Multi-fractarity Analysis of Artificial Stock Prices Generated by Multi-agent using Genetic Programming for Learning and its Applications

Tokinaga, Shozo
Faculty of Economics, Kyushu University : Professor

Matsuno, Seigo
Department of Business Administration, Ube National College Technology : Associate Professor

https://doi.org/10.15017/14629

出版情報：經濟學研究. 75 (4), pp.1-27, 2009-03-10. 九州大学経済学会
バージョン：

Multi-fractarity Analysis of Artificial Stock Prices Generated by Multi-agent using Genetic Programming for Learning and its Applications

Shozo Tokinaga

Seigo Matsuno

1 Introduction

In recent years, the multi-fractal formalism has been introduced to describe statistically the scaling properties of singular measures[1][2][4][7][23][24][33]-[35]. Multi-fractal analysis was first developed in the study of turbulence, and now we find various applications such as financial time series analysis. The analysis of origin of multi-fractality is seemed to be much more important than examining the existence of multi-fractalities, while these phenomena suggest us how to find anomalities in time series such as stock prices. Several conventional methods are proposed for generating multi-fractal time series, but there are no discussion about the origin of multi-fractality and related topics such as feature extractions[23][24].

In this paper, we show the multi-fractality analysis of time series in artificial stock market generated by multi-agent systems based on the Genetic Programming (GP) and its applications[23][24]. Despite many works suggesting methods to generate mono-fractal time series, there are few methods to emulate processes of multi-fractal time series. Riedi et al. proposed a method to generate multi-fractal time series based on modified wavelet-coefficients by changing the probability distribution of multipliers using the β distributions, but the relation between the probability distribution and multi-fractality is not shown[35]. Calvet et al. also suggested the generation of multi-fractal time series based on the multiplicative cascade, however, the relation between the characteristics of the multi-fractal time series and mathematical descriptions is not clear[7]. In previous works, we investigated the possibility to synthesize (generate) artificial fractal time series even chaotic ones based on multi-agent systems[10] [11][20][21]. Then, we think multi-agent systems are promising to figure out the multi-fractality revealed in artificial stock prices. In the agent system, there is a wide range of context by changing the member of agents, and then we examine the origin of multi-fractality arise out from the various simulation studies[10][11][20][21][23][24]. Moreover, the features of multi-fractal time series can be extracted by comparing time series with corresponding mono-fractal time series as counterparts. For the formalism of learning process of agents, we employ the GP which has been successfully applied to modeling of agents, analysis of chaotic dynamics and fractality, and system optimizations[9][10][11][18][19][20][21][22][23][24] [30][31][41].

There exit many conventional works related to the artificial stock market such ACE (Agent based Computational Economics) proposed by Arthur,Holland,LeBaron,Palmer,Taylor [3] [8] [12] [15] [28] [29] [36]. In contrast to conventional artificial stock markets, we treat a kind of heterogeneity of agents to some extent, by introducing five types of agents[10][20][21][23][24]. Agents are allowed to use forecast equations or decision rules which are improved based on the GP so as to ensure reproducibility of markets. Besides independent equations (rules), common ones are postulated for co-evolutionary GP learning. Within the framework, we demonstrated in previous works based on simulation studies , time series of stock prices are statistically resemble to real stock prices[10][20][21]. To investigate the origin of multi-fractality in the artificial stock prices, we emphasize the composition of agents to emulate the real market. To examine the role of agents of each type, we define six cases by changing the composition of five types of agents. For checking the multi-fractality we use an extended method based on the continuous time wavelet transform. As well as compositions of agents of each type, we investigate the effect of restriction on the selection of forecast methods. As a result, in several cases we find strict multi-fractality in artificial stock prices, and we see the relationship between the realizability (reproducibility) of multi-fractality and the system parameters.

