WORKING PAPER SERIES

Using Genetic Algorithms to Model the Evolution of Heterogeneous Beliefs

James Bullard
John Duffy

Working Paper 1994-028A
http://research.stlouisfed.org/wp/1994/94-028.pdf

PUBLISHED: Computational Economics, February 1999.

FEDERAL RESERVE BANK OF ST. LOUIS
Research Division
411 Locust Street
St. Louis, MO 63102

The views expressed are those of the individual authors and do not necessarily reflect official positions of the Federal Reserve Bank of St. Louis, the Federal Reserve System, or the Board of Governors.

Federal Reserve Bank of St. Louis Working Papers are preliminary materials circulated to stimulate discussion and critical comment. References in publications to Federal Reserve Bank of St. Louis Working Papers (other than an acknowledgment that the writer has had access to unpublished material) should be cleared with the author or authors.

Photo courtesy of The Gateway Arch, St. Louis, MO. www.gatewayarch.com

USING GENETIC ALGORITHMS TO MODEL THE EVOLUTION OF HETEROGENEOUS BELIEFS

ABSTRACT

Genetic algorithms have been used by economists to model the process by which a population of heterogeneous agents learn how to optimize a given objective. However, most general equilibrium models in use today presume that agents already know how to optimize. If agents face any uncertainty, it is typically with regard to their expectations about the future. In this paper, we show how a genetic algorithm can be used to model the process by which a population of agents with heterogeneous beliefs learns how to form rational expectation forecasts. We retain the assumption that agents optimally solve their maximization problem at each date given their beliefs at each date. Agents initially lack the ability to form rational expectations forecasts and have, instead, heterogeneous beliefs about the future. Using a genetic algorithm to model the evolution of these beliefs, we find that our population of artificial adaptive agents eventually coordinates their beliefs so as to achieve a rational expectations equilibrium of the model. We also report the results of a number of computational experiments that were performed using our genetic algorithm model.

KEYWORDS: learning, genetic algorithm, beliefs

JEL CLASSIFICATION: C63, D84

James Bullard
Senior Economist
Federal Reserve Bank of St. Louis
411 Locust Street
St. Louis, MO 63102

John Duffy
Department of Economics
University of Pittsburgh
Pittsburgh, PA 15260

I. Introduction

The rational expectations assumption has become a standard feature of general equilibrium economic theorizing. Many economists argue that while such an assumption may seem extreme, it can be justified as the eventual outcome of a (usually unspecified) learning process. This argument has led many to theorize as to how such a learning process might work and whether systems with expectations so defined would actually converge to a rational expectations equilibrium. Some authors have begun to investigate general equilibrium learning models based on *genetic algorithms*, with largely promising results.[1]

In this paper we illustrate that the modelling of such learning processes using genetic algorithms can be carried out in two different ways. In the first method, agents are viewed as *learning how to optimize* in the sense that they experiment with different values of their choice variable based on which values worked well for other agents in the past. All of the general equilibrium applications of genetic algorithms of which we are aware use this first method. In the second method, agents are viewed as *learning how to forecast*, meaning that they select a value for their forecast variable based on which values worked well in the past, and then solve a maximization problem to find the value of their choice variable given their forecast.[2] With this second method, the assumption that agents maximize utility is maintained. In this paper we provide an example of the second method and discuss its strengths and weaknesses.

In order to define an evolutionary approach to an individual agents' problem in the general equilibrium–homogeneous preferences environment that we consider, it is necessary both to define how the agent views the future and how the agent chooses a value of the choice

[1]See, for example, Arifovic (1992, 1994) and Bullard and Duffy (1993, 1994). See also the discussion in Sargent (1993). For an introduction to genetic algorithms see the book by Goldberg (1989).

[2]See Marimon and Sunder (1994) for a discussion of the same distinction between *learning how to optimize* and *learning how to forecast* in the context of setting up overlapping generations experiments with human subjects.

variable. In the *learning how to optimize* implementation of genetic algorithm learning, one assumes (implicitly or explicitly) that all agents have the same view of the future, and that the genetic algorithm is used to assign agents a value of the choice variable given the set of commonly held expectations. Clearly, if all the agents optimized a common objective given these common expectations, all agents would make the same decision and the heterogeneity on which the genetic algorithm depends would be lost.[3] Rather than optimize, the agents simply choose values of the choice variable according to the genetic algorithm assignment. This method has been successfully applied in several recent papers. In this case, however, the researcher is weakening both the assumption that agents have rational expectations (expectations are updated adaptively, since rational expectations are not well defined) as well as the assumption that agents optimize given their expectations. Nevertheless, once equilibrium is attained, beliefs and actions of all agents are consistent with rational expectations and utility maximization.

