Genetic Programming Learning and the Cobweb Model[*]

Shu-Heng Chen
AI-ECON Research Group
Department of Economics
National Chengchi University
Taipei, Taiwan 11623
E-mail: chchen@cc.nccu.edu.tw

Chia-Hsuan Yeh
AI-ECON Research Group
Department of Economics
National Chengchi University
Taipei, Taiwan 11623
E-mail: g3258501@grad.cc.nccu.edu.tw

Abstract

Using genetic programming to model the cobweb model as a multiagent system, this chapter generalizes the work done by Arifovic (1994), which is based on genetic algorithms. We find that the rational expectations equilibrium price which can be discovered by genetic algorithms can also be discovered by genetic programming. Furthermore, genetic programming requires much less prior knowledge than genetic algorithms. The reasonable upper limit of the price and the characteristic of the equilibrium which is assumed as the prior knowledge in genetic algorithms can all be discovered by genetic programming. In addition, GP-based markets have a self-stabilizing force which is capable of bringing any deviations caused by mutation back to the rational expectations equilibrium price. All of these features show that genetic programming can be a very useful tool for economists to model learning and adaptation in multiagent systems. In particular, with respect to the understanding of the dynamics of the market process, it provides us with a visible foundation for the "invisible hand".

1 Introduction

A current topic of interest in macroeconomics is to model macroeconomic dynamics as a multiagent system in lieu of a representative agent system. The advantage of this paradigm is that it enables us to observe how the coordination of activities in an economy can be well/ill developed such that resource allocation can be/fails to be efficient. Due to a lack of appropriate mathematical models for learning agents, the advancement of understanding about market failures had been limited until recently when multidisciplinary approaches to economics, especially the ever-increasing influence of distributed artificial intelligence or parallel distributed processing on economics, provided economists with a great chance to make breakthroughs.

---

[*] This paper is published in *Advances in Genetic Programming 2*, in P. Angeline and K. E. Kinnear, Jr. (eds.), Chapter 22, Cambridge, MA: MIT Press.

One recent breakthrough is exemplified by Arifovic (1994). Arifovic (1994) employed the recent development of evolutionary computation to model learning of economic agents in the cobweb model. In this model consumers base their decisions on the current market price, but producers decide how much to produce based on the past prices. Agricultural commodities serve as a good example of the cobweb model. The lagged response to price by firms might cause the price to be unstable or cyclic. The conditions for the instability were studied by Ezekiel (1938) and well documented in the cobweb theorem, which states that given producers expect that prices will remain constant at their previous level, the market price will converge to its equilibrium value if the slope of the demand curve (price as a function of quantity demanded) is smaller in absolute value than the slope of the supply curve (price as a function of quantity supplied). This is referred to as the stable case of the cobweb model. If the ratio is greater than one, then the market diverges from the equilibrium, which is referred to as the unstable case. This theorem, however, was built on the single-agent or representative-agent system and is contradicted by the recent results based on the experiments conducted by Wellford (1989), who found that the pattern of divergent prices predicted by the traditional cobweb theorem for the unstable case was not observed in his experiments with human subjects. Wellford's finding is impressive because it confirms a basic belief of many evolutionary economists. As Clower (1976) stated:

> ...unstable economic systems, although theoretically conceivable, are not practically viable. Systems that survive over long intervals of calendar time effectively declare themselves to be 'of stable type'.(p.225)

If unstable systems are not viable in the real world, then what are the fundamental forces neglected in the traditional analysis of stability? The first answer might be that there are some sorts of adaptation or learning mechanism which has a self-stabilizing feature. Recent studies have shown, however, that simple learning algorithms such as the least squares learning still result in divergent behavior for the cobweb unstable case. See, for instance, Bray (1982) and Marcet and Sargent (1989). Therefore, to explain the human behavior observed in the laboratory, other types of learning algorithm are needed.

Using the genetic algorithm developed by Holland (1975) to model the adaptive multiagent system, Arifovic (1994) examined the convergence results obtained by Wellford (1989). She applied two versions of GAs to this issue. One is the version described in Goldberg (1989) with three genetic operators: reproduction, crossover and mutation, which she called the basic GA. Arifovic found that in each simulation of the basic GA, individual quantities and prices exhibited fluctuations for its entire duration and do not result in convergence to the rational expectations equilibrium values, which is quite inconsistent with the Wellford's experimental results with human subjects.

Arifovic's second GA version, the augmented GA, includes the election operator in addition to reproduction, crossover, and mutation. The election operator involves two steps. First, crossover is performed. Second, the potential fitness of the newly generated offspring are compared with the actual fitness values of its parents. Then, among the two offspring and two parents, the two highest fitness individuals are chosen. The purpose of this operator is to overcome difficulties related to the way mutation influences the convergence process because the election operator can bring the variance of the population rules to zero as the algorithm converges to the equilibrium values. The results of the simulations show that the augmented GA converges to the rational expectations equilibrium values for all sets of cobweb model parameter values, including both stable and unstable cases, and can capture several features of the experimental behavior of human subjects better than other simple learning algorithms. To avoid the arbitrariness of choice of an adaptive scheme, Lucas (1986) suggests that comparison of the behavior of adaptive schemes with behavior observed in laboratory experiments with human subjects can facilitate the choice of a particular adaptive scheme. By this suggestion, the GA could be considered an appropriate choice to model learning agents in a multiagent system.

In addition to the support from experimental economics, the genetic algorithm has been shown in machine learning as a very efficient search algorithm as opposed to the totally blind search, especially when the search space is very large and cannot be arbitrarily reduced. For example, Eiben et. al. (1990), Chakraborty and Dastidar (1993), and Rudolph (1994) showed that, under certain conditions, GAs evolve to the optimum with probability 1. These two considerations are extremely important because global search techniques are essential to modeling agents in a complex environment with limited information or without structural knowledge. Unfortunately, it has taken time for economists to become familiar with global search techniques which can easily be implemented and also possess a good sense of economics. GAs satisfy both of these criteria and should provide economists with great potential value.

