Journal of the Operations Research
Society of Japan
2004, Vol. 47, No. 3, 163-181

MULTI-AGENT-BASED MODELING OF ARTIFICIAL STOCK MARKETS BY USING THE CO-EVOLUTIONARY GP APPROACH

Xiaorong Chen
Shanghai Jiaotong University

Shozo Tokinaga
Kyushu University

(Received January 10, 2003; Revised March 19, 2004)

Abstract This paper deals with multi-agent based modeling of artificial stock market by using the co-evolutionary Genetic Programming (GP) by considering social learning. Cognitive behaviors of agents are modeled by using the GP to introduce social learning as well as individual learning. Assuming five types of agents, in which rational agents prefer forecast models (equations) or production rules to support their decision making, and irrational agents select decisions at random like a speculator. Rational agents usually use their own knowledge base, but some of them utilize their public (common) knowledge base to improve trading decisions. By using the result of simulation studies on artificial market, it is shown that the time series for stock price is resemble to real stock price statistically. It is also shown that the lack of social learning leads the system to a very monotone market, and only a simple behavior of the market is realized. Moreover, we can see the effectiveness of classifier systems where we utilize a pool of decision rules in which not only prominent but also rules having potential rewards in fluctuating environment. It is also seen that the growth of wealth of irrational agent is almost always better than rational agents even though they analyze and behaves on reasonable decision. The result provide us the way to analyze real market where traders usually use social learning and environment-dependent rules.

Keywords: Finance, artificial market, multi-agent-based modeling, genetic programming, co-evolutionary learning

1. Introduction

In recent years, the theory and practice of Complex Adaptive System proposed by Holland has been a major focus of complex system research [7], [8]. The perspective that complexity originates from adaptation is the core of this theory and this theory emphasizes that main component of complex system is the active, adaptive agent. Especially, the multi-agent systems describing the artificial stock market is expected to provide us a way to analyze the behavior of agents (traders) in the stock market.

The real stock market can be considered to be a very complex system [12], [20]. Specifically, traders in real stock market base their current behavior partly on their past experience and partly on perceived market characteristics, which their past individual behavior has helped to determine. This feedback loop can lead to intricate relationship between behavior and outcomes that is difficult to understand and predict by standard analytical and statistical tools.

We treat such stock market as a complex adaptive system in this paper and abstract some types of agents from a great variety of traders in real stock market. Then we attempt to study the relationship between the behavior of agents and the characteristics of the stock market [4].

The use of Agent-Based Computational Modeling of artificial stock market is driven by a series of empirical puzzles, which are still hard to explain using traditional representative agents structures. Actually, the image of artificial stock market as groups of interacting agents, continually adapting to new information and updating their expectation models, seems like an accurate image of how real stock market operates [2], [6], [7], [8], [12], [17], [18] and [19].

A wide range of computer-based evolutionary algorithms existing can be applied for this objective, including classifier system, Genetic Algorithm (GA), Genetic Programming (GP), neural network, and etc. From many literatures related to application of modeling in artificial stock market, three important research papers are notable, in which three different schemata, are proposed [2], [6], [8] and [19].

In reference [18], GA has been proved to be a powerful method to locate improvement in complicated higher-dimensional spaces. However, agent's interaction with each other is not dealt with, namely an agent depends only on his own past experience and the historical datum entirely, without interacting with other one else. This is a type of individual learning. In reference [19], fuzzy method makes agents behave more like the participant in real market. In reference [2], there exist no rules but forecast models represented by the tree structure in the GP.

In this paper we attempt to tackle the insufficient heterogeneity of agents to some extent, by introducing five types of agents [4]. We emphasize the role of co-evolutionary GP and the classifier systems to emulate the real stock market based on the multi-agent systems. Contrast to conventional GP approaches, we introduce co-evolutionary GP in learning, and also utilize the production rules as well as the arithmetic models for stock price prediction. These agents are considerably different in the way of being rational or irrational, and preferring forecast equation models or simple trading rules to guide their decision making.

Specifically, the agents of type 1 and type 3 prefer applying forecast equation models to support their decision making, however they are different in owning their individual forecast model bases or just learning from public base. On the other hand, the agents of type 2 and type 4 are identical in preferring applying simple trading rules to support their decision making, but different in owning their individual bases or just learning from public base too. Actually, we not only implement individual learning through introducing the agents of type 1 and type 2, but also take social learning (co-evolutionary GP) into consideration by introducing the agents of type 3 and type 4. Besides these rational agents, one type of irrational agents is also defined.

Moreover, in this paper, we focus on applying the GP approach to model cognitive behavior of adaptive agents ,since this approach has been demonstrated effective to accomplish this task. However, different from our previous work with GP [3], [9], [10] and [11], in this paper genetic operations are applied not for the optimization purpose, but for maintaining a diverse evolving forecast model or trading rule population.

By using the result of simulation studies on artificial market, it is shown that the time series for stock price is resemble to real stock price statistically. It is also shown that the lack of social learning leads the system to a vary monotone market, and only a simple behavior of the market is realized [4], [9]. Moreover, we can see the effectiveness of classifier systems where we utilize a pool of decision rules in which not only prominent but also rules having potential rewards in fluctuating environment. The result provide us the way to analyze real market where traders usually use social learning and environment-dependent rules.

In the following, in Section 2, we propose a so-called multi-agent-based architecture for modeling artificial stock market. In Section 3 and Section 4, we show how to model adaptive behavior of rational agents by applying GP approach. In Section 5, results from simulation experiment are analyzed in details in order to demonstrate the effectiveness of this proposed architecture. And, in Section 6, several important issues are emphasized.

2. Multi-Agent Based Modeling of Artificial Stock Markets

2.1. Architecture of agents

At the beginning, we show an overview of the multi-agent system treated in the paper, especially by focusing on the relation between the knowledge used by agents and the characteristics of agents.

It is expected that an experiment consisting of artificial agents allows us to know the role of utility, risk aversion, information, knowledge, expectations and learning of each subject. Moreover, knowledge and learning of the artificial agents can provide us an insight into the reason of various state of market, and the effects of variations of the environment.

Figure 1 depicts the complicated computational architecture of stock market in detail. The major component of this architecture is the heterogeneous adaptive agents. And we find that we are faced with the same formidable task to design agents. Specifically, five types of agents are defined in this architecture.