In applying genetic algorithms to learning problems, many economists might want to relax the rational expectations assumption without relaxing the optimization postulate. One reason for adopting such an approach is that model economies where both assumptions hold tend to have multiple equilibria. It is not clear what an individual agent with rational expectations should believe since there are multiple outcomes that are consistent with equilibrium, and which one is "right" depends on what all the other agents believe. Achieving one of these equilibria requires a certain *coordination* of beliefs among all of the agents in the population.

In the example of genetic algorithm learning that we present in this paper, agents are viewed as *learning how to forecast.* Agents initially have heterogeneous views of the future which they use to individually solve their common maximization problem. The

[3]In most learning models in a macroeconomic context, including many with least squares learning, there is, in effect, a representative agent who maximizes given expectations, and the expectations are updated according to some fixed adaptive rule.

genetic algorithm is used only to update beliefs. Thus, in the example we develop, the only departure from standard assumptions is that agents initially have heterogeneous beliefs which they eventually learn to coordinate in order to achieve an equilibrium outcome. We believe that this exercise is an especially useful application of genetic algorithm learning, as it is applied to an area of economic modelling for which economists have the least knowledge: the formation and evolution of expectations. The fact that expectations are easily modeled and updated using a genetic algorithm is interesting in itself. Our example also helps illustrate the fact that genetic algorithms provide us with an extremely flexible tool that can be used in many different ways.

The model we use is a two–period endowment overlapping generations economy with fiat money. We outline the model in the next section. In section three we describe the model under learning, and in section four we show how to apply a genetic algorithm in a manner consistent with utility maximization. The final sections display the results of some computational experiments and provide a summary of the main points.

II. The Model

Time t is discrete with integer $t \in (-\infty, \infty)$. Agents live for two periods and seek to maximize utility over this two period horizon. The population of agents alive at any date t is fixed at $2 \times N$ where N is the number of agents in each generation. There is a single perishable consumption good and a fixed supply of fiat money. Agents are endowed with an amount ω_1 of the consumption good in the first period of life, and an amount ω_2 of the consumption good in the second period of life, where $\omega_1 > \omega_2 > 0$. In the first period of life, agents may choose to simply consume their endowments, or they may choose to save a portion of their first period endowment in order to augment consumption in the second period of life. Since the consumption good is perishable, agents in this economy can save only by trading a portion of their consumption good for fiat money. In the second period of life, they can use any fiat money they acquired in the first period to purchase amounts of the consumption good in excess of their second period endowment.

A representative agent born at time t solves the following problem:

$$\max_{c_t(t), c_t(t+1)} U = \ln c_t(t) + \ln c_t(t+1)$$

subject to:

$$c_t(t) + c_{t+1}(t+1)\beta(t) \leq \omega_1 + \omega_2 \beta(t).$$

where $c_t(t+i)$, $i = 0, 1$ denotes consumption by the agent born at time t in period $t+i$ and $\beta(t)$ denotes the time t forecast of the gross inflation factor between dates t and $t+1$:

$$F[P(t+1)] = \beta(t)P(t)$$

where $P(t)$ denotes the time t price of the consumption good in terms of fiat money, and $F[P(t+1)]$ is the time t forecast of the price of the consumption good at time $t+1$. This forecast can be formed in any number of ways. For the moment, we consider the case where all agents have *perfect foresight,* in which case $F[P(t+1)] = P(t+1)$, so that $\beta(t) = \frac{P(t+1)}{P(t)}$.

Combining the first order conditions with the budget constraint, one obtains:

$$c_t(t) = \frac{\omega_2}{2}\left[\lambda + \beta(t)\right],$$

where $\lambda = \omega_1/\omega_2$. It follows that the representative agent's savings decision at time t is given by:

$$s_t(t) = \omega_1 - c_t(t) = \frac{\omega_2}{2}\left[\lambda - \beta(t)\right]. \tag{1}$$

Fiat money is introduced into the economy by a government that endures forever. The government prints fiat money at each date t in the amount $M(t)$ per capita. It uses this money to purchase g units per capita of the consumption good in every period:

$$P(t)g = M(t) - M(t-1). \tag{2}$$

It is assumed that government consumption does not yield agents any additional utility.