Motivated by the original contribution of Arifovic (1994), this chapter attempts to generalize her modeling strategy by replacing the GA with genetic programming (GP) developed by Koza (1989). The significance of this generalization is three-fold. Firstly, while the prior knowledge required by GAs is less than that required by those traditional learning algorithms based on classical or Bayesian econometrics, GAs still require stronger prior knowledge than GP. This is primarily due to the difference in their representation of solutions. The representation problem for genetic algorithm has been well discussed in chapter 4 of Koza (1992). GAs usually express each solution as a fixed-length string. Hence the search space defined by GAs must be finite and the choice of the search space naturally will require some prior knowledge about the problem. Take Arifovic (1994) as an example. Her choice of the normalized factor $\overline{K}$ automatically sets an upper limit to firms' decision on how much to produce. However, in her model, there is no reason why such a limit should emerge. On the contrary, firms in her model are prepared to produce an enormous amount of goods as long as they expect the price to be exceptionally high. Hence, the upper limit, $\overline{K}$ must be considered prior knowledge. By doing this, we are dramatically restricting the possibility of price euphoria, i.e., the expectation that prices will stay up and go up, perhaps indefinitely. Thus, this setting makes it difficult to investigate under what conditions, the price mechanism is good enough to eliminate crashes caused by price euphoria. On the other hand, GP expresses each solution as a parse tree whose size can be controlled by user-defined parameters, and the search space it defines can, in principle, be very large. Hence, the prior knowledge required is very limited. By using GP, we will be able to observe under what circumstances the upper limit $\overline{K}$ can actually be learned by firms.

Secondly, most of the literature addressing learning in the context of the cobweb model considers what learning agents attempt to learn to be a regularity about the motion of the prices, which is a function. See, for instance, DeCanio (1979) and Bray (1982). However, due to the limitation of GA, what learning agents learn in the Arifovic's model is just a number corresponding to the equilibrium quantity, which does not make too much sense unless we are assuming that learning agents already come to the belief that the equilibrium of the system is characterized by a unique fixed point. Suppose the equilibrium is not a fixed point but a limit cycle or a strange attractor, then what the agents need to learn is not just a number but a functional relationship, such as $P_{t+1} = f(P_t, P_{t-1}, \cdots)$. Therefore, a more general strategy which requires fewer prior beliefs or knowledge is to model learning agents by selecting models from data rather than selecting numbers from data. In fact, considering the latter a special case of the former, we can ask: will learning agents come to the belief that there is a unique fixed point in the equilibrium when there is one? If they will, then we can further inquire whether learning agents can discover this fixed point. While all of these issues are difficult to be modeled with GAs, it is straightforward to address them through GP.

Thirdly, as Kinnear (1994) stated:

> One of the as yet unrealized strengths of genetic programming is that it is general enough to integrate many of the different techniques and approaches used in other styles of evolutionary computation into a more powerful whole.

By modeling learning agents with genetic programming, we are able to observe a richer class of agents' learning behavior, which includes many traditional learning algorithms such as the least squares learning. Thus, the failure of the least squares to capture the essential feature of human learning can be addressed in a more enlightening way by this general framework. For example, one may ask: how intelligent GP-based learning agents need to be so that a successful coordination of resource allocation can be expected.

Using genetic programming described in Koza (1992), this chapter compares the learning performance of GP-based learning agents with that of GA-based learning agents in the cobweb model. First of all, We find that, like GA-based learning agents, GP-based learning agents also can learn the equilibrium price under both the stable and unstable cobweb case. However, unlike the Arifovic's GA-based learning agents, our GP-based learning agents learn the equilibrium price by not knowing the upper limit of the price. But, since agents do not know the upper limit of the price, the phenomenon of price euphoria, which did not happen in the Arifovic's experiments, does show up quite often at the earlier stages of our experiments. Nevertheless, the interesting thing to notice is that GP-based learning can quickly coordinate agents' beliefs so that the emergence of price euphoria is only temporary. Therefore, while not knowing the upper limit of the price and the characterization of the equilibrium price, GP-based learning agents can perform as well as their GA-based counterparts.

Secondly, unlike Arifovic's experiments, this chapter does not use the election operator. The election operator, as Arifovic said, "enables an endogenous shut-off of mutation"(p.14). But, in experiments with human subjects, we do see events similar to mutation happen even after the seeming convergence to equilibrium is already achieved. One of the most recent examples can be found in Van Huyck et. al. (1994) (See their session 5, p.1001). Therefore, it is desirable to implement the experiment without shutting off mutation. Unfortunately, without using the election operator, Arifovic found that her simulations, i.e., her basic GA did not result in convergence to rational expectations equilibrium values. However, what we will see in this chapter is that, the economy composed of GP-based learning agents is capable of having a self-stabilizing feature, i.e., while mutation might deviate the temporary equilibrium price from rational expectations equilibrium price, the GP-based market mechanism can still bring it back to equilibrium. This mechanism, better known as the invisible hand in economics, was first noticed in Adam Smith's Wealth of Nations in 1776. What GP has done here is to give a visible foundation of this invisible hand.

Thirdly, to see the robustness of this self-stabilizing feature, we extend the analysis by Wellford and Arifovic by considering more unstable cases. The unstable case dealt with by Wellford and Arifovic is the one with cobweb ratio 1.05. We study the case with cobweb ratio 2 and 3 and we shall see later that these perturbation does not change the self-stabilizing feature of GP. The rest of this chapter is organized as follows. In Section 2, the cobweb model based on a mutliagent setting is briefly described. Section 3 describes the genetic programming applied in the context of the cobweb model. Computer simulations of the cobweb model using GP-based learning are detailed in Sections 4. Concluding remarks are given in Section 5.