Since there exists no way to define the multi-fractality in the time domain, as an alternative, we propose a method to extract features from the multi-fractal time series based on the approximation of mono-fractal time series derived from corresponding original one as counterparts[16][17][37][38][39]. After examining the multi-fractal processes in time series, we apply the method for feature extraction based on the prediction on fractal time series proposed in previous works. As a result, feature points of multi-fractal time series are effectively detected and extracted, and are used to figure out the roles of agents to generate multi-fractality. Then, a backward method is available to estimate the composition of agents who are seemed to be major members of stock market by examining the features of multi-fractality for stock price time series. We use conventional Multi-variate Discriminant Analysis by employing statistical package to identify dominant behavior of agents.

In the following, in Section 2, we propose multi-agent-based architecture based on the GP for modeling artificial stock market. In Section 3 we show learning of agents using the GP. In Section 4, results from simulation experiment are analyzed in details to give the condition of multi-fractality of stock price. In Section 5, feature extraction of multi-fractal time series is discussed combining with agents' behavior.

2 Multi-Agent based Modeling of Artificial Stock Markets

2.1 Problem formulation of the paper

The goal of multi-fractality analysis must ultimately be to explain causes of the multi-fractality in the time series. While many works suggesting methods to generate multi-fractal time series are valuable for the emulation of multi-fractal time series, but they do not satisfy our need to explain causes of multi-fractality.

In conventional works, authors proposed methods to generate multi-fractal time series based on modified wavelet-coefficients by changing the probability distribution of multipliers using the β distributions, and the multiplicative cascade[7][35]. However, the relation between the characteristics of the multi-fractal time series and mathematical descriptions is not clear.

We propose that the multi-fractal is a natural feature of models with forward-looking agents. Our key condition is that agents are not constrained by assumption to agree on a single expectation. Instead, we apply recent development in evolutionary system description to explain how forward-looking agents might choose among different forecasts. We assume that information arrives uniformly over time so that changes in stock price are due entirely to agents' behavior. Furthermore, our agents do not set out simply to invent among forecasts based on what they perceive to be fundamentals.

We focus on the artificial multi-agent systems for the stock market where we can observe

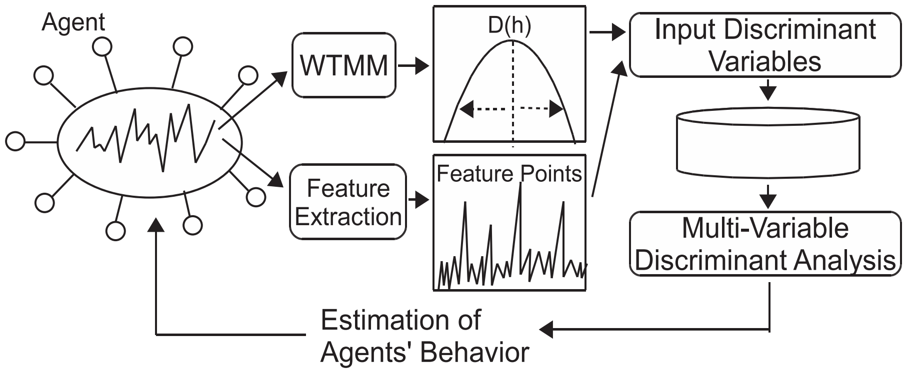

Figure 1: Schematic diagram of the problem formalization)

multi-fractality based on the agents' behavior[23][24]. In the agent system, by changing the member of agents we can examine the origin of multi-fractality. Moreover, the features of multi-fractal time series can be extracted by comparing time series with corresponding mono-fractal time series and system optimizations[7][8][11][12][13][15][16] [17][22][23][24][31].

Figure 1 show the schematic diagram of the problem formalization of the paper. At first, artificial stock price is generated as the result of trades among agents. Then, the time series of the stock price is analyzed by conventional tools such as wavelet-based multi-fractality analysis, and the feature extraction based on a prediction method of the mono-fractal time series which is the counter-part of the original multi-fractal time series. By iterative procedures by changing the composition of agents, we can accumulate sets of data representing the relation between the composition of agents and the multi-fractality of generated time series as input-output pairs. By applying the conventional method of discriminant analysis, we can estimate the discriminant functions which suggests us who are the dominant agents under the given condition (observed time series).

