random Processes.* Oxford University Press, New York, 2nd edition, 1992.

[46] F. Hausdorff. *Set Theory.* Chelsea, New York, 2nd edition, 1962.

[47] F. G. Heath. Origins of the binary code. *Scientific American*, 227(2):76–83, 1972.

[48] J. H. Holland. *Adaptation in Natural and Artificial Systems.* University of Michigan Press, Michagan, 1975.

[49] C. Holt and A. Villamil. A laboratory experiment with a single-person cobweb. *Atlantic Economic Journal*, 219:51–54, 1986.

[50] C. Hommes, J. Sonnemans, and H. van de Velden. Expectation formation in a cobweb economy : Some one person experiments. In W. A. Barnett, C. Chiarella, S. Keen, and H. Schnabl, editors, *Interation and Market Structure*, pages 253–266. Springer Verlag, Berlin, 1999.

[51] C. H. Hommes. On the consistency of backward-looking expectations : The case of the cobweb. *Journal of Economic Behaviour and Organization*, 33:333–362, 1998.

[52] S. Karlin and H. M. Taylor. *A First Course in Stochastic Processes.* Academic Press, New York, 2nd edition, 1975.

[53] J. G. Kemeny and J. L. Snell. *Finite Markov Chains.* Undergraduate texts in mathematics. Springer-Verlag, New York, 1976.

[54] J.F.C. Kingman and S.J. Taylor. *Introduction to Measure and Probability.* Cambridge University Press, Cambridge, 1966.

[55] W. W. Leontief. Verzögerte angebotsanpassing und partielles gleichgewicht. *Z. Nationalökon*, 5:670–676, 1934.

[56] S. S. Lim, E. C. Prescott, and S. Sunder. Stationary solution to the overlapping generations model of fiat money : Experimental evidence. *Empirical Economics*, 19:255–277, 1994.

[57] R. E. Lucas. Adaptive behaviour and economic theory. *Journal of Business*, 59(4):s401–s426, 1986.

[58] A. Marcet and T. J. Sargent. Convergence of least squares learning mechanisms in self-referential linear stochastic models. *Journal of Economic Theory*, 48(2):337–368, 1989.

[59] A. Marcet and T. J. Sargent. Least squares learning and the dynamics of hyperinflation. In W. A. Barnett, J. Geweke, and K. Shell, editors, *Economic complexity : Chaos, sunspots, bubbles and nonlinearity.* Cambridge University Press, Cambridge, 1989.

[60] R. Marimon, S. S. Spear, and S. Sunder. Expectationally driven market volatility : An experimental study. *Journal of Economic Theory*, 61:74–103, 1993.

[61] R. Marimon and S. Sunder. Indeterminancy of equilibria in the hyperinflationary world : Experimental evidence. *Econometrica*, 61:1073–1108, 1993.

[62] A. E. Nix and M. D. Vose. Modeling genetic algorithms with markov chains. *Annals of Mathemathematics and Artificial Intelligence*, 5:79–88, 1992.

[63] C. E. M. Pearce. The convergence of genetic learning algorithms, with particular reference to recent cobweb models. In W. A. Barnett, C. Chiarella, S. Keen, and H. Schnabl, editors, *Commerce, Complexity, and Evolution. Topics in Economics, Finance, Marketing, and Management : Proceedings of the Twelfth International Symposium in Economic Theory and Econometrics*, pages 277–288, Cambridge, 1996. Cambridge University Press.

[64] C. C. Pinter. *Set Theory.* Addison-Wesley, Readin, Massachusetts, 1971.

[65] T. Riechmann. Learning and behavioural stability : An economic interpretation of genetic algorithms. *Journal of Evolutionary Economics*, 9:225–242, 1999.

[66] S. M. Ross. *Introduction to Probability Models.* Academic Press, San Diego, California, 7th edition, 1980.

[67] G. Rudolph. Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Networks*, 5(1):96–101, 1994.

[68] P. Samuelson. An exact consumption-loan model of interest with or without the social contrivance of money. *Journal of Political Economy*, 66:467–482, 1958.

[69] E. Seneta. *Non-negative Matrices and Markov Chains.* Springer series in statistics. Springer-Verlag, New York, 2nd edition, 1981.

[70] W. M. Spears, K. A. De Jong, T. Baeck, D. B. Fogel, and H. de Garis. An overview of evolutionary computation. In P. B. Brazdil, editor, *Proceedings of the European Conference on Machine Learning (ECML-93)*, volume 667, pages 442–459. Springer Verlag, 1993.

[71] F. A. Valentine. *Convex Sets.* McGraw-Hill series in higher mathematics. McGraw-Hill, New York, 2nd edition, 1964.

[72] M. D. Vose and G.E. Liepins. Punctuated equilibria in genetic search. *Complex Systems*, 5:31–44, 1991.

[73] C. P. Wellford. *A laboratory analysis of price dynamics and expectations in the cobweb model.* Discussion paper no 89-15. Tuscon, University of Arizona, 1989.

[74] P. Whittle. *Probability via Expectation.* Springer texts in statistics. Springer-Verlag, New York, 3rd edition, 1992.

[75] M. Woodford. Learning to believe in sunspots. *Econometrica*, 58(2):277–307, 1990.