Since agents can save only by holding fiat money, the money market clearing condition is that *aggregate savings*, $Ns_t(t)$, equals the aggregate stock of real money balances $N\frac{M(t)}{P(t)}$, at every date t, or more simply that

$$s_t(t) = \frac{M(t)}{P(t)}. \tag{3}$$

By Walras' law, market clearing in the money market implies market clearing in the consumption good market as well. Substituting equations (1–2) into (3) and rearranging, we obtain a first order difference equation in $\beta(t)$:

$$\beta(t) = 1 + \lambda - \frac{2g}{\omega_2} - \frac{\lambda}{\beta(t-1)}. \tag{4}$$

Equation (4) has two stationary equilibrium solutions, given by

$$\beta^{H,\,L} = \frac{1 + \lambda - \frac{2g}{\omega_2} \pm \sqrt{\left(1 + \lambda - \frac{2g}{\omega_2}\right)^2 - 4\lambda}}{2}$$

where β^H denotes the *higher* of the two stationary values and β^L denotes the *lower* stationary value. These two solutions will be real valued if government purchases of the consumption good are not too great. In particular, we require that

$$0 < g < \frac{\omega_2}{2}\left[1 + \lambda - 2\sqrt{\lambda}\right]. \tag{5}$$

It is easily established that the Pareto superior steady state is the low inflation steady state, β^L. Under the assumption of perfect foresight this solution is *locally unstable*. The other steady state, β^H, is locally stable in the perfect foresight dynamics, and is an attractor for all initial values of the gross inflation factor $\beta(0) \in (\beta^L, \lambda)$.[4]

The two stationary equilibria are shown in Figure 1, which depicts the qualitative graph of equation (4) for a particular case that will be studied later in the paper. As government

[4]For an analysis of the dynamics of the model under perfect foresight see Sargent and Wallace (1981). For an analysis of the dynamics under a least squares learning scheme see Marcet and Sargent (1989).

expenditures per capita (g) increases, the curve representing equation (4) shifts downward and the two stationary equilibria, β^L and β^H move closer together. Thus, an increase in government purchases leads to an *increase* in the value of the low stationary inflation factor β^L, and a decrease in the value of the high stationary inflation factor, β^H.

III. Learning

The assumption that agents have perfect foresight is useful for understanding the dynamics of the model *when agents know the model.* We now relax the assumption that agents have perfect foresight knowledge of future prices. We assume instead that all N agents forecast future prices using the linear model:

$$F_i[P(t+1)] = b_i(t)P(t), \tag{6}$$

where $b_i(t)$ denotes the parameter that agent $i = 1, 2, ..., N$ of generation t uses to forecast next period's price. While all N agents use the same specification (6) for their forecast rule, each agent may have a different belief regarding the appropriate value for the unknown parameter b. We further restrict agent's beliefs regarding the parameter b to fall in the interval:

$$0 \leq b_i(t) \leq \lambda \text{ for all } i,\ t.$$

The lower bound ensures that price forecasts are always positive. The upper bound of λ represents the highest inflation factor that agents would need to forecast and still be consistent with equilibrium. Inflation factors above λ would imply that agents always consumed their endowments as the equation for individual savings (1) makes clear.

Each agent uses their forecast from equation (6) to solve the constrained maximization problem given in the previous section. The more accurate the agent's forecast, the higher is the agent's utility. Therefore, it is in the agent's interest to approximate the "true" value of the unknown parameter b as closely as possible. Of course, while agents are learning, this "true" value for the gross inflation factor will depend on all agents' beliefs, and will therefore be *time–varying.*

We stress that the specification for the agent's forecast rule (6) is consistent with the actual law of motion for prices when agents have perfect foresight. The consistency of the agent's forecast rule with the actual law of motion enables us to examine whether or not agents can learn the true model. Agents will have *coordinated beliefs* or, alternatively, ***will have converged upon a stationary equilibrium*** if $b_1(t) = b_2(t) = ... = b_N(t) = \beta(t) = P(t+1)/P(t)$, that is, if all agents have identical forecast rules, and their forecasts are always correct.

IV. The Evolution of Beliefs

We now apply a genetic algorithm in order to describe how the parameter $b_i(t)$ evolves over time. We first describe how beliefs are coded in a binary string, and then we illustrate how genetic operators are used to update the beliefs.

A. Coding of beliefs

At every moment in time t, there are two generations of N agents alive in the population. The first generation is the current "young" generation—agents in the first period of life, while the second generation is the current "old" population—agents in the second period of life. Each member of each generation may initially have a different belief about the parameter b. Their beliefs as to the true value of this parameter are encoded in a bit string of finite length ℓ. In the first period, $t = 1$, N ℓ–bit strings are chosen randomly for each generation. These bit strings are sufficient to completely characterize each agent's consumption and savings behavior as we shall now demonstrate.

