Computational Economics **13:** 41–60, 1999.

Using Genetic Algorithms to Model the Evolution of Heterogeneous Beliefs

JAMES BULLARD[1] and JOHN DUFFY[2]

[1]*Research Department, Federal Reserve Bank of St. Louis, P.O. Box 442, St. Louis, MO 63166, U.S.A. Tel.: (314) 444-8576; Fax: (314) 444-8731; E-mail: bullard@stls.frb.org*
[2]*Department of Economics, University of Pittsburgh, Pittsburgh, PA 15260, U.S.A. Tel.: (412) 648-1733; Fax: (412) 648-1793; E-mail: jduffy+@pitt.edu*

Abstract. We study a general equilibrium system where agents have heterogeneous beliefs concerning realizations of possible outcomes. The actual outcomes feed back into beliefs thus creating a complicated nonlinear system. Beliefs are updated via a genetic algorithm learning process which we interpret as representing communication among agents in the economy. We are able to illustrate a simple principle: genetic algorithms can be implemented so that they represent pure learning effects (i.e., beliefs updating based on realizations of endogenous variables in an environment with heterogeneous beliefs). Agents optimally solve their maximization problem at each date given their beliefs at each date. We report the results of a set of computational experiments in which we find that our population of artificial adaptive agents is usually able to coordinate their beliefs so as to achieve the Pareto superior rational expectations equilibrium of the model.

Key words: genetic algorithms, learning, equilibrium selection, heterogeneous beliefs

1. 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 researchers 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 study a general equilibrium system where agents have heterogeneous beliefs about the future values of endogenous variables. These beliefs affect actual outcomes, which in turn feed back into beliefs, creating a complicated nonlinear system. We use a genetic algorithm to update agents' beliefs. Our primary objective is to illustrate how the modelling of such a system can be implemented without compromising the standard economic assumption that agents optimize given their beliefs.

Such an illustration is interesting, in our view, because genetic algorithm learning can be implemented 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(s) based on which values worked well for other agents in the past. Most of the general equilibrium applications of genetic algorithms of which we are aware use this first method.[2] 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 have worked well in the past, and then solve a maximization problem to find the value of their choice variable given their forecast.[3] With this second method, the assumption that agents maximize utility is maintained. In this paper we provide an example of this 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 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.[4] 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 beliefs. 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 researchers 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 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 a 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 3 we describe the model under learning, and in Section 4 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 findings.

2. 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.

Each agent $i \in [1, N]$ born at time t solves the following problem:[5]

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

subject to:

$$c_t^i(t) + c_t^i(t+1)\beta^i(t) \leq \omega_1 + \omega_2 \beta^i(t),$$

where $c_t^i(t+j)$ denotes consumption in period $t+j$ by the agent i born at time t and $\beta^i(t)$ denotes agent i's time t forecast of the gross inflation factor between dates t and $t+1$:

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

where $P(t)$ denotes the time t price of the consumption good in terms of fiat money, and $F^i[P(t+1)]$ is agent i's 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 N agents have perfect foresight, in which case $F^i[P(t+1)] = P(t+1)$ for all i, so that $\beta^i(t) = \beta(t) = \frac{P(t+1)}{P(t)}$ for all i.

Combining the first order conditions with the budget constraint, one finds that the first period consumption decision for all N agents is given by:

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

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

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

Fiat money is introduced into this 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 a fixed, per capita amount g of the consumption good in every period:

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

It is assumed that these government purchases do not yield agents any additional utility. Note that while government purchases are exogenous, the evolution of the nominal money supply is determined endogenously depending on the realization of the price level $P(t)$ at each date t. The price level realization depends, in turn, on the forecast of the gross inflation factor. Thus, in this model the evolution of the nominal money supply depends in part on the beliefs of the agents.