2 The Cobweb Model

Consider a competitive market composed of n firms which produce the same goods by employing the same technology and which face the same cost function described in Equation (1).

$$C_{i,t} = xq_{i,t} + \frac{1}{2}ynq_{i,t}^2 \tag{1}$$

where $C_{i,t}$ is the cost function of firm i at time t, n is the number of firms, $q_{i,t}$ is the quantity supplied by firm i at time t, and x and y are the parameters of the cost function. The production of the goods takes one unit of time, i.e., the quantity supplied at time t must start to be produced at time $t-1$. Since at time $t-1$, the price of the goods at time t, P_t, is not available, the decision about optimal $q_{i,t}$ must be based on the expectation of P_t, i.e., $P_{i,t}^e$. The formation of $P_{i,t}^e$ is determined by genetic programming described later. Given $P_{i,t}^e$ and the cost function $C_{i,t}$, the expected profit of firm i at time t ($P_{i,t}^e$) can be expressed as follows:

$$\Pi_{i,t}^e = P_{i,t}^e q_{i,t} - C_{i,t} \tag{2}$$

Given $P_{i,t}^e$, $q_{i,t}$ is chosen at the level such that $\Pi_{i,t}^e$ can be maximized and, according to the first order condition, is given by

$$q_{i,t} = \frac{1}{yn}(P_{i,t}^e - x) \tag{3}$$

Once $q_{i,t}$ is decided, the aggregate supply of the goods at time t is fixed and the P_t which sets demand equal to supply is determined by the demand function:

$$P_t = A - B\sum_{i=1}^{n} q_{i,t} \tag{4}$$

Given P_t, the actual profit of firm i at time t ($\Pi_{i,t}$) is:

$$\Pi_{i,t} = P_t q_{i,t} - C_{i,t} \tag{5}$$

In a representative-agent model, firms are assumed to hold identical expectations and hence identical production, i.e., $P^e_{i,t} = P^e_t$ and $q_{i,t} = q_t$ for all i. In this case, Equation (4) can be rewritten as follows.

$$\begin{aligned} P_t &= A - Bnq_t \\ &= A - B\frac{1}{y}(P^e_t - x) \end{aligned} \tag{6}$$

If we further assume the expectation is realized, then Equation (6) can be further rewritten as Equation (7).

$$P_t = A - B\frac{1}{y}(P_t - x) \tag{7}$$

The only expectation which can be realized can then be found by solving Equation (7) directly, i.e.,

$$P^*_t = \frac{Ay + Bx}{B + y} \tag{8}$$

Given P^*_t, to make demand equal to supply, the quantities supplied, based on Equation (4), are given by

$$Q^*_t = \frac{A - x}{B + y} \tag{9}$$

Since P^*_t and Q^*_t appearing in Equations (8) and (9) are time invariant, we shall use P^* and Q^* to denote this steady state. The significance of P^* is that if each firm expects the price of the next period to be P^*, then the actual price will also be P^*. In economics, the rational expectations equilibrium is defined as a *fixed point* in a mapping from perceived to actual laws of motion. See, for example Sargent (1993). It is in this sense, P^* derived as above is called the *rational expectations equilibrium price* in the cobweb model, and Q^* is called the rational expectations equilibrium quantity.

Based on the description above, there are five parameters which are exogenously given in the cobweb model, namely, A, B, x, y and n. Given the learning process, i.e., the dynamics of $P^e_{i,t}$, there are at least three kinds of parameters which might affect the performance of the economy, namely, $\frac{B}{y}$, A and n. We shall describe them briefly as follows (See Table 1).

1. $\frac{B}{y}$ (cobweb ratio)

   If $P^e_{i,t} = P_{t-1}$, then by *Ezekiel's cobweb theorem*, the market price will converge to its equilibrium value if $\frac{B}{y} < 1$. Hence, the case $\frac{B}{y} < 1$ is called the *stable case* and the case $\frac{B}{y} > 1$ is called the *unstable case*. Intuitively speaking, B is the slope of the demand curve which measures the sensitivity of the price with respect to quantity supplied. If B is higher, then a slight deviation from the equilibrium quantity Q^* will cause a larger

Table 1: Parameter Values of the Cobweb Model
Used in Genetic Programming Simulations

Set	CASE 1	CASE 2	CASE 3	CASE 4
A	2.184	2.296	3.36	4.48
B	0.0152	0.0168	0.032	0.048
x	0	0	0	0
y	0.016	0.016	0.016	0.016
n	500	500	500	500
B/y	0.95	1.05	2	3
P^*	1.12	1.12	1.12	1.12

P^*: rational expectations equilibrium price.

deviation from the rational expectations equilibrium price P^*. Therefore, when $\frac{B}{y}$ is greater than 1, it works as an amplifier which amplifies any mistakes made by firms and thus makes the coordination of price expectations more difficult. In Wellford's experiment, he chose $\frac{B}{y} = 0.95$ as the stable case and $\frac{B}{y} = 1.05$ as the unstable case. These two cases were also studied by Arifovic (her sets 6 and 7.) and are named CASE 1 (stable) and CASE 2 (unstable) in this paper. Apart from the two cases, we will also study two more unstable cases, i.e., $\frac{B}{y} = 2$ (CASE 3) and $\frac{B}{y} = 3$ (CASE 4).

2. A

   In addition to the ratio of B to y, parameter A also has its significance. Based on equation (4), the demand function, A can be considered the upper limit for the prices. In other words, no matter how limited the goods are supplied, the price will not exceed A. One may wonder whether x is the lower limit corresponding to the upper limit A because, based on equation (3), firms will not produce anything unless the price is at least x. However, since the decision about production is one period ahead of the market, and it is possible that the expectation $P^e_{i,t} > x$ is not fulfilled, the price can be arbitrarily low and 0 is the lower limit. While, given the parameters, [0,A] is the range for the market clearing price, firms are not assumed to know the upper limit but must determine it individually.