Let the bit string for agent i be given by:

$$< a_{i1}(t), a_{i2}(t), ..., a_{i\ell}(t) > \quad \text{where } a_{ij}(t) \in \{0, 1\}$$

The agent's bit string can be decoded to a base 10 integer using the formula:

$$d_i(t) = \sum_{j=1}^{\ell} a_{ij}(t) \cdot 2^{\ell-j}$$

To calculate agent i's parameter estimate, $b_i(t)$, we take the value of $d_i(t)$ and divide it by the maximum possible decoded value: $d_{max} = \sum_{s=1}^{\ell} 2^{\ell-s}$. The result is a value in the interval $[0, 1]$. This fraction is then multiplied by the maximum gross inflation factor that the agent would need to forecast, consistent with equilibrium, which is given by the value of the parameter λ. Hence, each agent's value for $b_i(t)$ is determined according to the formula:

$$b_i(t) = \frac{d_i(t)}{d_{max}} \cdot \lambda.$$

Once a value for $b_i(t)$ is determined, the agent uses this value to forecast next period's price $P(t+1)$. With this forecast the model is closed and the agent is able to solve the maximization problem. The algorithm that we developed for this paper actually solves this constrained maximization problem for each agent, given the agent's parameter estimate for b. Thus agents have no difficulty in our framework in solving a constrained maximization problem. They are only uncertain as to the correct value of the parameter b. This uncertainty can be viewed as arising naturally if we think of agents as initially uncertain about the beliefs of the other agents. Initial uncertainty of this type might come about because, even if all agents understand well the nature of their situation, they are not sure what to believe since there are multiple beliefs which are consistent with equilibrium; which of these is correct depends on the beliefs of all of the other agents.

B. Genetic updating of beliefs

Agents of generation t form forecasts of future prices only in period t, when they are members of the "young" generation. The *actual* inflation factor between dates t and $t+1$ is determined according to equation (4), and will not be revealed to members of generation t until these agents are in the second period of their lives, that is, when they are members of the "old" generation. Thus, the success or failure of a particular forecast cannot be immediately ascertained.

The genetic updating of beliefs proceeds as follows. The first step is to calculate aggregate savings by the young generation born at time t. This is done by solving each young agent's maximization problem, conditional on that agent's belief, and obtaining an individual savings amount $s_{it}(t)$. Aggregate savings is then given by:

$$S(t) = \sum_{i=1}^{N} s_{it}(t)$$

Using this value for aggregate savings in equation (2), and using equation (3) to substitute out for real money balances, we have that the new, realized inflation factor $\beta(t-1)$ is given by:

$$\beta(t-1) = \frac{P(t)}{P(t-1)} = \frac{S(t-1)}{S(t) - Ng}.$$

The value of $\beta(t-1)$ depends on aggregate savings at time t and at time $t-1$, as well as on the value of per capita government purchases, g. Once $\beta(t-1)$ is known, it is possible to evaluate the forecasts made by generation $t-1$. Alternatively, one can now calculate the *actual* lifetime utility achieved by each member of generation $t-1$. These lifetime utility values will be used in the first step of the genetic algorithm.

The genetic algorithm is used to model how the next generation's beliefs evolve. The first step in the genetic algorithm is *reproduction.* Here we use a simple tournament selection method. Two members of the old generation are selected at random and their lifetime utility values are compared. Comparison of lifetime utility values is equivalent to assessing how close each of these two agents came to correctly forecasting actual inflation, since the two agent's forecasts were used to solve the same utility maximization problem. The old agent with the highest lifetime utility value (closest forecast) is copied and placed in the population of "newborn" agents. This tournament selection process is repeated N times, where N is the (constant) size of each generation. We stress that it is forecasts that are being copied. These forecasts have been shown to be relatively more successful than other forecasts made by members of generation $t-1$.

The next steps in the genetic algorithm are the crossover and mutation operators. In addition to these two standard operators, we have added an elitist selection operator that we will refer to as the *election operator* following Arifovic (1992). We view all three of these operators as describing a process by which the "newborn" generation (the product of the reproduction operator) experiments with "alternative forecasts" before deciding upon the forecast they will actually use when born into next period's young generation.

These "alternative forecasts" are created through the crossover and mutation operators. *Crossover* is applied to all strings in the newborn population with probability p^c. Strings are paired randomly. We use *single–point crossover*, choosing a point at random in both bit strings, cutting the strings at that point, and swapping all bits to the right of that point. *Mutation* is applied to each bit in each newborn string with probability p^m: bit $a_{ij}(t)$ is changed to equal $1 - a_{ij}(t)$ with probability p^m.

Once crossover and mutation has been performed on the N newborn strings, the newborns must decide whether they want to adopt any of these alternative forecasts as their own. Therefore, the alternative forecasts are evaluated as to how they would have performed. These forecasts are decoded to obtain an inflation forecast, and the utility maximization problem is then solved, given this forecast. Utility is then evaluated using the most recent actual inflation rate $\beta(t-1)$, and a lifetime (expected) utility value is calculated for each alternative forecast.