Since agents can save only by holding fiat money, the money market clearing condition is that aggregate savings equals the aggregate stock of real money balances at every date t:

$$\sum_{i=1}^{N} s_t^i(t) = N\frac{M(t)}{P(t)}. \tag{3}$$

The explicit introduction of unsecured debt – in our case, the fiat money printed by the government – serves to ensure that Walras' Law holds for this economy.[6] Therefore, 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, one obtains the following 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

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

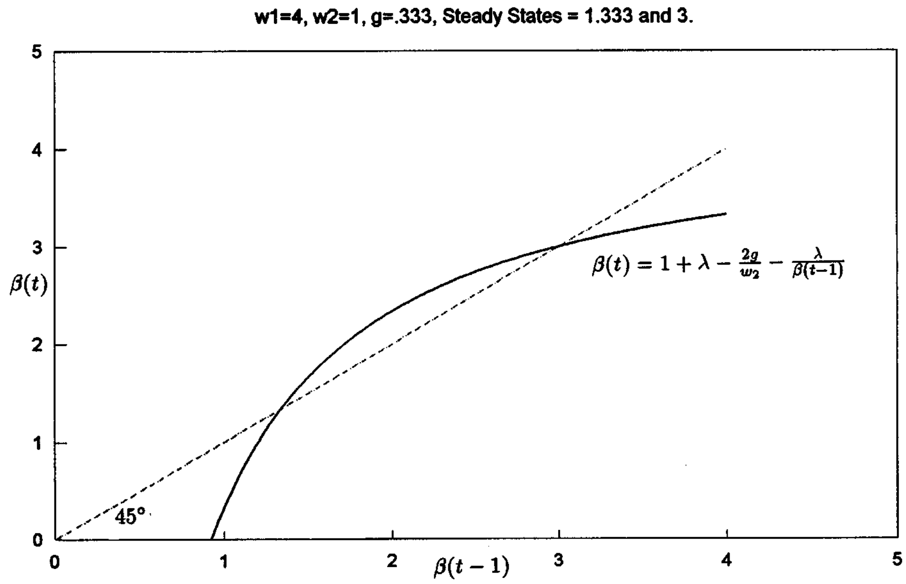

Figure 1. The model under perfect foresight.

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

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

Given condition (5) and the restrictions on endowments that imply that $\lambda > 1$, Equation (1) implies that both stationary solutions, β^H and β^L, involve positive savings amounts, so that fiat money is positively valued at both stationary equilibria. Condition (5) can then be interpreted as a restriction on the government's ability to finance all of its purchases through the printing of money while maintaining a positive valued fiat currency.[7] 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 $\beta(0) \in (\beta^L, \lambda)$.[8]

The two stationary equilibria are illustrated in Figure 1, which depicts the qualitative graph of Equation (4) for a particular case that will be studied later in the paper. In the case illustrated $g = 0.333$, $\beta^L = 1.333$ and $\beta^H = 3.0$. As government expenditures per capita, g, increase, the curve representing Equation (4) shifts downward and the two stationary equilibrium values, β^L and β^H, move closer together.

3. 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. Instead we assume that all N agents who are in the first period of life at time t forecast future prices using the simple 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 model, each agent may have a *different belief* regarding the appropriate value of the unknown parameter b. We further restrict agent's beliefs regarding the parameter b to fall in the interval:

$$0 \leq b^i(t) \leq \lambda, \quad \forall\, i,\ t.$$

The lower bound ensures that price forecasts are always nonnegative. The upper bound of λ represents the highest inflation factor that agents would need to forecast in order to achieve a feasible equilibrium. From Equation (1) we see that inflation forecasts in excess of λ imply that the agent's optimal savings decision is negative, that is, the agent would like to borrow from another agent in the 'consumption loan' market. However, for simplicity, we have chosen to rule out the possibility of borrowing by agents.[9] Thus inflation forecasts that are equal to or exceed λ will all result in the same consumption allocation, namely that agents save nothing and simply consume their endowments. Later in the paper, we consider an example economy where the domain for inflation forecasts is enlarged to include forecasts that may exceed the value of λ.

Each agent uses their individual forecast of future inflation 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 model (6) is consistent with the actual law of motion for prices *when agents have perfect foresight*. The consistency of the agent's forecast model 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 $b^1(t) = b^2(t) = \ldots = b^N(t) = \beta(t) = P(t+1)/P(t)$, that is, if all agents have identical forecast models, and their forecasts are always correct.

Of course, with this specification, forecast models differ only slightly across agents. One could easily design a more complicated example where the set of forecast models varied to a much greater degree.[10] Our intention in this paper, however, is simply to illustrate an alternative approach to genetic algorithm learning that maintains the assumption that agents optimize given their beliefs.

4. The Evolution of Beliefs

We now introduce the genetic algorithm which we use to determine the evolution of the parameters $b^i(t)$ over time. We first describe how forecast models are coded as binary strings and then we illustrate how the genetic operators of the genetic algorithm are used to update agents' beliefs.