2.2 Architecture of Multi-agent System

The market structure is set up to be as simple as possible in terms of its economic components. It also attempts to use ingredients from existing models wherever possible. The goal of the artificial agents is to build forecasts of the future price and dividend which they will use in their demand functions. Agents do this by maintaining a list of several hypothesis, or candidate of forecasting methods.

In the following, we define the computational architecture of stock market in detail based on the system consisting of multi-agents each of which acts as a trader[10][20][21][23][24]. Specifically, five types of agents are defined in this architecture. Figure 2 shows the overview of the architecture of multi-agent system.

2.3 Agents of type 1 and type 2

The agents of both type 1 and type 2 are assumed to build forecast models for stock price and dividend of the next period by using arithmetic expressions which are called "forecast equations" including state variables of the market as operands. For example, an agent has an equation to estimate the stock price p_{t+1} in the next period as

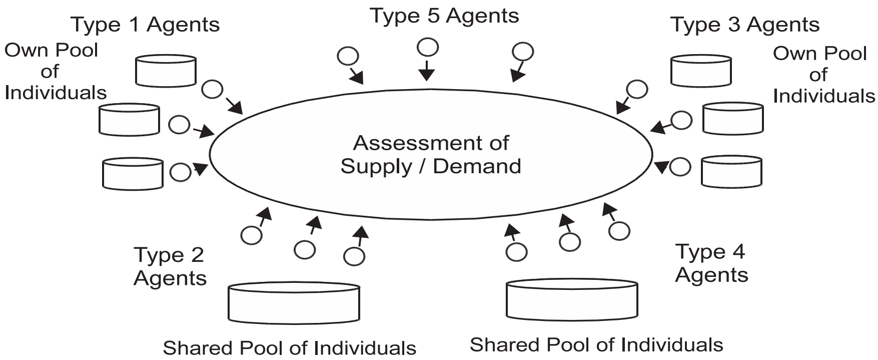

Figure 2: Overview of architecture of multi-agent system

```
price(1)+dividend(1)+(price(1)-price(2))*1.05+(dividend(1)-dividend(2))*1.05
```

where the variables such as price(1) and dividend(2) are defined by using past trades.

The difference of these two types comes from the way to share the forecast equations among agents to predict future prices. Agents of type 1 usually use their own forecast equations for predicting stock price. On the other hand, agents of type 2 only use a common pool of forecast equations to map state into forecasts without their own forecast equations. We assume that agents of type 1 are allowed to have individual 50 forecast equations for each, and agents of type 3 share 50 forecast equations.

However, we also assume that besides agents of type 2, the agents of type 1 may also make a decision to access this the common pool of the forecast equations when they feel unsatisfied with the performances of their own forecast equations. While they may appreciate common forecast equations than their own equations if they meet sudden fall of wealth. The agents are allowed to learn and change their behavior by altering this set of the forecast equations by eliminating poorly performing equations and adding new ones through the use of GP oriented for the forecast (arithmetic) equations.

The agent models include the agents' objectives and their adaptation, and these frameworks have been used in many conventional researches for artificial stock market, for example, in works by Chen et al. and LeBaron et al.[8][28]. Now we show the behavior of agents by introducing their utility functions. For simplicity, we assume that all agents using forecast equations share the same utility function. More specifically, this function is assumed to be a Constant Absolute Risk Aversion (CARA) utility function.