3. n

   The parameter n plays a dual role here, one in economics and one in genetic programming. In economics, n could affect the prices. But suppose

$$P^e_{i,t} = P^*, \quad \forall\, i,t \tag{10}$$

then Equation (4) can be written as follows.

$$P_t = A - \frac{B}{y}(P^* - x) \tag{11}$$

So, in the rational expectations equilibrium, the price is independent of n. In genetic programming, n is a major control parameter. But given the interpretation of the *single-* and *multi-* population GA in economics made by Holland and Miller (1991) and adopted by Arifovic (1994), the choice of n can be considered a degree of *sophistication* and need not have anything to do with the number of firms. Therefore, we can independently choose the number of firms (n) in the market and the population size (M) to control genetic programming. Here, we simply let $n= M$, i.e., in Arifovic's terms, we only study the single-population GP. As to the choice of M, we keep the default choice made by Koza (1992), i.e., $n = M = 500$. A different choice such as M=100 might affect the short-run behavior of the market to a slight degree, but since it has little effect on the long-run behavior, and due to the space limit, only the discussion of the case $n = M = 500$ will be given here.

Since in our cobweb model, the rational expectations (RE) equilibrium price P^* depends only on A, B, x and y, following Wellford and Arifovic, we have set them in the way such that P^* is identical among all cases. In CASE 1 and CASE 2, the choice of A, B, x and y is the same as that of Arifovic and Wellford. Since they do not consider CASE 3 and CASE 4, we have to make our own choice of A and B by keeping x, y, B/y and P^* fixed as those of CASE 1 and CASE 2. Given the parameter values of A, B, x, y in Table 1, $P^* = 1.12$ is a result of Equation (8).

The necessary and sufficient condition for this economy to operate in its most efficient state is that all firms have correctly perceived that the RE equilibrium price is time invariant and is 1.12. However, since in our model firms are assumed to have information only about their technology, i.e., x and y, and they do not know the market condition, i.e., A and B, and the formation of other firms' forecasting, i.e., $P^e_{i,t}$, whether they can actually learn how to forecast correctly does not seem to have an obvious answer. While the experimental economics established long ago by Vernon Smith has lent us positive support to the famous Hayek Hypothesis that market has a mechanism to coordinate firms' beliefs to correctly forecast the price even though their initial knowledge about the market is very poor, the theoretical understanding of this mechanism is still limited (See Smith, 1982). This chapter will show that the convergence results observed in the laboratory with human subjects can be replicated through the computer simulations of the market composed of GP-based learning firms.

3 Population Learning via Genetic Programming

This section provides a brief description of the way we apply genetic programming to modeling population learning in the cobweb model. Detailed descriptions of genetic programming implemented here can be found in Koza (1992).

Let GP_t represent a collection of firms' price forecasting functions at time period t. A firm i, $i = 1, \cdots, n$, makes a decision about its production for time t using a tree, $gp_{i,t}$ ($gp_{i,t} \in GP_t$), a *parse tree* written over the *function set* and *terminal set* which are given in Table 2. In this paper, all simulations conducted are based on the terminal set which includes the ephemeral

Table 2: Tableau of GP-Based Learning

Population size	500
The number of trees created by complete growth	50
The number of trees created by partial growth	50
Function set	$\{+, -, \times, \%, EXP, RLOG, Sin, Cos\}$
Terminal set	$\{P_{t-1}, P_{t-2}, \cdots, P_{t-10}, R\}$
The number of trees created by reproduction	50
The number of trees created by crossover	350
The number of trees created by mutation	100
The probability of mutation	0.0033
The maximum depth of tree	17
The probability of leaf selection under crossover	0.5
The number of generations	1000
The maximum number in the domain of Exp	1700
Criterion of fitness	Profits

% is the protected division function. See Koza (1992), p.82.

random floating-point constant R ranging over the interval [-9.99, 9.99] and the price lagged up to h periods, i.e., $P_{t-1}, \cdots, P_{t-h}$. Therefore, the forecasting functions that firms may use are the linear and nonlinear functions of $P_{t-1}, \cdots, P_{t-h}$. The parameter h determines agents' ability to recall the past. To endow agents with the capability to learn not to be myopic when the cobweb ratio is high, h must be set large enough. In this paper, h is set to be 10.

The decoding of a parse tree $gp_{i,t}$ gives the forecasting function used by firm i at time period t, i.e., $P^e_{i,t}(\Omega_{t-1})$ where Ω_{t-1} is the information of the past prices up to P_{t-1}. Evaluating $P^e_{i,t}(\Omega_{t-1})$ at the realization of Ω_{t-1} will give the price predicted by firm i at time period t, i.e., $P^e_{i,t}$. The *raw fitness* of a parse tree $gp_{i,t}$ is determined by the value of the firm's profit earned at the end of time t based on Equation (5). To avoid a negative fitness value, each raw fitness value is then adjusted to produce an *adjusted fitness* measure $\mu_{i,t}$ and is given as follows.

$$\begin{aligned} \mu_{i,t} &= \Pi_{i,t} + 10 \quad if \;\; \Pi_{i,t} \geq -10, \\ &= 0 \qquad\qquad\; if \;\; \Pi_{i,t} < -10. \end{aligned} \tag{12}$$

By doing this, we are assuming that the forecasting functions $P^e_{i,t}$ which make firms lose more than \$10 will be immediately deleted in the following genetic operations. The choice of "-10" is due to the following consideration. Since at the early stages of the market operation, firms have very limited knowledge about the market, their expectations are sort of random guessing and, as a result, it is very likely that most of them could lose money. If we only consider firms with positive profits, then the selection process can easily be dominated by the few firms who earn positive profits at the initial stages. After a few generations when most of the firms start to take off and earn positive profits, this protection no longer plays any effective role.