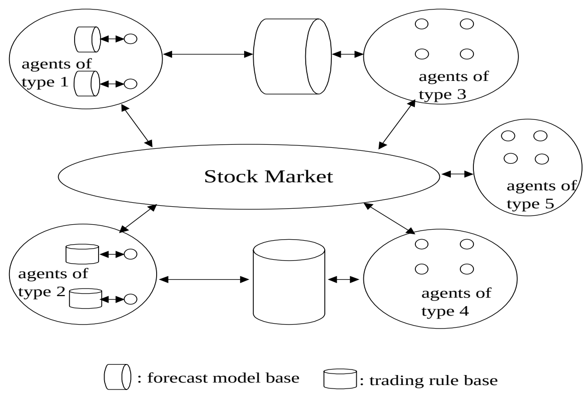


Figure 1: Architecture of artificial stock market

Agents of type 1 and type 3

The agents of both type 1 and type 3 are agents who forecast the value of stock price and dividend of the next period by using an adequate forecast equation model selected from a forecast model base. For example, they use following equation for predicting future price and dividend.

$$P_{t+1} + D_{t+1} = price(1) + dividend(1) + (price(1) - price(2)) * 1.05 \\ +(dividend(1) - dividend(2)) * 1.05$$

where, P_t, D_t are the price and dividend of stock at time t, and the equation used to predict the price and dividend at time $t + 1$. The terms such as $price(1), dividend(1)$ are functions to get characteristics of past stock price.

The difference of these two types is that the agents of type 1 possess their individual forecast model bases, but the agents of type 3 only learn from a public forecast model base, without their own bases. This public forecast model base, which can be supposed like a mass media providing a public place for social learning, can be accessed by all agents of type 3 simultaneously.

Besides the agents of type 3, the agents of type 1 may also make a decision to access this public forecast model base when they feel unsatisfied with the growth speed of their wealth and all the equation models they own seem not effective enough. But compared with learning from public forecast model base, they prefer updating their own forecast model bases more frequently when necessary, because they perhaps have more confidence with own equation models. Therefore, a so-called stochastic learning mechanism is presented in our model, letting the agents select from these two alternatives stochastically.

Agents of type 2 and type 4

Moreover, in reality the complexity of the market forces agents to act inductively, using simple rules of thumb. Using these simple trading rules to guide buy or sell decision, seems to be effective at some time. For example, they use following production rule to decide sell/buy stock in the next time point. if $price(1) + price(2) > av(10)$ then buy
if $price(2) > av(10)$ *and* $price(1) < max(10)$ then sell
where terms such as $price(1), av(2)$ are functions obtained from past stock price.

Based on this consideration, then we design the agents of type 2 and type 4, who will use trading rules to support their decision. The difference of these two types is that the agents of type 2 possess their individual trading rule bases, but the agents of type 4 only learn from a public trading rule base, without their own bases.

Agents of type 5

Even though the agents of type 1, type 2, type 3 and type 4 have different characteristics respectively, they do have one common characteristic, namely their rationality when making a decision whether to trade or not. On the other hand, different from these agents above, the agents of type 5 seem to behave irrationally, in the way that they do not use any reasonable approaches to support their decision making process.

2.2. On rational agents

The word " rational investor " in the corporate finance is usually defined as an investor who has a capability to exactly analyze all of available information and to avoid risk so that he can maximize his expected return based on the utility function described by the expected profit and bearing risks. In the economics, the activity is defined as a behavior based on the rational expectation, which is the counterpart of the adaptive investor who utilizes limited information for investment based on the adaptive expectation.

At the beginning, we would like to previously notice that the definition of " rational agent " of the paper is not the same as the definition of rational investor in real market mentioned above. In the paper, agents use only the stock price which is realized in the artificial stock market, and their definition is different from the rational investor in real market who behaves based on the rational expectation. Therefore, the types of agents are distinguished only by the definition of rationality for a single information of stock price. Even though the two definitions are not the same, it is useful to make some comparison.

By comparing the rational investor in the corporate finance with the agents of type 1 through 5, type 1 and 3 agents make their decision on the appropriate amount of stock based on their own prediction model, and as a result they can have optimal stock holding which maximize the utility function defined by the expected return and risks. From these aspects, we can say that they make a similar decision as the rational investor in corporate finance does.

However, they have no capability of processing other information such as the news and the change of interest rate. Moreover, type 2 and 4 agents act to maximize the expected return, but they have no concern about the risks. Type 5 agents seem to be speculators who decide buy or sell action and the volume to trade.

2.3. On classifier systems

Here, before making a decision, a rational agent must select from many candidate equation models from forecast model base or trading rules from the corresponding base according to the accuracy of the models of rules, evaluated using historical datum. We allow the agents to adopt a stochastic selection strategy, namely they can select an equation model or a trading rule which obtains better fitness to the environment with the accuracy above the average level. It is necessary to note here that a selection strategy to select an equation or a trading rule with highest fitness is not employed in this paper, because we do not believe that an equation model or a trading rule with a maximum accuracy on historical datum is always successful in future in the context of violently changeable market. The searching behavior of agents can be realized through this mechanism.

The idea flows the LCS (Learning Classifier System) proposed by Holland which models its environment by activating appropriate clusters of rules[7], [8]. Overt actions affecting the environment are the result of messages directed to the output, while information from the environment is received via messages. A LCS rule does not automatically post its message when its condition part is satisfied. Rather, it enters a competition with other rules having satisfied conditions.

In this paper, the cognitive behavior of adaptive agents will be modeled by applying the GP approach. Specifically, forecast equation model and trading rule are represented by tree structure. Forecast model bases and trading rule bases are evolving in responding to the market dynamics, enforced the genetic operations. The details will be discussed in Section 3.1 and 3.2 respectively. Then as the users of these evolving bases, agents can behave like they are adaptive to the market dynamics. In other words, all the components in this artificial stock market are co-evolving as the time going on.

3. Agents of Type 1 and 3 Using Stock Price Prediction

3.1. Basic computational model

It is assumed that the market structure is set up to be a traditional two-asset market. There are two assets traded, a stock with price P_t that pays an uncertain dividend D_t and a risk-free bond that pays a constant interest rate r_f. Stock prices to clear the market are set endogenously. The dividend D_t is assumed to follow $AR(1)$ process as follows, in which ϵ_t is Gaussian noise (i.i.d and $N(0, \sigma_\epsilon^2)$)and $\bar{D}$,ρ are constants[6].

$$D_t = \bar{D} + \rho(D_{t-1} - \bar{D}) + \epsilon_t \tag{1}$$

For simplicity, there are N stocks and M heterogeneous agents in the market, and each agent initially endowed with a fixed number of stocks equivalently.