Once the lifetime expected utility associated with the alternative forecast is calculated, the *election* operator determines how newborn agents choose between the string they have inherited and the alternative string they have "created." Newborn agents are matched pairwise with their associated alternatives. The election operator then chooses the two forecasts that yielded the highest lifetime utility from among the two newborns and the two alternatives. The two "winners" become members of the newborn generation; the "losers" are discarded. The election operator is applied $N/2$ times so as to obtain a newborn generation of N agents.[5]

Once the strings of the newborn generation have been chosen, time changes to the next period, $t+1$, and the population of agents is aged appropriately. Agents who were born at time $t-1$, and who were members of the old generation at time t, cease to exist. Agents who were born at time t and who were members of the young generation at time t now become members of the old generation. The newborn generation is the new young generation "born" at time t. The process described in this section is then repeated again, with a new value calculated for aggregate savings, $S(t+1)$.

As a matter of interpretation, we stress that we do not need to think of the model as sets of agents actually passing genetic information via a biological process. Instead, we might view new agents coming into the model as new entrants to the workforce. They communicate with other agents concerning possible forecasts for future inflation, and take actions based on the forecast they adopt. Thus, agents can be viewed as exchanging ideas about the best way to forecast the future. The reproduction operator ensures that the better ideas from the older generation are adopted by the younger generation. The crossover and mutation operators allow the agents to experiment with alternative forecasts. And the election operator ensures that agents are not forced to adopt any "bad ideas."

Our genetic algorithm learning system generates a sequence of gross inflation rates, a sequence of N-string generations, and a sequence of sets of N forecast errors. We allow the system to evolve until the following convergence criteria are met. First, we required that inflation is at a steady state level predicted by the model under perfect foresight; second,

[5]The election operator is properly viewed as an elitist selection operator. Some type of elitist selection is necessary to ensure that the genetic algorithm converges asymptotically to the global optimum. See Rudolph (1994).

all strings within the most recent generation must be identical; and third, the most recent two sets of forecast errors must all be equal to zero up to a predefined tolerance. If these criteria were not met after 1,000 iterations, the process was terminated.

V. Parameterization and Results

We now discuss the parameterizations and results of a number of simulations that we conducted using our genetic algorithm model of the evolution of beliefs. We begin with parameter values relating to the genetic algorithm aspect of the model. In all of our simulations, we chose to set a high rate of crossover, $p^c = 1$, and a relatively low rate of mutation, $p^m = .033$. The high probability of crossover is possible because of the election operator: if agents are allowed to discard 'bad ideas,' there is no harm in experimenting extensively. We chose to consider populations of two different sizes, $N = 30$ and $N = 60$. These parameter values all fall within the ranges recommended in the genetic algorithm literature.[6] In addition, we chose two different values for the length of the agent's bit string: $\ell = 4$, and $\ell = 8$. When $\ell = 4$, agents choose from among $2^4 - 1$ or 15 different parameter values for b. When $\ell = 8$, a similar calculation reveals that agents choose from among 255 different parameter values for b.

We also had to chose values for a number of parameters relating to the overlapping generations economy. We chose to use the same endowment amounts in all simulations: $\omega_1 = 4$ and $\omega_2 = 1$. We considered two different values for per capita government purchases, $g = .3333$, and $g = .45$. The principle advantage to considering two different levels for g is that the two steady state equilibria are moved closer together as g increases. In particular, when $g = .3333$, the two stationary values for inflation are $\beta^L = 1.3333$ and $\beta^H = 3$. When g is increased to .45, these two values change to $\beta^L = 1.6$ and $\beta^H = 2.5$.

Our main result is that, in all of the computational experiments that we conducted, the

[6]See, for instance, Grefenstette (1986) or Goldberg (1989).

algorithm converged upon the low inflation stationary equilibrium, β^L. We now report the results of a few experiments that we performed to determine the role played by the different parameter values discussed above.

A. Experiment 1

In this experiment, we fixed the size of each generation at $N = 30$, and varied the length of agents' bit strings between $\ell = 4$ and $\ell = 8$. When $\ell = 4$, the population of 30 agents considers just 15 different values for b, so the ratio of different possible beliefs to agents is .5. When $\ell = 8$, the population of 30 agents considers 255 different values for b and the ratio of different possible beliefs to agents is 17 times higher, at 8.5. This experiment is intended to determine whether the degree of heterogeneity is a factor in the speed with which the algorithm converges to the low stationary inflation value. The results are reported in the first column of Table 1, which presents the mean and standard deviation of the number of iterations to convergence from 100 computational experiments for each parameterization. As the table reveals, increasing the heterogeneity of beliefs by lengthening the bit string from 4 to 8 led to an increase in the mean number of iterations it took the algorithm to converge, as well as an increase in the standard deviation. We conclude that the increase in the number of inflation forecasts that agents might consider made it more difficult for these agents to coordinate on a single forecast corresponding to β^L. Figure 2 depicts the inflation forecasts of the 30 agents at each iteration from a typical computational experiment in the case where $\ell = 8$. As the figure reveals, the agents quickly coordinate in a neighborhood of the low inflation stationary equilibrium, $\beta^L = 1.333$; however it takes agents a long time to actually reach consensus on the same inflation forecast value.