4.1. 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 belief as to the true value of this parameter – their 'model' – is 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:

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

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: $d_{\text{max}} = \sum_{s=1}^{\ell} 2^{\ell-s}$. 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:

$$b^i(t) = \frac{d_i(t)}{d_{\text{max}}} \cdot \lambda.$$

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 may come about even if all agents understand well the nature of their situation. Since there are multiple beliefs that are consistent with equilibrium, the 'correct' belief at every date depends on the beliefs of all of the other agents.

4.2. 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 $t+1$ depends on the aggregate savings decision of the subsequent young generation $t+1$, 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 $s_t^i(t)$. Aggregate savings is then given by:

$$S(t) = \sum_{i=1}^{N} s_t^i(t).$$

Using this value for aggregate savings in Equation (3), and using Equation (2) 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.[11] 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* based on relative fitness (i.e. natural selection). Here we use a simple tournament selection method. Two members (bit strings) of the most recent old generation alive at time $t-1$ 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 forecast rules were used to solve the same utility maximization problem. The bit string of the old agent with the highest lifetime utility value (the closest forecast) is copied and placed in the population of 'newborn' agents. This tournament selection process is repeated N times so as to create a population of N newborn bit strings. We stress that it is forecast models that are being copied. These forecast models have been shown to be relatively more successful than other forecast models used by members of generation $t - 1$.

The next step in the genetic algorithm is the application of the crossover and mutation operators. In addition to these two standard genetic operators, we have augmented our genetic algorithm with an elitist selection operator that we will refer to as the *election operator* following Arifovic (1994). 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 forecast models' before deciding upon the forecast model they will actually use when they are 'born' into next period's young generation. The 'alternative forecast models' are created through the crossover and mutation operators.

The crossover operator is applied to all N strings in the newborn population. First, the N newborn strings are randomly paired. Then, for each pair of strings, the crossover operation is performed with some probability $p^c > 0$; with probability $1 - p^c$ crossover is not performed on the pair. If crossover is to be performed, the pair of newborn strings are cut at a randomly chosen integer point in $[1, \ell - 1]$. All bits to the right of the cut point are then swapped and the two strings are recombined. The result is two new strings that share bits of the genetic material that made up the original two newborn strings. Following application of the crossover operator the resulting strings are subjected to the mutation operator. Every bit in all N bitstrings is subject to being mutated. With probability $p^m > 0$ each bit, $a_{ij}(t)$, is changed to the value $1 - a_{ij}(t)$; with probability $1 - p^m$, the bit remains unchanged. The result of the crossover and mutation operators is a set of N alternative forecast models.

Following application of the crossover and mutation operators, the N newborns must decide whether they want to adopt any of the alternative forecast models as their own. In order to make this decision, the newborns consider how well the alternative forecast models *would have performed* had these models been used in the recent past. The alternative forecast models are first decoded and then used to obtain an inflation forecast. The utility maximization problem is then solved, given this forecast. Utility is evaluated using the most recent actual inflation rate $\beta(t - 1)$, and a lifetime (expected) utility value is calculated for each alternative forecast model. Once this process is complete, the *election* operator determines how newborn agents choose between the string (model) they have inherited and the alternative string (model) they have 'created'.

Pairs of newborn agents are matched with their associated alternatives. The election operator then chooses the two forecast models (out of four) that yielded the highest lifetime utility from among the two newborns and the two alternatives.

The two 'winners' become the forecast models used by the two members of the newborn generation; the 'losers' are discarded.[12] The election operator is applied $N/2$ times so as to obtain a newborn generation of N agents.[13]

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 $t + 1$. The process described in this section is then repeated again, beginning with the calculation of a new value for aggregate savings, $S(t + 1)$.

Our genetic algorithm learning system generates a sequence of gross inflation factors, 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 require that inflation is at a steady state level predicted by the model under perfect foresight; second, 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.