Agents of type 1 and type 3 make predictions about the expectation of future return and risk using accurate forecast equation models, and then buy or sell stocks in corresponding with their expectation. Agents are assumed to have CARA (Constant Absolute Risk Aversion) utility function as follows, in which W represents wealth and λ is degree of relative risk aversion. And agents are assumed to be myopic to endeavor to maximize their expected utility.

$$U(W) = -exp(-\lambda W) \tag{2}$$

Although the way of decision making of these two types of agents are similar fundamentally, they are heterogeneous in terms of their individual expectation of future stock price and dividend. Assuming that agent i's expectation about stock price and dividend in time $t+1$ is distributed with mean $\hat{E}_{i,t}[P_{t+1} + D_{t+1}]$ and variance $\hat{\sigma}^2_{i,t}$. Under the assumption that conditional stock price and dividend are Gaussian, agent i's preferable stock position $X_{i,t}$ can be gained by the following equation[6].

$$X_{i,t} = \frac{\hat{E}_{i,t}[P_{t+1} + D_{t+1}] - P_t(1 + r_f)}{\lambda \hat{\sigma}^2_{i,t}} \tag{3}$$

The value r_f denotes the interest rate of risk-free asset, and the amount $X_{i,t}$ means the volume of stock which the agent i would like to possess in the next time period. The difference between current amount of stock and the value in equation (3) decides the volume of trade of agent i in the market.

Compared with agents using accurate forecast equation models, agents of type 2 and type 4 prefer using simple rules of thumb to decide whether to trade or not , but they can not know the optimal quantities to trade exactly. These rules are called condition-action rules, which means if the condition part of the rule is satisfied then the action represented by the action part will be implemented. For example, a rule can be noted as 'if the price of period $t-1$ is lower than the average price of 50 periods previous, then buy stock '. Moreover, these types of agents are designed to buy or sell stocks at a stochastic quantity.

Compared with the agents described above, agents of type 5 are the only irrational agents. Whether to buy or sell stocks and the adequate quantity to trade are not determined at an explicit way. In this paper, we assume that this type of agents observe the trading movement in the market first and then decide to do as others do or select the opposite way at a stochastic fashion.

3.2. Assessing stock price

Actually, there is little possibility that the total number of stocks demanded by all agents will equal the total number of stocks supplied by all agents exactly. Then an adjustment mechanism is necessary and for simplicity designed as follows. Demand of some agents can be satisfied completely, but the others are not lucky enough.

A stock price to clear the market can be found by balancing the demand and the supply of stocks, for simplicity we also utilize the same price adjustment schema as used in Reference [5]. The stock price can be adjusted according to the following equation, in which β is a function of the difference between total quantity agents would like to buy denoted as B_t and total quantity agents would like to sell denoted as O_t.

$$P_{t+1} = P_t(1 + \beta(B_t - O_t)) \tag{4}$$

Then we consider one form of function β as follows, where tanh is the hyperbolic tangent function.

$$\beta(B_t - O_t) = \begin{cases} tanh(\beta_1(B_t - O_t)) & (B_t \geq O_t\ ) \\ tanh(\beta_2(B_t - O_t)) & (B_t < O_t) \end{cases} \tag{5}$$

In summary, a agent makes a decision about his preferable stock position in his own way independently. Then a stock price to balance the demand and supply of stocks can be revealed endogenously. This procedure is repeated again and again and the market dynamics can be generated gradually.

Here it is obvious that the market dynamics is resulted from the co-operation of all types of agents. The key issue is to design adequate mechanisms how agents of type 1 and type 3 can form individual expectation about $\hat{E}_{i,t}[P_{t+1} + D_{t+1}]$ applying forecast equation models and how agents of type 2 and type 4 can decide trading action applying trading rules effectively. We will discuss this issue in detail in next section.

Table 1: Explanation of some functions

Function	Explanation
min(t)	minimum price in period[itime-t,itime-1]
max(t)	maximum price in period[itime-t,itime-1]
av(t)	average price in period[itime-t,itime-1]
price(t)	stock price in period itime-t
dividend(t)	dividend in period itime-t

3.3. Modeling adaptive agents utilizing forecast model by the GP

As mentioned above, despite agents of type 1 and type 3 access different forecast model bases, they are identical in the way that they apply forecast equation models to support their decision making process.

In the GP, each forecast model is represented in the tree structure (called individual) [14]-[16]. In the parse tree, the non-terminal node is taken from the function sets, containing
+,-,÷,×
exp, abs ,sqrt
log, min, max, av, price, dividend
The explanations about the functions like min, max, av, price, dividend, having only one operand, are shown in Table 1, in which *itime* denotes the current period. It should be emphasized here that the functions +,-,÷,× have two operands and by contrast, the functions exp, abs, sqrt, log, min, max, av,price, dividend have only one operand. In order to tackle the functions in the same way, the functions with only one operand will be allowed to have a dummy operand.

Terminal node consists of arguments chosen from set of constants.

Therefore, we deviate from earlier researches in that agents are not restricted to use linear prediction models. Moreover, we allow agents to use any historical information as they like to build forecasts. Of course, at some time complicated equations without explicit practical meanings can be generated through the GP, which are difficult to understand but perhaps have good forecast accuracy. We allow agent's individual forecast model base to contain 50 models and public forecast model base to contain 100 models.

For representation of tree structure, the prefix representation is utilized. The prefix representation follows traditional representation by using the Lisp syntax. For example, we have the following example of prefix representation.

$[6.43 \times x - y] \times [z - 3.54] \rightarrow \times - \times 6.43 \;\; x \;\; y \;\; - \;\; z \;\; 3.54$

The evaluation of prefix representation is done with the stack operation. We begin to scan the prefix representation, and if we meet the terminal node (operand) then we push down the term into the stack. If we meet the non-terminal node (operator), then we take out the operands from the stack, and execute the operation. The result of the operation is also pushed down to the stack.

We define the fitness as the accuracy of forecast equation model on historical datum. Specifically, the fitness of forecast model is the reciprocal number of squared forecast error denoted as $e_{i,j,t}$, in which i is the index of agent, j is the index of forecast model in the base and t accounts for time horizon. $e_{i,j,t}$ can be calculated in a smooth exponentially weighted fashion as follows,in which w denotes the weight.

$$e^2_{i,j,t} = (1-w)e^2_{i,j,t-1} + w[(P_t + D_t) - \hat{E}_{i,j,t-1}(P_t + D_t)]^2 \qquad (6)$$

In this paper, we let the variance $\hat{\sigma}^2_{i,t}$ estimated by agents of type 1 and type 3 equal to the squared forecast error of selected forecast model. At the ending of every period, the fitness of all forecast models in agent's individual forecast model base and public forecast model base will be reevaluated automatically according to their performance.

As time going on, the market dynamics change quickly and the agents must adapt to this changeable environment as quickly as possible in order to survive. In other words, the forecast model bases used by agents are not static and need to be reevaluated and updated according to their performance. To implement this, genetic operations are implemented at a certain frequency, namely crossover operation and mutation operation[9]-[11], [13]-[16], as shown in Figure 2.

In crossover operation, two fit parent individuals are selected. To keep the crossover operation always producing syntactically and semantically valid equation models, we look for the part which can be a sub-tree in the crossover operation and check for no violation. Then new individuals are generated by exchanging sub-trees between these two parent individuals.