Table 1: Convergence Results for Different GA Parameterizations

Length of bit string	Number of $b_i(t)$ values	$N = 30$ Mean	Std. Dev.	$N = 60$ Mean	Std. Dev.
4	15	11.24	3.47	10.43	1.46
8	255	50.49	54.96	22.39	7.84

Table 2: Convergence Results for Different Values of g

Length of bit string	Number of $b_i(t)$ values	$g = .333$ Mean	Std. Dev.	$g = .45$ Mean	Std. Dev.
4	15	11.24	3.47	13.19	7.90

B. Experiment 2

In a second experiment, we repeated Experiment 1, but increased the size of each generation from $N = 30$ to $N = 60$. The results are reported in the second column of Table 1. When N is increased to 60, the ratio of different possible forecasts to agents *decreases*, and so it takes agents less time to find good forecasts—*sampling* by the population has increased. Evidently, when $\ell = 4$ and there are only 15 inflation forecasts, the increase in the population size does not make much difference. However, when there are more possible forecasts than agents, as when $\ell = 8$, an increase in the population size leads to a considerable reduction in the mean number of iterations to convergence.

C. Experiment 3

In a final experiment, we returned to the situation where $\ell = 4$ and $N = 30$ and examined the effect of increasing the size of government expenditures from $g = .33$ to $g = .45$. This increase in g moves the two stationary equilibria closer together. The hypothesis we sought to test was whether the algorithm would have greater difficulty coordinating on the low inflation stationary equilibrium when it was closer to the high inflation stationary equilibrium. The mean number of iterations to convergence from 100 computational experiments for each value of g is reported in Table 2, which repeats some information found in Table 1. The increase in g does lead to an increase in the mean number of iterations to convergence as well as in the standard deviation, indicating that coordination is made more difficult when equilibria are closer together.

VI. Summary

Economists have only recently begun to apply genetic algorithms to economic problems. In this paper we have provided a simple illustration of an alternative implementation of the genetic algorithm in an overlapping generations economy. In typical applications, agents are viewed as *learning how to optimize*, while in our alternative implementation, agents are viewed as *learning how to forecast.* The agents in our implementation optimize given their beliefs, so that the researcher relaxes standard economic assumptions along only one dimension, proceeding from homogeneous to heterogeneous beliefs. Our implementation may be viewed as especially useful for economists who wish to study problems of coordination of beliefs.[7]

Our experimental findings are mainly illustrative. We found that agents can indeed coordinate beliefs and learn the Pareto superior equilibrium of an overlapping generations model. Our results are consistent with the much more extensive results of Arifovic (1992), who used a *learning how to optimize* implementation of the genetic algorithm. However, our *learning how to forecast* implementation of the genetic algorithm converges much faster to the low stationary equilibrium value than does the *learning how to optimize* implementation studied by Arifovic. The faster convergence obtained in our implementation may be due to the fact that agents in our model optimize at every date, and only have to learn to coordinate their initially heterogeneous beliefs. The faster convergence that we found in our computational experiments is in fact consistent with results found in a series of two–

[7]For examples of genetic algorithm learning in other types of coordination problems, see Arifovic and Eaton (1994) and Bullard and Duffy (1994).

period overlapping generations experiments with human subjects conducted by Marimon and Sunder (1994). These authors report that learning to make good forecasts "seems to come faster" to their human subjects than does learning to solve a maximization problem.[8] We also found that coordination was more difficult when the number of inflation values considered by agents was higher, and when the two stationary equilibria of the model were closer together.

[8] Marimon and Sunder (1994), p. 143.

References

Arifovic, J. (1992), "Genetic Algorithms in Overlapping Generations Economies," mimeo, Simon Fraser University.

Arifovic, J. (1994), "The Behavior of the Exchange Rate in Genetic Algorithm and Experimental Economies," mimeo, Simon Fraser University.

Arifovic, J. and C. Eaton, (1994), "Coordination Via Genetic Learning," mimeo, Simon Fraser University.

Bullard, J. and J. Duffy (1993), "A Model of Learning and Emulation with Artificial Adaptive Agents," mimeo, Federal Reserve Bank of St. Louis and University of Pittsburgh.

Bullard, J. and J. Duffy (1994), "Learning and Pervasive Indeterminacy," mimeo, Federal Reserve Bank of St. Louis and University of Pittsburgh.