4.3. Remarks on Interpretation

We prefer to think of the agents in this economy as choosing a *forecast model*. This forecast model is then used to predict future prices and hence future gross inflation factors. Thus, in principle it is *different forecast models* that agents are experimenting with, not different beliefs about future inflation. However, in the simple application that we consider here, it turns out that the forecast model parameter value $b^i(t)$ that agents are learning about is equivalent to their individual forecast of gross inflation. As we have previously noted, we chose this forecast model specification in order to keep our illustration simple. One can easily imagine a different environment where agents considered a more complicated set of forecast models with more than one parameter value, and in such cases, there would no longer be a one-to-one mapping from parameter values to forecast values.[14]

We note that the election operator implies that newborn agents are capable of assessing the relative performance of different forecast models. Given this ability, one might wonder why all newborn agents don't simply choose *the* forecast model that yielded the highest lifetime fitness value in the most recent past. In our example economy this would amount to all newborn agents setting the parameter b equal to last period's gross inflation factor β, the standard against which all forecast models are assessed. One reason that agents might not behave in this manner is that the economy is not initially in a steady state (and there is no guarantee that it will necessarily ever achieve a steady state). Prior to the achievement of a steady state, the actual inflation factor will not remain constant but will instead vary from one period to the next. If agents recognize the time-varying nature of the inflation factor during the transition to a steady state then they may rationally choose to use forecast models that differ from those that worked best in the previous period. Thus, during the transition to a steady state it may not make sense for agents to simply set the parameter b equal to the previous period's realized inflation factor, β, even though the previous β is used by newborn agents to assess the lifetime utility they might expect to obtain from each forecast model.[15]

We also stress that we do not need to think of the model as sets of agents actually passing along 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 forecast models for future inflation, and take actions based on the forecast model 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. The election operator ensures that agents are not forced to adopt any 'bad ideas'.

5. Parameterization and Results

Our results are intended to illustrate our *learning how to forecast* implementation of genetic algorithm learning, and should be regarded as suggestive rather than an exhaustive study of this interpretation of genetic algorithm learning. We begin with our choice of parameter values for 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, $p^m = 0.033$. 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.[16] 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, $g = 0.333$, and $g = 0.45$. 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 $g = 0.333$, the two stationary values for inflation are $\beta^L = 1.333$ and $\beta^H = 3.0$. When g is increased to 0.45, these two values change to $\beta^L = 1.6$ and $\beta^H = 2.5$.

5.1. Main Findings and Interpretation

Our main result is that, in almost all of the computational experiments that we conducted, the algorithm satisfied our criterion for convergence to the low inflation stationary equilibrium, β^L, of the model within the allotted 1,000 iterations. In some replications of the last experiment reported below, convergence failed to obtain within 1,000 iterations.

The genetic algorithm's selection of the low inflation equilibrium stands in contrast to the stability properties of the model under the perfect foresight assumption. Recall from our earlier discussion that under perfect foresight, it is the *high* inflation stationary equilibrium, β^H, that is the attractor for all initial values of inflation in the interval (β^L, λ).

However, the genetic algorithm's selection of the *low* inflation stationary equilibrium is in accord with the predictions of a number of studies that replace the perfect foresight assumption in the overlapping generations economy with some kind of adaptive expectations scheme. Lucas (1986), for example, showed that if agents forecast future prices using a simple past average of prices, the economy would be locally convergent to the low inflation stationary equilibrium.[17] Marcet and Sargent (1989) obtained a local stability result for the low inflation stationary equilibrium when agents forecast future prices using a least squares autoregression on past prices, but only for situations where the level of the government's real deficit, g, was low enough. Bullard (1994) analyzed, in a closely related model, the bifurcation involved in moving from a money growth rate that was too low to one that was too high under the Marcet and Sargent learning scheme. The picture that emerges from these studies is that stability of the low inflation steady state of this model under the adaptive learning schemes considered is at best local, and that for some parameter configurations even local stability fails to obtain.

Arifovic (1995) studied genetic algorithm learning in the Marcet–Sargent model using a *learning how to optimize* implementation. She also found that the genetic algorithm system she studied converged to the low inflation steady state even in cases where least squares learning failed to converge.

While our computational experiments are only suggestive, we find that again, the low inflation stationary equilibrium seems to be much more of an attractor under our genetic algorithm learning scheme than it is under the perfect foresight assumption. As in Arifovic (1995), the genetic algorithm learning approach has a much more global flavor as compared with least squares learning, since the strings representing agent's forecast models in the genetic algorithm are initially randomly generated and thus the economy may start very far away from equilibrium.