For checking the validity of underlying parse tree, the so-called stack count (denoted as *StackCount* in the paper) is useful [9]-[11]. The *StackCount* is the number of arguments it places on minus the number of arguments it takes off from the stack. The cumulative *StackCount* never becomes positive until we reach the end at which point the overall sum still needs to be 1.

By using the *StackCount* we can see which loci on the prefix expression is available to cut off the tree for the crossover operation, and we can validate whether the mutation operation is allowed.

If final count is 1, then the prefix representation (tree) corresponds properly to a system equation. Otherwise, the tree structure is not relevant to represent the equation.

The goal of the mutation operation is the reintroduction of some diversity in the base. Two types of mutation operations are implemented to replace a part of the tree by another element, namely global mutation and local mutation. Select at random a locus in a parse tree to which the mutation is applied. Local mutation is an operation in which only the element in identified place will be replaced by another value(a primitive function or a constant). On the other hand, global mutation is an operation in which the sub-tree behind this identified place will be replaced by another sub-tree in a newly generated individual, certainly having checked for no violation.

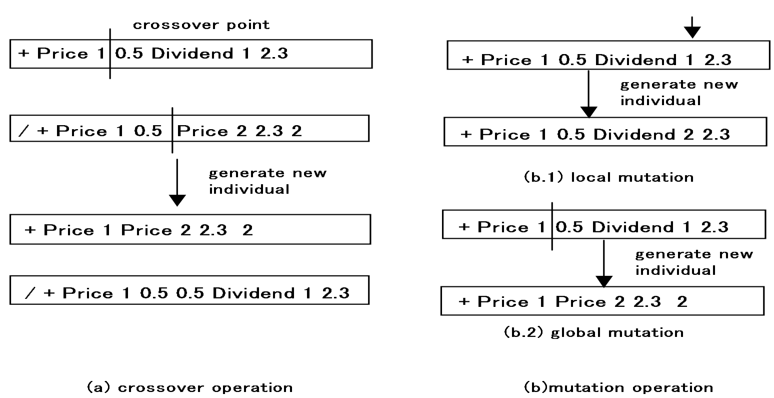


Figure 2: Genetic operations on forecast models

Therefore through genetic operations, poorly performing equation models are eliminated

and new equation models will be added to the existing bases. This process can be considered as the learning process of adaptive agents.
(Step 1)

Generate an initial population of random composition of possible functions and terminals for the problem at hand. The random tree must be syntactically correct program.
(Step 2)

Execute each individual (evaluation of system equation) in population by applying the optimization of the constants included in the individual. Then, assign it a fitness value giving partial credit for getting close to the correct output. Then, sort the individuals according to the fitness S_i.
(Step 3)

Select a pair of individuals chosen with a probability p_i based on the fitness. The probability p_i is defined for ith individual as follows.

$$p_i = (S_i - S_{min}) / \sum^{N} (S_i - S_{min}) \tag{10}$$

where S_{min} is the minimum value of S_i, and N is the population size.
(Step 4)

Then, create new individuals (offsprings) from the selected pair by genetically recombining randomly chosen parts of two existing individuals using the crossover operation applied at a randomly chosen crossover point. The algorithm is the same as the roulette strategy.
(Step 5)

If the result designation is obtained by the GP (the maximum value of the fitness become larger than the prescribed value), then terminate the algorithm, otherwise go to Step 2.

4. Modeling Agents of Type 2 and 4 Using Production Rules

4.1. Modeling trading rule by the GP

As described above, despite agents of type 2 and type 4 access different trading rule bases, they are identical in the way that they apply trading rules to support their decision making process.

In this paper, trading rules are referred as condition-action rules and generally represented as 'if A then B', where A is an antecedent (condition part) and B represents a consequence (action part). In this paper, the action part of a rule is defined as a signal to buy or sell, signal 1 for buy action and 0 for sell action. The condition part includes description of the features of stock market and we can define the condition part of a rule as connection of statements with logical operators, including AND,OR,NOT. For example, the condition part is denoted as '$statement$ 1 AND $statement$ 2 OR $statement$ 3'. And the statement can be defined as connection of two equations with comparative operators, including $>, \geq, <, \leq, =, \neq$,for example, a statement is denoted as '$price(1) > av(10)$'. The equation part of a statement can be generated in the same way as described in Section 3.1.

Similarly, we let each trading rule be represented by GP individual, as described in Figure 3(a). The overall length of the individual, being divided into condition part and action part, is set to 50 nodes. The max length of the condition part is 49 nodes, while only the last node is applied to represent the action part of a trading rule.

The condition part is also represented in the tree structure and the same style of prefix representation. To simplify the system configuration, we assume that the condition part of a rule consists of logical expressions which are represented by a single logical operator and two statements (a combination of binomial logical expressions). Then, the condition part can be represented by a prefix representation like an arithmetic expression where the arithmetic operators are replaced by logical operators. Each statement in logical expression can be represented by two arithmetic expressions and one comparative operator as depicted in Figure 3.

The relation between the logical expression and statements are realized by a hierarchical data structure as in Figure 4. In Figure 4,.the overall expression of logical expression is stored in the first level, and the links to combine the statement stored in the second level are used to aggregate comprehensive data structure.

In the parse tree, the non-terminal node is taken from the logical function sets, containing AND,OR,NOT, while the index of a statement is placed into a terminal node. We also define the fitness as the accuracy of trading rule on historical datum, specifically the percentage of successful trades (increase the wealth) in 10 periods recently.

Moreover, each statement is represented by GP individual also, as described in Figure 3(b). Similarly, The overall length of the individual is set to 51 nodes, being divided into three parts, i.e. part of equation 1, part of equation 2 and part of comparative operand. The max length of the former two parts is set to be 25 nodes respectively, while only the last node is applied to hold the part of comparative operator. And the equations in the statement are basically mathematic equations, therefore being represented just in the same way as described in Section 3.1.

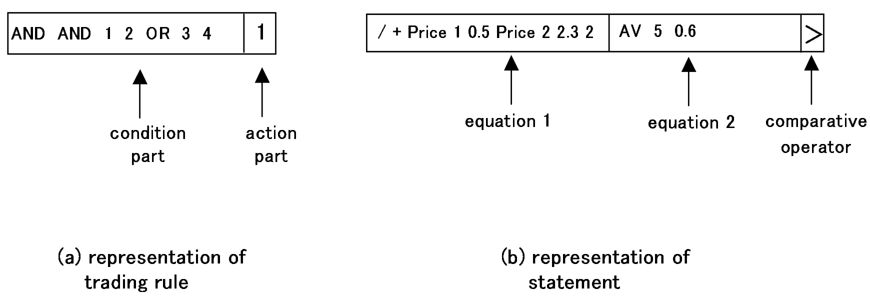


Figure 3: Representation of trading rule and statement as an individual

Then, trading rules are put into a so-called trading rule base, while statements are put into a so-called statement base. The corresponding relationship between trading rule base and statement base is shown in Figure 4. We allow agent's individual trading rule base to contain 50 rules and public trading rule base to contain 100 rules, while statement bases to contain 50 statements, respectively.