Grefenstette, J.J. (1986), "Optimization of Control Parameters for Genetic Algorithms," *IEEE Transactions on Systems, Man, and Cybernetics* 16, 122–28.

Goldberg, D.E. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning* Addison–Wesley, Menlo–Park, CA.

Marcet, A. and T.J. Sargent (1989), "Least Squares Learning and the Dynamics of Hyperinflation," in: W.A. Barnett, J. Geweke, and K. Shell, eds., *Economic Complexity: Chaos, Sunspots, Bubbles and Nonlinearity*, Cambridge University Press, Cambridge.

Marimon, R., and S. Sunder (1994), "Expectations and Learning Under Alternative Monetary Regimes: An Experimental Approach." *Economic Theory*, 4, 131–162.

Rudolph, G. (1994), "Convergence Analysis of Canonical Genetic Algorithms," *IEEE Transactions on Neural Networks* 5, 96–101.

Sargent, T.J. (1993), *Bounded Rationality in Macroeconomics*, Oxford University Press, Oxford.

Sargent T.J. and N. Wallace (1981), "Some Unpleasant Monetarist Arithmetic," Federal Reserve Bank of Minneapolis *Quarterly Review*, Fall.

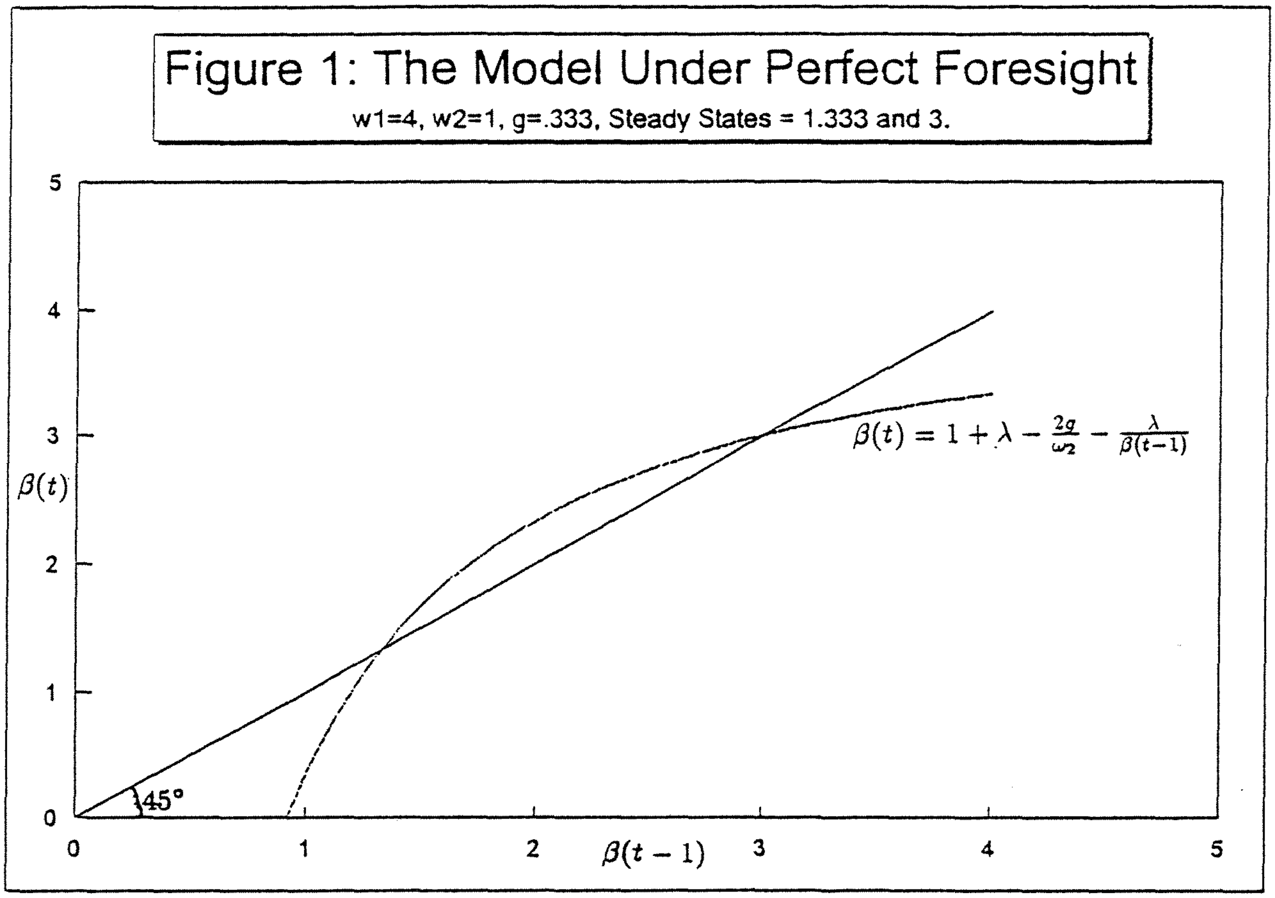
Figure 1: The Model Under Perfect Foresight
w1=4, w2=1, g=.333, Steady States = 1.333 and 3.
5
4
3
2
1
0
$\beta(t)$
$\beta(t) = 1 + \lambda - \frac{2g}{\omega_2} - \frac{\lambda}{\beta(t-1)}$
45°
0
1
2
3
4
5
$\beta(t-1)$