The explanation for the convergence results we obtain under genetic algorithm learning differs from the explanations for convergence offered by Lucas and Marcet and Sargent. Both Lucas and Marcet and Sargent showed that under their respective adaptive learning schemes, the dynamics of the model environment were reversed, so that the low inflation stationary equilibrium became the attractor, and the high inflation stationary equilibrium became unstable. The explanation for the convergence of the genetic algorithm learning model to the low inflation stationary equilibrium would seem to be that this equilibrium provides agents with the highest lifetime utility (fitness) possible in this economy as it is the Pareto superior equilibrium of the model. The genetic learning algorithm conducts an extensive directed search of the parameter space; the aim of this search is to find this global optimum. Thus, one interpretation is that when convergence is obtained, it is because the genetic algorithm has located the global optimum, the object of its search.

This explanation for the convergence of the genetic algorithm to the low inflation steady state would be straightforward if agents were learning in a static environment with a unique and unchanging global optimum. However, as noted previously, agents are in a dynamic environment where their beliefs interact with outcomes, and outcomes interact with agents' beliefs, so that the landscape that is being searched may be constantly changing. In such an environment, the low steady state inflation factor will only yield the highest possible level of lifetime utility if all agents have coordinated on forecasting this level of inflation. Prior to such coordination, there may be other forecast rules that lead to higher levels of lifetime utility. Thus a question remains as to how the genetic algorithm is able to achieve coordination on the low inflation steady state in the dynamic environment that we consider.[18] In an effort to address this question, we have examined the evolution of lifetime utility, or *lifetime fitness* for a couple of different forecast rules in a number of our simulations. In particular, we have looked at the evolution of the fitness value that would be assigned to a forecast model that always forecast the low steady state inflation factor as well as the fitness value that would be assigned to a forecast model that always forecast the high steady state inflation factor.[19] With the exception of the first few initial periods, we always find that the fitness value of the low inflation steady state forecast is significantly greater than that of the high inflation steady state forecast. Therefore, a fitness distinction between these two stationary outcomes is nearly always present in the landscape that agents are searching. We believe that the presence of this distinction in steady state fitness levels is responsible for the convergence results that we are obtaining in most parameterizations of our model.

Figure 2 serves to illustrate this fitness distinction. This figure depicts the evolution of the hypothetical fitness value that would be attached to both the low and the high steady state inflation forecasts from one of our computational experiments where $N = 30$, $g = 0.333$ and $\ell = 8$. The figure also shows the evolution of the actual *average* fitness value from the population of 30 agents. This illustration is typical of other simulations we have conducted. We see that the fitness value associated with the low inflation steady state forecast is always higher than the fitness value associated with the high inflation steady state forecast. Notice that these fitness values vary over time due to the interaction of outcomes and beliefs. Note further that the *average* population fitness value in this illustration is initially intermediate to the low and high steady state inflation fitness values but

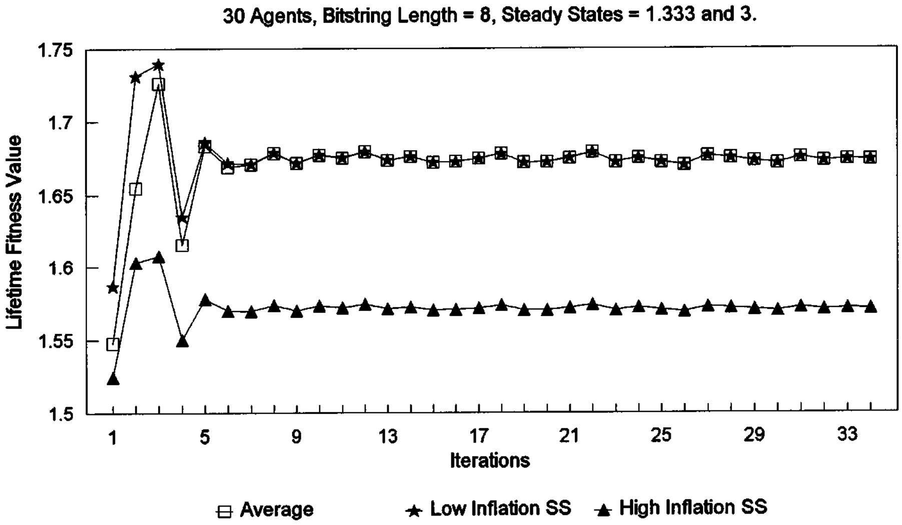

Figure 2. The evolution of fitness values.

very quickly moves toward the fitness level associated with the low inflation steady state forecast and follows this level very closely until the convergence criteria have been satisfied at the end of 34 iterations. We conclude from this exercise that there is typically a distinct advantage, in terms of lifetime fitness, from a forecast model that is consistent with the low inflation steady state, and that this advantage may well explain the convergence results that we are obtaining.