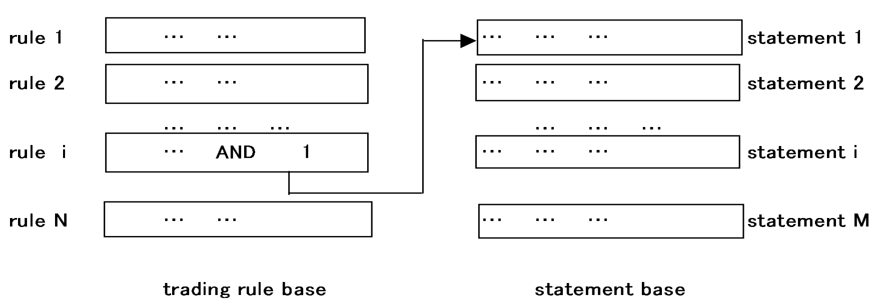


Figure 4: Corresponding relationship between trading rule base and statement base

These trading rule bases and statement bases are also manipulated by genetic operations in the same way as forecast model bases in order to simulate behavior of adaptive agents.

Some examples of genetic operations on trading rules and statements are provided in Figure 5 and Figure 6, respectively. However, in the case of crossover operation on statements, the crossover point of two parent individuals are requested to be at the same part of the statements, which guarantees no violation of new individuals.

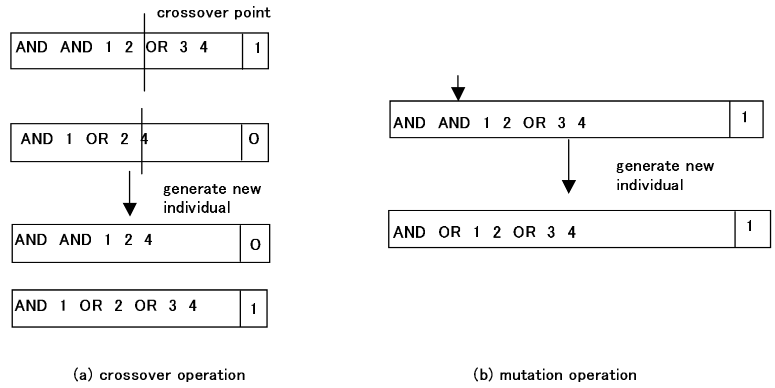


Figure 5: Genetic operations on trading rules

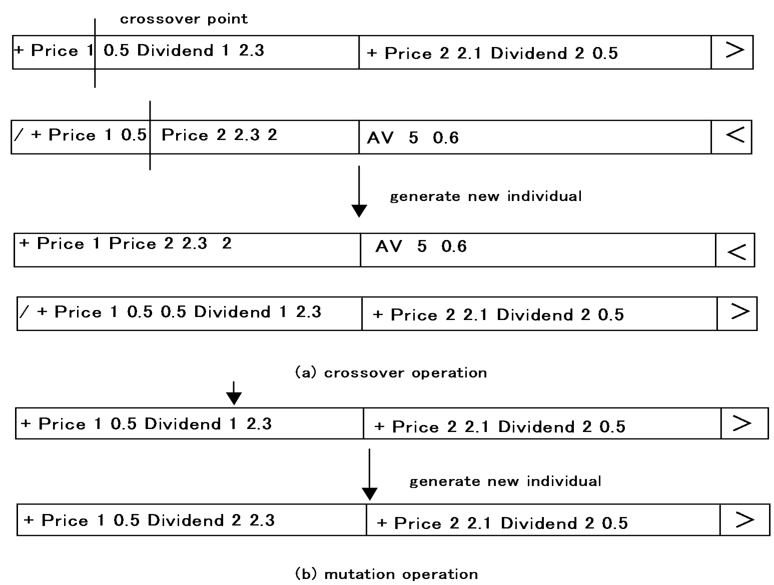


Figure 6: Genetic operations on statements

5. Computational Experiments

5.1. Experimental design

The computational experiment is designed as follows. The multi-agent-based artificial stock market was run for 3000 time periods to allow sufficient learning. There are totally 200 agents and the total quantity of stocks is fixed to 1000. Each agent must trade inside a predefined budget restriction and is also restricted to trade a maximum of 10 shares per trading. Moreover, public forecast equation model base and public trading rule base are updated every 10 periods, however agents of type 1 and type 2 prefer updating their bases more frequently, once per 5 periods.

Table 2: Statistics of return series

mean	0.00068
standard deviation	0.02
skewness	-0.115(0.045)
kurtosis	-1.414(0.089)
Kolmogorov-Smirnov test	8.552*
Jarque-Bera test	2435.425*
First-order autocorrelation	−0.257
Box-Ljung test	197.684*
ARCH LM(1)	19.0796*

Different from the researches so far, at the beginning of the experiment, the forecast model bases and trading rules bases are not initialized in a random fashion. Instead, through learning from a piece of historical datum (100 periods), more suitable forecast models and trading rules are initialized. Therefore, long early transients are avoided. In the next section, we will concentrate on the statistic features of time series generated from this artificial stock market and some other interesting features in detail.

The condition for simulation study is summarized as follows.

Number of agents

Type 1 and 3:50

Type 2 and 4:100

Type 5:50

Number of individuals

Type 1 and 3:50 for each agent

Type 1 and 3:100 for all agents (common pool)

Number of generated stock price

For obtaining statistical meaningful result, we run 1000 simulation studies to generated artificial stock price.

5.2. Basic statistics of time series

An example of time series plot of stock price is drawn in Figure 7, showing a high amount of variability. The return series is derived as the natural logarithm of stock value relatives, where dividends are part of total value. The basis statistics of return series for 1000 artificially generated stock prices are summarized in Table 2,in which numbers in parenthesis are standard deviations. These include the following statistics, i.e. mean, standard deviation, skewness, kurtosis, first-order autocorrelation, and etc. We apply both Kolmogorov-Smirnov test and Jarque-Bera test to test the normality, and Box-Ljung test to test whether autocorrelation exists. To test whether ARCH effect exists, we utilize ARCH LM(1) test [1], [5] and [21].

The null hypothesis that return series is normally distributed is completely rejected by both Kolmogorov-Smirnov test and Jarque-Bera test at the 95% confidence level. This lines up well with the stylized fact that return is not normally distributed [13]. Moreover, empirical studies on the statistics properties of stock market have shown that time series of stock return exhibits some autocorrelation for short lags and we can see the similar property in the return series generated from our artificial stock market, since the null hypothesis of no autocorrelation in return series is rejected by Box-Ljung test at the 95% confidence level.