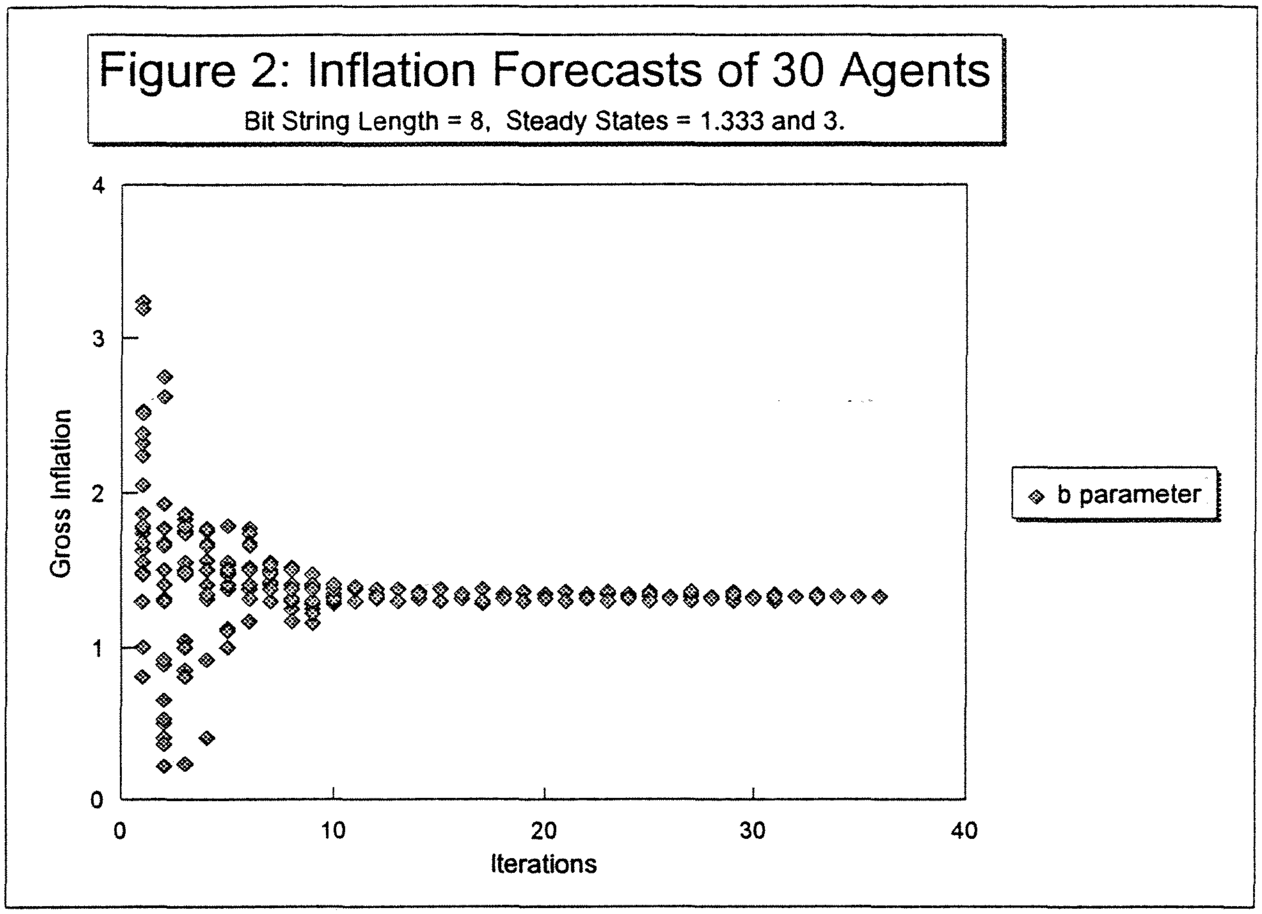

Figure 2: Inflation Forecasts of 30 Agents
Bit String Length = 8, Steady States = 1.333 and 3.
Gross Inflation
Iterations
b parameter
0
1
2
3
4
10
20
30
40