We now turn to a discussion of some of the more specific results of the experiments that we performed to determine the role played by the different parameter values of the model.

5.2. EXPERIMENT 1

In our first experiment, we set $g = 0.333$, $N = 30$, and we considered two different values for the length of agents' bit strings: $\ell = 4$ 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 0.5. 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 I, 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

Table I. Convergence results for different GA parameterizations.

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

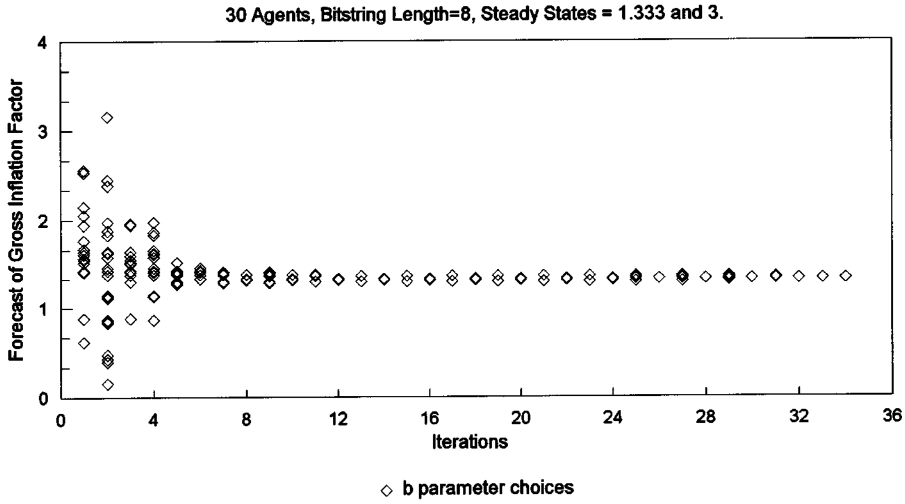

Figure 3. The evolution of inflation forecasts.

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 3 depicts the inflation forecasts of 30 agents at each iteration from one of the simulations conducted in Experiment 1 where $\ell = 8$ (the same experiment illustrated in Figure 2). We see that agents very quickly coordinate on a neighborhood of the low inflation stationary equilibrium, $\beta^L = 1.333$ within about 10 iterations; however it takes agents a total of 34 iterations to actually reach consensus on the same inflation forecast value.

5.3. Experiment 2

In another 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 I1. 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

Table II. Convergence results for different values of g.

Length of bit string	Number of $b^i(t)$ values	$g = 0.333$ Mean	Std. dev.	$g = 0.45$ Mean	Std. dev.
4	15	11.24	3.47	13.19	7.90

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.

5.4. EXPERIMENT 3

In a third experiment we once again set $\ell = 4$ and $N = 30$ and examined the effect of increasing the size of government expenditures from $g = 0.333$ to $g = 0.45$. 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 II, which repeats some information found in Table I. 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.

5.5. EXPERIMENT 4

In the final experiment that we report we set $g = 0.333$, $N = 30$ and $\ell = 8$, and we considered whether our convergence results were robust to an increase in the maximum inflation forecast that agents could make. Recall that the domain of possible inflation forecasts in all previous experiments was the interval from 0 to $\lambda = \frac{\omega_1}{\omega_2} = 4$. Note that this interval contains both of the stationary inflation values in all of the experiments we considered. When young agents forecast inflation factors above λ, their optimal consumption decision is to consume more than their endowment in the first period, through borrowing. Consequently their savings is negative. Since consumption loans from old agents to young agents are not possible, and since we do not allow young agents to lend or borrow among themselves, inflation forecasts above λ would simply result in the agent consuming his endowment in both periods and saving nothing. Thus, inflation forecasts above λ have the same effect on aggregate savings as an inflation forecast equal to λ. Nevertheless, increasing the maximum forecast above λ implies that more agents will initially choose to save zero, and this could affect our convergence results.

Table III. Convergence results for different maximum inflation forecasts.