The null hypothesis of no ARCH effect in return series is also rejected by ARCH LM(1) test at the 95% confidence level and this reveals that our multi-agent-based artificial stock market can generate the pattern of volatility persistence. This fact is consistent with the empirical result that security returns exhibit conditional time-varying variability and ARCH effects have generally been found to be high significant in equity market [13], [21].

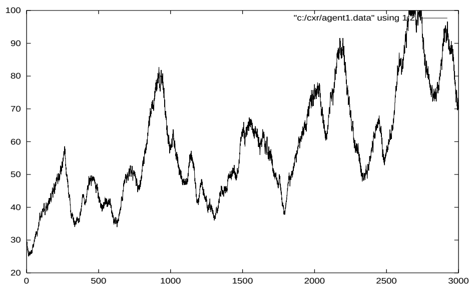


Figure 7: Example of plot of stock price time series

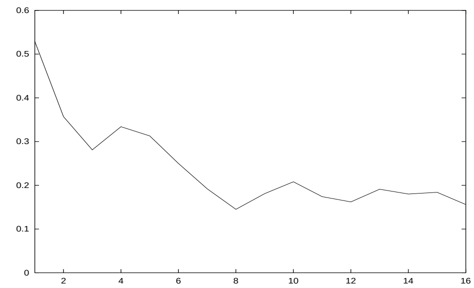


Figure 8: Example of autocorrelation of trading volume

Another well-known characteristic about stock market is that there is autocorrelation in the time series of trading volume. Figure 8 shows an example of trading volume autocorrelation in our artificial stock market. It is obvious that trading volume is autocorrelated and moreover this fact is consistent with the stylized fact that positive autocorrelation can be usually found in times series of trading volume in real stock market. Therefore, we can draw a conclusion that our artificial stock market can generate the similar pattern of volume persistence.

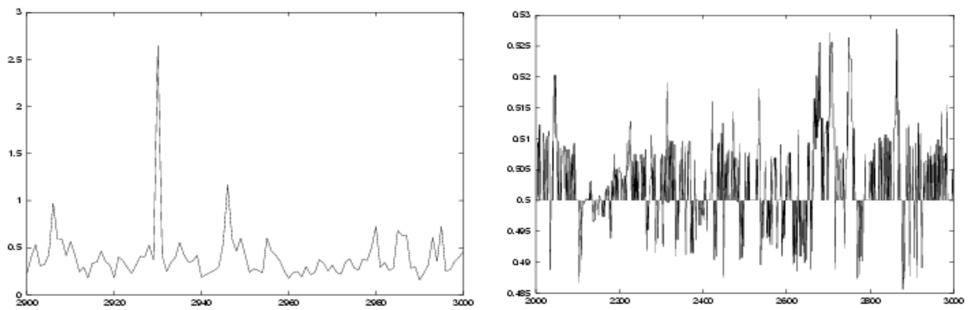
To summary, the computational experiment based on this multi-agent-based architecture has demonstrated rich dynamics of stock price and return similar to which real stock market can generate. Specifically, return series has been tested not normally distributed and shows indication of first-order autocorrelation. Moreover, there is strong evidence that ARCH effect exists in this return series. Trading volume is also tested to have first-order autocorrelation. Through these discussions, the effectiveness of this schema to model the complex behavior of agents can be proved, therefore.

5.3. Agent behavior

One important advantage of computational artificial stock market is that it can provide us with the ability to study the behavior of agents closely, which can be kept track of as time going on. In this section, we will analyze the transition of agents' behavior from two aspects, namely the transition of average fitness and average complexity of trading rule bases and forecast model bases. Through tracking the transition of agents' behavior, two characteristics of agents in our model can be revealed.

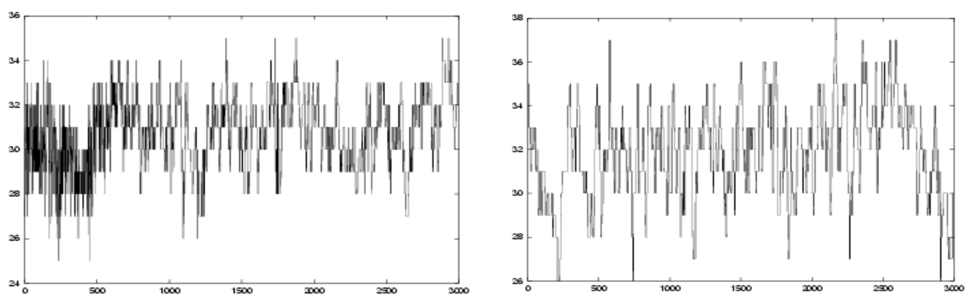
Firstly, Figure 9(a) plots an example for the change of average fitness of forecast model base owned by agents of type 1 and Figure 9(b) plots an example for the change of average fitness of trading rule base owned by agents of type 2, respectively. As time going on, it seems that there is no significant improvement of the average fitness of trading rules or forecast models, despite we expected that agents are rational enough to find more successful trading rule or forecast model gradually. We consider the reason is that in the context of violently changeable market, the value of a successful trading rule or forecast model will depreciate at a high speed. In other words, it is impossible for a successful trading rule or forecast model to success forever.

Secondly, Figure 10(a) plots an example of the change of average complexity of forecast model base owned by agents of type 1 and Figure 10(b) plots an example of the change of average complexity of trading rule base owned by agents of type 2, respectively. We define the complexity as the length of a trading rule or a forecast model. It seems that there is no significant tendency that trading rules or forecast models getting more and more complicated. In other words, this reveals a fact that the effective rule or forecast model is not equal to a complicated one.



(a) fitness of forecast model (b) fitness of trading rule

Figure 9: Example of change of average fitness



(a) complexity of forecast model (b) complexity of trading rule

Figure 10: Example of change of average complexity

5.4. Novelty of the proposed system

In terms of the agent-based system for artificial stock market, as we described in the Introduction, several researches showed agent system using the fuzzy inference system, the GA and the GP in learning process of agents. Simulation studies in these researches showed that the artificially realized stock market resembles to real market by depicting the statistical properties of generated time series and trading volumes. However, it is not impressive to us that we know the agent system behaves like a real market.

The model of artificial market should realize and show another aspect of unusual phenomena in stock market such as the emerging bubble. Furthermore, conventional researches use the proposition that every agent is homogenous in the decision making. But, it is natural to understand that in real stock market, various kinds of traders have many different methods for prediction and expectation, and the heterogeneity produces complicated change of stock price.

Our system is capable of including such kinds of frameworks, and the fact confirms the novelty of the paper. Namely, we introduce five kinds of agents including agent using production rules and irrational agent, and also the classifier system. In the discussion, we show the difference between stock prices with and without classifier system, and the random behavior of type 5 agent. We define agents by introducing several learning process. Some agents predict the stock price by numerical forecast, and determine the appropriate volume of trade. Some agents utilize the production rules based on the past record of stock price to make a trade.