Each such adjusted fitness value $\mu_{i,t}$ is then normalized. The *normalized fitness* value $\pi_{i,t}$ is given in Equation (13).

$$\pi_{i,t} = \frac{\mu_{i,t}}{\sum_{i=1}^{n} \mu_{i,t}} \tag{13}$$

It is clear that normalized fitness is a *probability measure.* Moreover, $\pi_{i,t}$ is greater for a better parse tree $gp_{i,t}$. Once $\pi_{i,t}$ is determined, GP_{t+1} is generated from GP_t by three primary genetic operators, i.e., *reproduction, crossover*, and *mutation.* They are described below.

1. **Reproduction:**

   Reproduction makes the copies of individual parse trees. The criterion used in copying is the normalized fitness value $\pi_{i,t}$. If $gp_{i,t}$ is an individual in the population GP_t with the normalized fitness value $\pi_{i,t}$, then in each run of selection, it will be copied into the next generation with probability $\pi_{i,t}$. The operation of reproduction does not create anything new in the population and the offspring generated by reproduction constitute only part of the population GP_{t+1}. As specified in Table 2, reproduction is performed on only 10% (50 out of 500) of the population. The rest of the offspring are generated by the other two operators, *crossover* and *mutation.*

2. **Crossover:**

   The crossover operation for the genetic programming paradigm is a sexual operation that starts with two parental parse trees which are randomly selected from population GP_t in accordance with the normalized fitness described above. Next, by exchanging the parts of these parents, two offspring are produced. This exchange begins by randomly and independently selecting one point in each parental parse tree using a uniform distribution described below.

   By the syntax of LISP, each point (atom) of a parse tree could be either a *leaf* (terminal) or an *inner node* (function). Thus, the point (atom) selected could either be a terminal or a function. As specified in Table 2, the probability that the crossover point is a terminal or a function is the same, i.e., one half. Given that a terminal or function is to be the point chosen for crossover, the probability that any terminal or function is chosen as the crossover point is uniformly distributed. For example, if the crossover point is to be a terminal, and then there are three terminals in the parse tree, the probability that any one of the three terminals is chosen for the crossover point is one-third (1/3). Unlike reproduction, the crossover operation creates new individuals in the populations. As shown in Table 2, 70% (350 out of 500) of the new generation is created in this way.

3. **Mutation:**

   The operation of mutation also allows new individuals to be created. It begins by selecting a parse tree $gp_{i,t}$ from the population GP_t based on $\pi_{i,t}$. Once a particular $gp_{i,t}$ is selected, mutation is a process of a random change of the value of points (atoms) within $gp_{i,t}$. Each point (atom) has a small probability of being altered by mutation, which is independent of other points (atoms). As specified in Table 2, the probability used throughout this paper is 0.0033. To be a syntactically and semantically valid LISP S-expression, terminals can only be altered by the member from the terminal set and functions can only be altered by the member with the same number of arguments from the function set. The altered individual is then copied into the next generation of the population. 20% (100 out of 500) of the new generation is created in this way.

These three operators together determine the dynamics of genetic programming, i.e., the dynamics of GP_t. Considering the *state* of the GP_t, Σ_t, which is made up of the price forecasting functions (beliefs) at a given time, and coupling it with the price P_t, one may conjecture what will happen for

$$\Xi_t = \begin{bmatrix} \Sigma_t \\ P_t \end{bmatrix} = \begin{bmatrix} \{P^e_{i,t}(\Omega_{t-1})\}_{i=1}^{500} \\ P_t \end{bmatrix}$$

in the limit. Let us consider the rational expectations (RE) equilibrium in the cobweb model. Given the description in Section 2,

$$P^e_{i,t} = 1.12 \quad \forall\, i,t \tag{14}$$

is a RE equilibrium. Furthermore, suppose the history of the prices are already in the RE equilibrium position, i.e.,

$$P_t = P^* = 1.12 \quad \forall\, t, \tag{15}$$

then any function of the variables in Ω_{t-1} which evaluated at the history $P_{t-1} = P_{t-2} = \cdots = 1.12$ is 1.12 is also an RE equilibrium. For example, the following forecasting functions are RE equilibria.

$$P^e_{i,t} = P_{t-1}, \quad \forall\, i,t \tag{16}$$

$$P^e_{i,t} = \frac{1}{2}P_{t-1} + \frac{1}{2}P_{t-2}, \quad \forall\, i,t \tag{17}$$

Also, in the RE equilibrium, firms do not necessarily have common beliefs. For example, the following composition of population is also a RE equilibrium associated with $P_t = 1.12$, $\forall\, t$.

$$\begin{aligned} P^e_{i,t} &= \frac{1}{2}P_{t-1} + \frac{1}{2}P_{t-2} \qquad if\ \ i = 1, \cdots, 300, \\ &\quad \frac{1}{2}P_{t-3} + \frac{1}{2}P_{t-4} \qquad if\ \ i = 301, \cdots, 500. \end{aligned} \tag{18}$$

While there are many possible RE equilibria with mutation on, none of them can possibly be a fixed point of the dynamics of the GP_t. We can certainly shut off *mutation* by using the election operator and make all those RE equilibria fixed points. But, in view of the existence of mutation-like behavior observed in experimental economics, it is not desirable to do so. The issue we are addressing is two-fold.

- Can the market composed of GP-based learning firms collectively discovers any of the RE equilibria?
- Once a RE equilibrium is discovered, can the market protect itself from deviations from the rational expectations equilibrium price which are caused by mutation? In other words, does the market composed of GP-based learning firms have a *self-stabilizing feature*?

4 Results of Simulations

4.1 Convergence and Stability

To answer the questions raised above, simulations are conducted for Cases 1, 2, 3, 4 in accordance with Table 1. For each case, we run five simulations and each simulation is conducted for one thousand periods (generations). Price patterns for one out of five simulations are exhibited in Figure 1 (corresponding to CASE 1), 2 (corresponding to CASE 2), 3 (corresponding to CASE 3), and 4 (corresponding to CASE 4). In addition, basic statistics such as average prices, standard deviations about average prices for all cases are given in Table 3. In the following, we will provide our answers based on these results.