Length of bit string	Number of $b^i(t)$ values	Max forecast $= \lambda$		Max forecast $= \lambda + 1$	
		Mean	Std. dev.	Mean	Std. dev.
8	255	50.49	54.96	67.60	85.90

The experiment we considered was increasing the maximum inflation forecast from $\lambda = 4$ to $\lambda + 1 = 5$, (while maintaining the same endowment sequence, $\omega_1 = 4$ and $\omega_2 = 1$). With $\ell = 8$, the number of possible inflation forecasts remains fixed at 255. However, the number of inflation forecasts that imply a zero savings decision has increased substantially. When the upper bound on inflation forecasts is equal to $\lambda = 4$, only 1 out of 255 possible inflation forecasts will imply a zero savings decision, but when the maximum inflation forecast is 5, there are 51 out of 255 inflation forecasts or 20% of all possible forecasts that will imply a zero savings decision. The mean number of iterations to convergence from 100 computational experiments in which the maximum inflation forecasts are λ and $\lambda + 1$ are reported in Table 3, which repeats some information from Table I.

In Table III, the mean and standard deviation of the number of iterations to convergence in the final column are based on those simulations where convergence was obtained. In 17 out of 100 replications for the case where the maximum forecast was $\lambda + 1$, the system failed to meet our convergence criteria within the allotted 1,000 iterations. Nevertheless, it is quite possible that the algorithm would eventually have satisfied the convergence criterion if it were allowed to continue.[20] The weight of the evidence, then, is that it does take longer for the system to converge in the case where the maximum forecast is $\lambda + 1$ as opposed to the case where the maximum forecast is λ. We conclude from this exercise that researchers will have to give some consideration to the set of possible forecast rules they allow agents to choose from.

6. 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.[21]

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. We have offered a possible explanation for this result. Our results are consistent with the much more extensive results of Arifovic (1995), who used a *learning how to optimize* implementation of the genetic algorithm. Our initial impression is that the *learning how to forecast* version of genetic algorithm learning converges faster than the *learning how to optimize* implementation studied by Arifovic (1995). To the extent this result holds up under further computational experimentation, it would be consistent with results found in a series of two-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.[22] We also found that coordination was more difficult when the number of inflation values considered by agents was higher, when the two stationary equilibria of the model were closer together, and when agents entertained inflation rate forecasts outside the bounds of possible stationary equilibria.

Acknowledgements

We thank Chris Birchenhall and three anonymous referees for helpful comments. Any views expressed are those of the authors and do not necessarily reflect the views of the Federal Reserve Bank of St. Louis or the Federal Reserve System.

Notes

1. See, for example, Arifovic (1995, 1996), Arifovic, Bullard and Duffy (1997), Bullard and Duffy (1998a, b), Routledge (1995) and Sargent (1993). For some other economic applications of genetic algorithms see the special issue of *Computational Economics*, Vol. 8, No. 3 (1995), edited by Chris Birchenhall. Goldberg (1989) and Mitchell (1996) provide excellent introductions to the use of genetic algorithms.
2. Bullard and Duffy (1998a, b) are an exception.
3. Marimon and Sunder (1994) view the distinction between *learning how to optimize* and *learning how to forecast* as a key experimental design challenge in the context of setting up overlapping generations experiments with human subjects.
4. 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.
5. The choice of logarithmic preferences implies that consumption in both periods of life are gross substitutes. This choice of preferences rules out the possibility that the limiting perfect foresight dynamics are periodic or chaotic. For an analysis of genetic learning in a model where consumption in the two periods of life are non-gross substitutes, see Bullard and Duffy (1998b).
6. This condition for Walras' Law to hold is discussed in Pingle and Tesfatsion (1994). More generally, as Wilson (1981) and others have shown, Walras' Law may fail to hold in infinite horizon overlapping generations economies.
7. The government's purchase of g units per capita of the consumption good at every date t is feasible since each agent alive at date t is endowed with some amount ω_1 or ω_2 of the consumption good and both of these amounts exceed g as can be seen from condition (5).
8. For an analysis of the dynamics under a least squares learning scheme see Marcet and Sargent (1989) and Bullard (1994).