Even more, we define another type of agent by introducing the knowledge bases which are commonly used or not by agents. Agents using no knowledge of the market are also members of the artificial market.

It is also assumed that traders do not use the best strategy for investment, and sometime use second-best one depending on the news of stock market and their feeling. For these reasons, we also consider the realizability of stock price by introducing the classifier system in the determination of trading rules.

Since the main purpose of the paper is to show the realization of artificial stock market based on the multi-agent systems using co-evolutionary Genetic Programming, the details of the performance and various properties of the artificial stock market will be described in another paper. Then, in the following, we quickly summarize the overview.

Chaoticity and parameters of agent systems

In chapter 5, we mainly show that the statistical properties of stock price realized by the artificial market of multi-agent systems is resemble to real stock price. However, we also see interesting phenomena by changing the parameters of the system. Namely, we can generate stock price ranging from the chaotic time series to the fractal time series depending on the parameters of system.

At first, for generating chaotic stock price, we assume that we remove type 5 agents from the system, and we do not use the classifier system. Moreover, the value of parameter λ included in equation (3) is kept in a range. Then, we see that the generated stock price reveal as chaotic time series. For testing the chaoticity of time series, we use the maximum value of Liapunov Exponents.

Generating fractal time series

Secondly, if type 5 agents become the members of the system, and the classifier system is employed, then we see that the generated stock prices bear fractality (fractal time series). For testing the fractality of stock prices, we use the statistical properties of variance for the wavelet coefficients of time series.

Usually, the real stock prices are statistically equivalent to fractal time series, then the fact confirm the capability of realization of real market in our system.

However, it is also interesting that in the system we can adjust the fractal dimension of the time series in some range by changing the ratio of agents belonging to five types, and the stock price deviated from ideal Brownian motion can be generated.

Moreover, by changing the selection of individuals in pools in the classifier systems by counting the number of individuals having higher fitness, we can control the fractality of the stock price.

Other topics such as the effect of introducing the co-evolutionary Genetic Programming, and the characteristics of classifier system and type 5 agents will be discussed in Chapter 6.

6. Discussion

6.1. Roles of co-evolutionary GP and classifier systems

Until now, the researches of artificial stock market utilizing evolutionary approach have been concentrating on either individual learning(learning from individual knowledge and experience) or social learning (learning from public knowledge). However, it must be admitted that in real stock market there exist some types of agents doing both individual learning and social learning simultaneously, just different in the learning frequency. It is a pity that the researches until now ignored this possibility, to some extent.

In our research, in order to compensate this shortage, agents of type 1 and type 3 are designed and moreover, co-evolutionary GP is applied to implement individual learning and social learning mechanism of these two types at the same time.

If we reject the type 2 and 4 agents from the system, and realize only individual knowledge base for agents (without co-evolutionary GP), then we observe following unrealistic feature in stock price.

(1) monotone stock price

We observe a simple behavior of stock price, and the characteristics found in real price is removed. Moreover, some evidence for unrealistic feature is found in statistics.

(2) Abrupt decrease in wealth

In the environment using Co-evolutionary GP, we find very gradual decrease and increase of wealth (cash plus stock) for each agents in the system. However, in the trade without co-evolutionary GP, the decrease of wealth in agents are rapid. The fact corresponds to the tendency of every traders for using a simple prediction and rule in a extremely crowded manner.

(3) Increase in variance of fitness in prediction

In case where agents use no co-evolutionary GP, the variances of fitness found in prediction equations increase compared to the case using co-evolutionary GP. For example, we find a set of mean and variance as follows.

with co-evolutionary GP:mean fitness=0.5013, variance=0.00805^2

without co-evolutionary GP:mean fitness=0.5137, variance=0.0339^2

The fat shows that the usage of co-evolutionary GP deletes irrelevant rules from the pool of individuals in the GP, and the fitness of each individual drops in some range. On the other hand, under the GP without co-evolutionary learning, irrelevant individuals stay for a long time in the pool, and that make the variance larger.

¿From the result, it is seen trading rules bear more diversity and have efficient decision in the co-evolutionary GP compared to the GP without co-evolutionary learning by obtaining adequate rules and information through the social learning.

6.2. On classifier system

Then, we examine the role of classifier system by making the pool of individuals in a specific way where we select and store only individuals who currently have high fitness. As a result, we see very simple behavior of stock price compared to the cases where we fully use the co-evolutionary GP and classifier systems where the individuals having relatively low fitness are also reserved in the system. Figure 12 depicts an example.

The fact is explained by considering agents' behavior having monotone decision rules. They usually use the same prediction and rules to buy/sell stocks. If they decide to buy the stock, then the price increases at the next time, and then they act to sell stock as the reaction for the increase of price.

Then, we can conclude that the classifier system also play an important role to realize a sound artificial stock market.

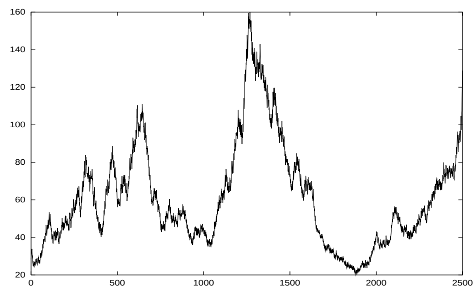


Figure 11: Example of stock price obtained by agents without co-evolutionary GP

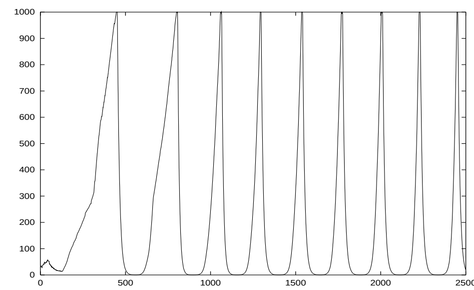


Figure 12: Example of stock price obtained by agents without LCS

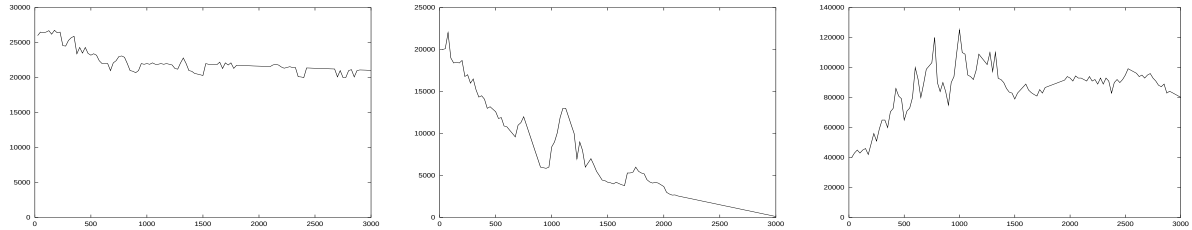
6.3. On irrational agents

We have found that the population of agents whose influence holds a dominant position will govern the evolution of the stock market, which means that the market dynamics will indeed manifest a trend expected to emerge. As a result, their wealth will increase continually or on the other hand, they will suffer fewer losses compared with others. For example, if the influence of irrational agents holds a dominant position, the market dynamics will not follow the expectation of the rational agents, since the original market movement expected to emerge is ruined by the random actions of irrational agents. Therefore, the market dynamics seems to be chaotic, to some extent.