$$U(W_{i,t}) = -\exp(-\lambda W_{i,t}), \tag{1}$$

where $W_{i,t}$ is the wealth of agent i at time period t, and λ is the degree of relative risk aversion. Agents can accumulate their wealth by making investments. There are two assets available for agents to invest. One is the riskless interest-bearing asset (money), and the other is the risky asset (stock). In other words, at each time point, each trader has two way to keep the wealth as

$$W_{i,t} = M_{i,t} + P_t h_{i,t}, \tag{2}$$

where $M_{i,t}$ and $h_{i,t}$ denote the money and shares of the stock held by agent i at time t. Given this portfolio $(M_{i,t}, h_{i,t})$, the agent's total wealth $W_{i,t+1}$ is thus

$$W_{i,t+1} = (1+r)M_{i,t} + h_{i,t}(P_{t+1} + D_{t+1}), \tag{3}$$

where P_t is the price of the stock at time period t, D_t is per-share cash dividends paid by the companies issuing the stocks, and r is the riskless interest rate. We assume D_t can follow a stationary stochastic process having i.i.d. normal process which is not known to agents. Here, we assume that the divided follows to a process defined by itself rather than stock prices, while the evaluation of another factors such as the profit of firms which is reflected to the stock price is very difficult in the agent simulation scheme treated here. Given this wealth dynamics, the goal of each agents is to myopically maximize the one-period expected utility function as

$$E_{i,t}(U(W_{i,t}) = E[-\exp(-\lambda W_{i,t})|I_{i,t}], \tag{4}$$

subject to the condition shown in equation (3), where $E_{i,t}(.)$ is the conditional expectation of wealth $W_{i,t+1}$ by the agent i under the information up to t (the information set is denoted as $I_{i,t}$.

It is well known that under CARA utility and Gaussian distribution for forecast, the desired demand $X_{i,t}$ by the agent i for holding share of risky asset is linear in the expected excess return, which is derived by Grossman and Stiglizt[14].

$$X_{i,t} = \frac{E_{i,t}(P_{t+1} + D_{t+1}) - (1+r)P_t}{\lambda \sigma^2_{i,t}}, \tag{5}$$

where $\sigma^2_{i,t}$ is the conditional variance of $P_{t+1} + D_{t+1}$ under the given $I_{i,t}$. The amount $X_{i,t}$ means the appropriate share estimate by the agent i. The formula is also well known, and used in real trades[5].

Now we address the agents' expectation $E_{i,t}(P_{t+1} + D_{t+1})$ and $\sigma_{i,t}$. It must be noted that the function $E_{i,t}(.)$ is generated by the GP procedure in the paper. But in this case, the range of $E_{i,t}(.)$ can hardly be held in check, and that can potentially make the price dance. Additionally, one may be interested in the forecasting increments, instead of level. Therefore, we consider following specific function form for $E_{i,t}(.)$.

$$E_{i,t}(P_{t+1} + D_{t+1}) = (P_t + D_t)(1 + \tanh(\theta_1 \dot{f}_{i,t})), \tag{6}$$

which is also motivated by the martingale hypothesis in finance. The function $f_{i,t}$ generated by the GP procedure is fed into a hyper-tangent transformation and multiplied by a constant θ_1. The equation (6) can be interpreted as a forecast of the growth rare (rate of return) of $P_t + D_t$.

The virtue for this function form is that, if $f_{i,t} = 0$, then the agent actually validates the martingale hypothesis. Since D_t follows as i.i.d process with constant mean value, the best forecast for tomorrow's price is today's price which is known as the martingale hypothesis in finance. Assume that martingale hypothesis is consistent with microbeliefs of agents. Then, alternatively speaking, on the top of the market is efficient in the martingale sense, and on the bottom agent i is also believes. Therefore, from the cardinality of set $\{i|f_{i,t} = 0\}$ denoted by $N_{1,t}$, we can know how well the efficient market hypothesis is accepted among agents. The population of functions $f_{i,t}, (i = 1, 2, ..., N)$ is determined by the GP procedure proposed in the paper. As to the subjective risk equation, we take a modification of the equation originally used by Arthur et al [3].

$$\sigma^2_{i,t} = (1-w)\sigma^2_{t-1|n_1} + w[(P_t + D_t - E_{i,t}(P_t + D_t))^2], \tag{7}$$

where

$$\sigma_{t|n_1} = \sum_{j=0}^{n_1-1} (P_{t-j} - \bar{P}_{t|n_1})^2/(n_1 - 1). \tag{8}$$

In other words, $\sigma_{t-1|n_1}$ is simply the historical volatility based on the past n_1 observations.