9. When all agents have the same endowments, preferences and beliefs, a consumption loan market involving borrowing and lending among agents of the same generation cannot exist. However, when agents are heterogeneous in some respect, e.g. when they have heterogeneous beliefs as we assume here, then an active consumption loan market becomes possible. The implementation of a consumption loan market in an economy where agents have heterogeneous beliefs is a challenging task which we leave to future research.
10. See e.g. Bullard and Duffy (1998a).
11. Note that we must also have $(S(t)-Ng) > 0$ to ensure that the inflation factor is positive. Given the restrictions on g, and assuming perfect foresight, this condition will always be true. However, under a learning assumption, such as GA learning, this condition may be violated. The algorithm that we developed checks at each iteration to ensure that the condition $(S(t) - Ng) > 0$ is satisfied. If it is not, the algorithm is reinitialized and the simulation is begun anew.
12. Thus, in contrast to Arifovic (1994), it is the forecast models of agents that are discarded, rather than the agents themselves.
13. 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).
14. See Bullard and Duffy (1998a) for an example of such an environment.
15. At issue is the following trade-off: while it is important to have a universal fitness criterion so as to apply some selection pressure, it is also important to maintain some heterogeneity in the population of candidate forecast models so as to ensure a good global search.
16. See, for instance, Grefenstette (1986) or Goldberg (1989).
17. The Lucas (1986) example would correspond to ours if $g = 0$. When $g \neq 0$, using the average of past prices as a learning rule will never suffice since the equilibrium price sequence would be nonstationary.
18. Arifovic and Eaton (1995) have an application of genetic algorithm learning in a dynamic environment in which, under certain parameterizations, the genetic algorithm fails to find a Pareto dominant equilibrium, converging instead to a Pareto inferior equilibrium. Thus, there is no guarantee that a genetic algorithm will always find the global optimum.
19. Note that these are hypothetical fitness values, associated with unchanging, steady state forecast rules that are not necessarily present in the population of decision rules. If each of these rules were actually in use in the population then the observed outcomes would be slightly altered.
20. In all instances of non-convergence the algorithm was observed to be rather close to the low inflation stationary equilibrium. We speculate that convergence to the low stationary equilibrium would have occurred if a mutation or two in a particular bit value had occurred and the correctly mutated string had been randomly selected by the reproduction tournament.
21. For examples of genetic algorithm learning in other types of coordination problems, see Arifovic and Eaton (1995) and Bullard and Duffy (1998b).
22. Marimon and Sunder (1994), p. 143.

References

Arifovic, Jasmina (1994). Genetic algorithm learning and the cobweb model. *Journal of Economic Dynamics and Control*, **18**, 3–28.

Arifovic, Jasmina (1995). Genetic algorithms and inflationary economies. *Journal of Monetary Economics*, **36**, 219–243.

Arifovic, Jasmina (1996). The behavior of the exchange rate in the genetic algorithm and experimental economies. *Journal of Political Economy*, **104**, 510–541.

Arifovic, Jasmina, Bullard, James and Duffy, John (1997). The transition from stagnation to growth: an adaptive learning approach. *Journal of Economic Growth*, **2**, 185–209.

Arifovic, Jasmina and Eaton, Curtis (1995). Coordination via genetic learning. *Computational Economics*, **8**, 181–203.

Bullard, James (1994). Learning equilibria. *Journal of Economic Theory*, **64**, 468–485.

Bullard, James and Duffy, John (1998a). A model of learning and emulation with artificial adaptive agents, forthcoming. *Journal of Economic Dynamics and Control*.

Bullard, James and Duffy, John (1998b). On learning and the stability of cycles, forthcoming. *Macroeconomic Dynamics*.

Grefenstette, John J. (1986). Optimization of control parameters for genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, **16**, 122–128.

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

Lucas, Robert E. Jr. (1986). Adaptive behavior and economic theory. *Journal of Business*, **59**, S401–S426.

Marcet, Albert and Sargent, Thomas J. (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, MA.

Marimon, Ramon and Sunder, Shyam (1994). Expectations and learning under alternative monetary regimes: an experimental approach. *Economic Theory*, **4**, 131–162.

Mitchell, Melanie (1996). *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA.

Pingle, Mark and Tesfatsion, Leigh (1994). Walras' law, pareto efficiency and intermediation in overlapping generations economies. Economic Report No. 34, Iowa State University.

Routledge, Bryan R. (1995). Adaptive learning in financial markets. Working paper. Carnegie Mellon University.

Rudolph, Günter (1994). Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Networks*, **5**, 96–101.

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

Wilson, Charles A. (1981). Equilibrium in dynamic models with an infinity of agents. *Journal of Economic Theory*, **24**, 95–111.