On the other hand, consider the case that the influence of rational agents is governing the market. The random actions of irrational agents have not the strength enough to change the trend of the market dynamics at all. In this case, two extreme phenomena could be found in accordance with the expectation of rational agents. If the majority of rational agents are too optimistic, the bubble of stock price will form and the stock price will turn to be higher and higher. On the contrary if they are too pessimistic, the stock price will fall continually. Through the description above, it is obvious that the market dynamics seems very sensitive to the composition of the stock market, such as the component ratio of different types of agents.

As a matter of fact, there are many parameters in modeling this artificial stock market. Even a slight adjustment of values of parameters may lead to a big change of market dynamics. The relationship between these two have been studied, but still remains to be a difficult problem to be tackled. The research to elucidate the relationship between the values of parameters and the resulting market dynamics more clearly will be continued in the future.

Thirdly, for simplicity, exogenous events have not been considered in this model yet, and the market dynamics will only be influenced by endogenous factors. It is to some extent unrealistic because in real stock market, the behavior of agents is always influenced by some unexpected events and then market dynamics will change accordingly. It is no doubt that implementation of the mechanism of how agents act when faced with exogenous events, will improve the explanation power of this multi-agent based artificial stock market and this improvement will be done in the future.



(a) type 1 and 3 agents (b) type 2 and 4 agents (c) type 5 agents

Figure 13: Change of wealth in each agent

7. Conclusion

In this paper, we proposed an improved multi-agent-based architecture to study artificial stock market and attempt to tackle the insufficient heterogeneity of agents. Moreover, we focus on applying the GP approach to model cognitive behavior of adaptive agents. Trading rules and forecast models are reevaluated according to their performance and updated in responding to market dynamics through genetic operations. The agents defined in our multi-agent-based artificial stock market model can be considered adaptive and all forecast model bases and trading rule bases can be considered co-evolving with the market dynamics. And another issue important to be addressed is that co-evolutionary GP is applied in our research to realize both the individual learning and the social learning mechanism of adaptive agents at the same time.

The problems to be solved still remain such as the extension of multi-agent-based market to the market with extreme such as bad news or world-wide accidents like a war. It is also interesting to apply the scheme to the auctions. Further works by the authors will be continued.

References

[1] T. Bollerslev, R. Y. Chou and K. F .Kroner: ARCH modeling in finance-a review of the theory and empirical evidence. *Journal of Econometrics*, **52** (1992), 5-59.

[2] S. H. Chen and C. H. Yeh: Evolving traders and the business school with genetic programming: a new architecture of the agent-based artificial stock market. *Journal of Economic Dynamic and Control*, **25** (2001), 363-393.

[3] X. Chen and S. Tokinaga: Approximation of chaotic dynamics for input pricing at service facilities based on the GP and the control of chaos. *Transaction of the IEICE*, **E85-A-9** (2002), 2107-2117.

[4] X. Chen and S. Tokinaga: Design of multi-agent based system by using the co-evolutionary GP and its applications. *Technical Report of IEICE*, **NLP2002-62** (2002), 81-86.

[5] Robert F. Engle: Autoregressive conditional heteroskedasticity with estimates of the variance of United Kingdom inflation. *Transaction of the Econometrica*, **50-4** (1982), 987-1007.

[6] S. Grossman: On the efficiency of competitive stock market where trades have diverse information. *Journal of Finance*, **XXXI-2** (1976), 573-585.

[7] J. H. Holland: *Hidden Order: How Adaption Builds Complexity* (Addison-Wesley pub-

lishing company, 1995).

[8] J. H. Holland and J. H. Miller: Artificial adaptive agents in economic theory. *Learning and Adaptive Economic Behavior*, **81-2** (1992), 365-370.

[9] Y. Ikeda: Estimation of the chaotic ordinary differential equations by co-evolutionary genetic programming. *Transaction of the IEICE*, **E85-A-4** (2002), 424-433.

[10] Y. Ikeda and S. Tokinaga: Approximation of chaotic dynamics by using smaller number of data based upon the genetic programming. *Transaction of the IEICE*, **E83-A-8** (2000), 1599-1607.

[11] Y. Ikeda and S. Tokinaga: Controlling the chaotic dynamics by using approximated system equations obtained by the genetic programming. *Transaction of the IEICE*, **E84-A-9** (2001), 2118-2127.

[12] K. Izumu: Does learning by market participants makes financial market complicated? *Proceedings of the AAAI Workshop Multi-Agent Modeling and Simulation of Economic Systems*, (2002).

[13] J. M. Karpov: The relation between price changes and trading volume: a survey. *Journal of Financial and Quantitative Analysis*, **22** (1987), 109-126.

[14] J. Koza: Genetic programming: a paradigm for genetically breeding populations of computer programs to solve problems. *Report No.STAN-CS-90-1314*, Dept. of Compute Science, Stanford University, (1990).

[15] J. Koza: Evaluation and subsumption using genetic programming. *Proceedings of the First European Conference on Artificial Life* (MIT Press, 1991).

[16] J. Koza, *Genetic Programming* (MIT Press, 1992).

[17] B. Lebaron: Agent-based computational finance: suggested readings and early research. *Journal of Economic Dynamic and Control*, **24** (2000), 679-702.

[18] B. Lebaron, W. B. Arthur and R. Palmer: Time series properties of an artificial stock market. *Journal of Economic Dynamic and Control*, **23** (1999), 1487-1526.

[19] N. S. P. Tay and S. C. Linn: Fuzzy inductive reasoning, expectation formation and the behavior of security prices. *Journal of Economic Dynamic and Control*, **25** (2001), 321-361.

[20] L. Tesfatsion: Introduction to the special issue on agent-based computational economic. *Journal of Economic Dynamic and Control*, **25** (2000), 281-293.

[21] A. Vedat: Conditional heteroscedasticity in time series of stock returns: evidence and forecast. *Journal of Business*, **62-1** (1989), 55-80.

Shozo Tokinaga
Graduate School of Economics
Kyushu University
6-19-1 Hakozaki, Higashi-ku
Fukuoka 812-8581, Japan
E-mail: tokinaga@en.kyushu-u.ac.jp