First, Figures 1-4 indicate that the price patterns of GP-based markets have a tendency toward the rational expectations equilibrium (REE) price ($P^* = 1.12$) for all cases. Secondly, while, due to the effect of mutation, the price series of GP-based markets do not converge to the REE price in a strict sense, there seems to be a force to bring back every deviation from P^*. In other words, GP-based markets have a self-stabilizing feature. These properties are also revealed by Table 3. Based on the average of the price from generation 201 to 1000, i.e., $\overline{P_b}$, the $\overline{P_b}$ of almost all simulations does not deviate from P^* by more than 2%. The only exception is Simulation 4-2 whose $\overline{P_b}$ deviates from P^* by 4.8%. Also, comparing $\delta_{P,a}$ with $\delta_{P,b}$ or $\delta_{P^*,a}$ with $\delta_{P^*,b}$ for each simulation, we can see that after 200 periods of learning, all of the GP-based markets improve their stability by getting rid of *price euphoria* (This will be discussed in the next section).

While the cobweb ratio plays no role in determining the *long-run behavior* of GP-based markets, through mutation, it does have a temporary effect. As illustrated before, the cobweb ratio functions as an amplifier of forecasting errors. The higher the cobweb ratio, the wider the fluctuation caused by mutation. In fact, this is what we observed in the simulations. By comparing $\delta_{P,b}$ or $\delta_{P^*,b}$ across different cases in Table 3, we find that the standard deviation of unstable cases has a tendency to increase with the cobweb ratio.

4.2 Is Irrational behavior Sustainable?

In view of the restriction of GA's representation, it is quite understandable why Arifovic (1994) has to assume that firms behave as if they have already known the upper limit of P_t. One of the interesting features of these simulations with the use of GP is to see *whether the market composed of GP-based learning firms can actually learn a reasonable upper limit of P_t*. Put in another way, *is irrational behavior sustainable?*

Table 3: Results of the simulations of GP

CASE/Simulation		1	2	3	4	5
1	$\overline{P_a}$	1.114	1.109	1.110	1.119	1.126
	$\delta_{P,a}$	0.135	0.138	0.185	0.131	0.147
	$\delta_{P^*,a}$	0.135	0.138	0.186	0.131	0.147
1	$\overline{P_b}$	1.120	1.122	1.117	1.120	1.126
	$\delta_{P,b}$	0.004	0.007	0.017	0.008	0.056
	$\delta_{P^*,b}$	0.004	0.007	0.017	0.008	0.056
2	$\overline{P_a}$	1.119	1.128	1.113	1.113	1.120
	$\delta_{P,a}$	0.163	0.161	0.152	0.163	0.172
	$\delta_{P^*,a}$	0.163	0.161	0.152	0.163	0.172
2	$\overline{P_b}$	1.119	1.121	1.118	1.120	1.120
	$\delta_{P,b}$	0.003	0.010	0.009	0.004	0.023
	$\delta_{P^*,b}$	0.003	0.010	0.009	0.004	0.023
3	$\overline{P_a}$	1.118	1.120	1.098	1.161	1.112
	$\delta_{P,a}$	0.192	0.160	0.186	0.261	0.298
	$\delta_{P^*,a}$	0.192	0.160	0.188	0.264	0.298
3	$\overline{P_b}$	1.119	1.122	1.114	1.137	1.108
	$\delta_{P,b}$	0.009	0.018	0.047	0.154	0.234
	$\delta_{P^*,b}$	0.009	0.018	0.048	0.155	0.235
4	$\overline{P_a}$	1.152	1.178	1.131	1.125	1.134
	$\delta_{P,a}$	0.342	0.327	0.252	0.352	0.198
	$\delta_{P^*,a}$	0.344	0.333	0.252	0.352	0.199
4	$\overline{P_b}$	1.140	1.174	1.120	1.113	1.117
	$\delta_{P,b}$	0.170	0.231	0.078	0.328	0.096
	$\delta_{P^*,b}$	0.172	0.237	0.078	0.328	0.096

$\overline{P_a}$ = average price of a simulation (from Generation 1 to 1000).
$\overline{P_b}$ = average price of a simulation (from Generation 201 to 1000).
$\delta_{P,a}$ = standard deviation about the average price of a simulation (from Generation 1 to 1000).
$\delta_{P,b}$ = standard deviation about the average price of a simulation (from Generation 201 to 1000).
$\delta_{P^*,a}$ = standard deviation about the rational expectations equilibrium price of a simulation (from Generation 1 to 1000).
$\delta_{P^*,b}$ = standard deviation about the rational expectations equilibrium price of a simulation (from Generation 201 to 1000).

Table 4: The Frequency and the Duration of Crashes

CASE	Frequency	Duration
1	12, 12, 20, 9, 8	49, 100, 63, 69, 121
2	12, 12, 15, 12, 13	106, 46, 60, 108, 171
3	17, 12, 9, 23, 30	63, 46, 99, 213, 566
4	30, 22, 22, 29, 13	380, 274, 68, 440, 93

The frequency column shows the frequency $P_t = 0$ over 1000 generations for the five simulations in each case, and the duration column shows when the last crash happens. The five numbers in each box are ordered from the first simulation to the fifth.

In the context of the cobweb model, "irrational behavior" means the price expectations based on the *mass euphoria*. These extremely optimistic price expectations can result in huge supply of goods which will push the market-clearing price down to 0 and can induce further instability of the economy. Basic economics will argue that, even though people are not rational by *nature*, they are at least rational by *nurture*, and our simulations of GP-based markets show that this is exactly the case.

From Figures 1 to 4, i.e., from the stable case to the most unstable, it seems that the irrational behavior characterized by mass price euphoria exerts great influence on the operation of the early stages of every market. This can be seen from the fact that the price drops down to 0 (crash) very frequently. However, the interesting thing is that this phenomenon is only temporary. In most cases, it disappears before the 400th period (See Table 4). The worst record that we have is Simulation 3.5 in which the crash continues until period 566 and then it disappears. Based on the results of these simulations, we come to the conclusion that markets composed of GP-based learning firms can reasonably learn the upper limit of the price and that the chance of crashes decreases with time. Therefore, even if irrational behavior is sustainable, its occurrence turns out to be an exception rather than a rule.