Table 1: Table 1-Variables calculated from past stock prices

variables	definition
min(t)	minimum price in past t periods till now
max(t)	maximum price in past t periods till now
av(t)	average price in past t periods till now
price(t)	stock price at t past time from now
dividend(t)	dividend at t past time from now

Table 2: Table 2-Exampes of forecast equations used by agents

$f_{i,t}$=1.0089 * sqrt(min(5)) + 1.0018 * av(4) + (7.8802 / sqrt(min(5)) - 4.714) / abs(price(7)-price(8)) + price (1)
$f_{i,t}$=1-1.8618 / (price(2) - price(3)) -(3.3443 - 1.7301 * sqrt(dividend(6))) * dividend(5) + exp(dividend(4)) / max(10) + price(1)
$f_{i,t}$=price(1)+ 0.9020* ((dividend(2) + 4.3850) / min(3) - price(3) + price(4)) / min(4)

In the GP, forecast equations are represented in the tree structure (called individual)[9]-[11][18]-[27][30][31][41]. In the parse tree, the non-terminal node is taken from the function sets, containing, +,-,÷,×, exp, abs ,sqrt.

The operands for these functions (operations) are taken from past stock prices themselves, constants and variables listed in Table 1 which are obtained by simple arithmetic calculation for the past stock prices. Table 2 shows examples of forecast equations used by agents of type 1 and 2 when $\theta_1 = 1.0$. At the end of trading, agents update the accuracy of all matched forecast equations according to the forecast error. We define the fitness as the accuracy of forecast equations on historical datum. Specifically, the fitness of a forecast equation is defined by the inverse 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 pool of individuals of GP, and t accounts for time horizon. As is previously described, in this paper, this squared forecast error $e^2_{i,j,t}$ of selected forecast model as the variance $\sigma^2_{i,t}$ used for the estimation of share demand $X_{i,t}$ by agents of type 1 and type 2.

If we have sufficient time of trades for learning, the record of forecast error $e_{i,j,t}$ is available for evaluating forecast equation j of agent i. Then, the evolutionary procedure based on the GP is applied to improve forecast equations in the pool of individuals of type 1 and type 2 agents.

2.4 Agents of type 3 and type 4

Moreover, in reality the complexity of the market forces agents to act inductively by using simple rules of thumb. Usage of these simple trading rules to guide buy or sell decision seems to be effective at some time. We assume that agents of type 3 and 4 use the production rules for trading stocks (called forecast rules). These agents forecast only the rise/fall of stock prices based on the decision rules, and the volume to be traded is assigned at random. The decision rules are also represented as individuals, and 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

Table 3: Exampes of forecast rules used by agents

buy if	av(8)$\leq$ price(2) XOR (dividend(6) $\geq$ dividend(10) AND (price(3) $<$ max(6) OR min(8) $\neq$ price(7)))
buy if	min(9) = price(9)OR (price(8) $\geq$ av(8) AND (price(10) $\geq$ max(10) OR max(2) = max(6)))
sell if	av(10) $>$ av(6)OR (dividend(2) $\leq$ (price(9)-price(10)) AND (price(6) $\leq$ max(2) OR dividend(10) $\neq$ av(6) OR min(5) = price(2)))
sell if	dividend(3) $<$ (price(1)-price(2)) AND min(5) $\neq$ min(7)

part). For example, an agent may has a forecast rile such as

```
if price(1) > av(10) and  price(2) < max(10) then buy
```

The difference of these two types exists in the way to use the common decision rules. It is assumed that the agents of type 3 possess their individual decision rules, but the agents of type 4 only learn from a common pool of decision rules for trading without their own rule bases. Similarly, agents of type 3 are allowed to access this common decision rules when they feel little satisfaction with the performances of their own decision rules. We assume that agents of type 3 are allowed to have individual 50 decision rules for each, and agents of type 4 share 50 decision rules.