This result also reveals the difference between GA and GP in search space. The search space given by GA is finite, say, Ψ= $[0, P_\kappa]$, and $P^* \in \Psi$. Thus, we can choose a reasonable size of Ψ such that price euphoria will not happen. This is what we observed in Arifovic's experiments. However, by doing this, the evolution process (or the learning process) by which agents learn the reasonable size of Ψ is completely missing. On this matter, Koza (1992) has stated: "The predetermination of the size and shape of solutions and preidentification of the particular components of solutions has been a bane of machine learning systems from the earliest times. (p.63)".

Compared with GA, the search space in terms of the price forecasts determined by GP is Φ =$[0, P_\kappa]$, and P_κ is the largest real number that computer can handle. Since $\Psi \subset \Phi$, it takes a while for GP to pinpoint the Ψ in Φ. The time resource estimated to be needed to perform this job is listed on the duration column of Table 4. Once this is done, GP has rediscovered the knowledge which is given to GA as manna. From this viewpoint, GP enables us to investigate a complete evolutionary process and GA can be considered part of the outcome of this process.

4.3 The Dynamics of Beliefs: From the Sophisticated to the Simple

In addition to the dynamics of P_t, it is also interesting to see the evolution of Σ_t, i.e., firms' beliefs. In all simulations, Σ_t starts with quite a large variety of beliefs. They usually contain programs with length ranging from three to more than a hundred and three is the minimun length that we can have. Moreover, complex programs are frequently selected as the best forecasting functions at the early stages of evolution. Take Simulation 4.5 as an example. The following five forecasting functions (beliefs) are selected as the best individual in Generation 4, 15, 22, 65, 72 respectively.

$$\begin{aligned}
P^e_{best,4} &= ((2.33 * Sin(Exp(Cos(-8.16\% P_{t-1})))) - ((P_{t-3}\% P_{t-1}) - (P_{t-2}\% P_{t-4}))) \\
P^e_{best,15} &= ((((P_{t-9} - P_{t-1})\%(P_{t-5} + 9.03)) + Log(Sin(P_{t-8} * P_{t-6}))) * (Log(Cos(P_{t-3}) \\
&\quad *Cos(P_{t-1})) - (Exp(Exp(-7.70)) - (P_{t-1}\% Log(3.97))))) \\
P^e_{best,22} &= Exp(Cos(Sin(((P_{t-6} * (-6.16\% Sin(Cos(-8.16 - P_{t-7})\% Sin(P_{t-2}))))\% P_{t-3}) \\
&\quad -(P_{t-9} - P_{t-1})) - Log(P_{t-3}))) \\
P^e_{best,65} &= (Sin((Cos(Cos(Exp(P_{t-7}) + Log(Cos(Cos P_{t-3} * P_{t-9}))) + P_{t-8}) + P_{t-7})\% \\
&\quad Sin(Sin(Cos(P_{t-6}\% P_{t-3})))) - P_{t-8}) \\
P^e_{best,72} &= ((Cos(Cos(Cos(Exp((P_{t-3} - P_{t-7}) * Cos(P_{t-9})))) + P_{t-8}) + P_{t-3})\% Sin(\\
&\quad Sin(Cos(P_{t-6}\% Exp(Cos(P_{t-10}))))))
\end{aligned}$$

However, as time passes, Σ_t gradually loses its diversity and the size of the program also becomes smaller. For example, in Simulation 4.5, the following five forecasting functions are selected as the best individual in Generation 323, 328, 351, 352, 381 respectively.

$$\begin{aligned}
P^e_{best,323} &= ((P_{t-5} * ((P_{t-5} * (P_{t-3} * (P_{t-1} * P_{t-7}))) * P_{t-4})) * P_{t-4}) \\
P^e_{best,328} &= (((P_{t-7} * P_{t-9}) * P_{t-9}) * (P_{t-1} * P_{t-4})) \\
P^e_{best,351} &= ((P_{t-2} * (P_{t-4} * P_{t-3})) * P_{t-3}) \\
P^e_{best,352} &= (((P_{t-3} * P_{t-1}) * P_{t-1}) * P_{t-4}) \\
P^e_{best,381} &= (P_{t-2} * P_{t-1})
\end{aligned}$$

Eventually, it ends up with programs whose code length is only three, i.e., the simple functions (beliefs) P_{t-1}, P_{t-2},$\cdots$, and the best individual selected is just one of these types. For example, in Simulation 4.5, after Generation 388, the best individual of each generation is a member of the simple functions. These combination of simple beliefs associated with P^* is just one of the RE equilibria mentioned above. Hence, GP-based markets are capable of discovering the RE equilibria. Moreover, based on the outcomes of the simulations, Σ_t will not end up in any state which is not the member of the RE equilibria.

The last result appears to offer nothing exciting, since to survive firms only need to adopt the simplest strategy. But, in fact, this result is rather impressive. It confirms one of the important lessons taught in economics, i.e., under perfect competition, all kinds of sophisticated strategic behavior will eventually be useless because the price in the rational expectations equilibrium will not fluctuate very much and, under these circumstances, it simply doesn't pay to speculate with sophisticated strategies. The emergence and extinction of speculation can be briefly described as follows.

From Figures 1-4, since almost all GP-based markets start with a high volatility of the price, this fluctuation allows entrepreneurs room for speculation, i.e., they can make a huge profit or avoid a huge loss as long as they can capture the law of motion of P_t. At the early stages of the evolution, many strategies are selected as the best not because they can successfully discover P^* but because they just luckily predict a coming boom or crash of the price. For example, the function $P^e_{best,4}$ is the best because it successfully predicts a coming boom, i.e., $P_4 = 2.3$ and earns the profit 0.34 which is substantially higher than the profit firms can earn under the rational expectations equilibrium, i.e., 0.14. On the other hand, the function $P^e_{best,65}$ is the best because it correctly predicts a coming crash, i.e., $P_{65}= 0$ and leads the believer to withdraw from the market at period 65 and safely avoid a collapse.

The use of GP in this paper distinguishes itself from most of the others in the following respect. It is not used to solve a traditional optimization problem. The way we search for the solution will not affect the solution itself. Here, the solution is *endogenous*, and it depends on how we search. There will be a heavy loss if everybody anticipates that there will be a huge profit, and vice versa. To *go against the wind* is always the best strategy to make profits. Nevertheless, the imitation of successful rivals undertaken by the reproduction operator gradually makes the beliefs of the firms more homogenous, and hence squeezes the space for speculation. This can be seen from the decreasing volatility of prices and from the fact that there is simply no need to speculate when the price does not change much. In sum, the dynamics of Σ_t and P_t interactively works in a reinforcing way. On one hand, the evolution of Σ_t diminishes the volatility of P_t. On the other hand, the dynamics of P_t simplifies and homogenizes Σ_t.

5 Concluding Remarks

This paper applies genetic programming to extending the analysis of Arifovic's GA-based cobweb model. While the advantage of GP over GA has been well illustrated in Koza (1992), this paper realizes those advantages in the aspect of market processes. It shows that even though firms in GP-based markets require less prior knowledge than firms in GA-based markets, in the long run they perform equally well in terms of discovering the rational expectations equilibrium price. Furthermore, when we make the agents in our models more *limited* in their prior knowledge, we actually enrich the transition dynamics. In this paper, the phenomenon of mass euphoria and speculation-like activities are observed. None of these phenomena survive in the long run. In sum, a complete evolutionary process of the market is observed. The emergence of GA-like behavior studied by Arifovic and the emergence of common expectations (representative agent) are just the products at different stages of evolution.

Recently, issues which are difficult to be solved analytically in learning, adaptation and coordination in complex adaptive economic systems have been tackled by experimental economics. While its findings are fruitful, experimental economics can only be operated in a limited domain with a small sample as the cost of human subjects are simply not affordable. If GP can successfully replicate the results of experiments implemented in a small domain, then it can certainly help us explore the properties existing in a larger domain which the experimental approach is unable to handle. Given its advantages, we believe that GP can be a very promising approach in economics.

Acknowledgments

Research support from NCS grant No.84-2415-H-004-001 is gratefully acknowledged. My special thanks go to John Duffy for his valuable suggestions.

References

[1] Arifovic, J. (1994), "Genetic Algorithm Learning and the Cobweb Model," *Journal of Dynamics and Control*, Vol.18, No.1, January, pp. 3-28.

[2] Bray, M. M. (1982), "Learning, Estimation and the Stability of Rational Expectations," *Journal of Economic Theory*, 26, pp. 318-339.

[3] Chakraborty, U. K. and D. G. Dastidar (1993), "Using Reliability Analysis to Estimate the Number of Generations to Convergence in Genetic Algorithms," *Information Processing Letters* 46, pp.199-209.

[4] Clower, Robert W. (1976), "A reconsideration of the theory of inflation", in *Money and Markets*,essays by Robert W. Clower , edited by Donald A. Walker, Cambridge University Press.

[5] DeCanio, S. J. (1979), "Rational Expectations and Learning from Experience," *Quarterly Journal of Economics* 93, pp. 47-57.

[6] Eiben, A. E., E. H. L. Aarts and K. M. Van Hee (1990), *Global Convergence of Genetic Algorithms: An Infinite Markov Chain Analysis*, Computing Science Notes, Eindhoven University of Technology.

[7] Ezekiel, M. (1938), "The Cobweb Theorem," *Quarterly Journal of Economics* 52, pp. 255-280.

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

[9] Holland, J. H. (1975), *Adaptation in Natural and Artificial Systems*. University of Michigan Press.

[10] Holland, J. H. and J. H. Miller (1991), "Artificial Adaptive Agents in Economic Theory," *American Economic Review*, Vol. 81, No. 2, pp. 365-370.

[11] Kinnear, Jr. K. E. (1994) (ed.), *Advances in Genetic Programming*, MIT Press.

[12] Koza, J. R. (1989), "Hierarchical Genetic Algorithms Operating on Populations of Computer Program", in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, Morgan Kaufmann.

[13] Koza, J. R. (1992), *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press.

[14] Lucas, R. E., Jr. (1986),"Adaptive Behavior and Economic Theory," *Journal of Business* 59, pp. 401-426.

[15] Marcet, A. and T. Sargent (1989), "Convergence of Least Squares Learning Mechanisms in Self-Referential Linear Stochastic Models," *Journal of Economic Theory* 48, pp.337-368.

[16] Muth, J. F. (1961), "Rational Expectations and the Theory of Price Movements," *Econometrica* 29, pp. 315-335.

[17] Rudolph, G. (1994), "Convergence Analysis of Canonical Genetic Algorithms," *IEEE Transactions on Neural Networks* Vol. 5, No.1, pp. 96-101.

[18] Sargent, T. J. (1993), *Bounded Rationality in Macroeconomics*, Oxford Press.

[19] Smith, V. L. (1982), "Market as a Economizers of Information: Experimental Examination of the "Hayek Hypothesis"," *Economic Inquiry*, pp.165-179.

[20] Van Hyuck, J. B., J. P. Cook, and R. C. Battalio (1994), "Selection Dynamics, Asymptotic Stability, and Adaptive Behavior," *Journal of Political Economy*, Vol. 102, No. 5, pp. 975-1005.

[21] Wellford, C. P. (1989),"A laboratory analysis of price and dynamics and expectations in the cobweb model", Discussion paper 89-15 (University of Arizona, Tucson, AZ).