Agents of type 3 and 4 are allowed to dynamically decide to sell/buy stocks connected with decision rules. Agents prefer using simple rules of thumb to decide whether to trade or not. However, they do not decide optimal quantities to trade exactly, but rather at a stochastic quantity. These decision rules are identical to conventional production rules where if the condition is satisfied then the behavior of action part will be implemented.

Similar to the forecast equations, we define the condition part of decision rules as logical expressions which is represented as a connection of predicates with logical operators, including AND, OR and XOR(exclusive OR). The predicates can be defined as a combination of two arithmetic expressions (equations) given by the prefix representations like forecast equations connected by using comparative operators, including $>, \geq, <, \leq, =, \neq$. Table 3 shows example of decision rules used by agents of type 3 and 4. Since the condition parts of decision rules possesses the same structure as forecast equations (parse tree) by changing the arithmetic operators by logical operators and operands by predicates, we assume that the logical expressions are also represented as individuals in the GP. To simplify the system configuration, we assume that the condition part of a decision rule consists of logical expressions which are represented by a single logical operator and two predicates (a combination of binomial logical expressions).

The relations between the logical expression and predicates are realized by a hierarchical data structure with two level of pool of individuals. The overall structure of logical expression is stored in the first level, and the links to combine the predicates stored in the second level. They 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,XOR, while the index of a predicate 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.

The crossover and mutation operations for the logical expression are the same as the GP in arithmetic expression by replacing the operators and operands suitable for logical expressions.

2.5 Agents of type 5

As another related modeling of agents different from agents of types 1 through 4, we assume agent of type 5 who make decisions which are less than perfectly rational. Agents of type 5 decide at random when they should sell/buy stocks and how much they should trade. Then, agents of type 5 have different characteristics, and behave like speculators.

(1) probability of taking behaviors

Agents of type 5 take their behavior in one of three decisions, namely, they sell, buy stocks and wait (do nothing). The probability of these three decision making is assumed to be same (at the probability of 1/3).

(2) volume to trade

If agents sell (buy) stocks, then they determine the volume to trade from a random number uniformly distributed between 1 and 5.

(3) restriction on trades

If agents have not enough money to buy stocks at the volume they wish, then the investments are restricted less than the money at hand. Similarly, if agents have not enough stocks to sell at the volume they wish, then the trades are restricted less than the volume at hand.

2.6 Probabilistic decision

It may be general behavior that agents use forecast equations or decision rules with highest fitness (ability) in their own knowledge bases. However, evidences of volatility persistence in financial markets suggests us that this kind of deterministic scheme lead agents frequent change of trades, and it results in poor strategies. Therefore, a kind of restriction is introduced on the selection of forecast methods. We assume that agents of type 1 through 4 decide their trades by using one of individuals at random (not deterministic) from the pool which have relatively larger fitness, and we call the strategy as the probabilistic decision. The probabilistic decision corresponds to the conventional method of LCS (Learning Classifier System) in which agents select prediction or rule by activating appropriate clusters promising relatively better results[6].

2.7 Assessing Stock Price

Once agents have settled on forecast equations and rules, they then substitute forecast parameters into their demands for shares. Generally, there is little possibility that the total number of stocks demanded by agents will equal the total number of stocks supplied by agents exactly. Now it is trivial exercise to find a price which clears the markets by balancing the demand to the fixed supply of shares.

Stock prices to clear the market are set endogenously. Since 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 usually used[8]. Let O_t be the number of shares which agents would like to submit at period t, and let B_t be the number of shares which agents would like to offer to sell at time t. Then, the stock price can be adjusted according to the following equation, in which the function $\beta(.)$ is defined by the difference between total demand and supply.

$$P_{t+1} = P_t[1 + \tanh(\beta(B_t - O_t))], \tag{9}$$

where the function $\beta(B_t - O_t)$ means the transformation function of demand into the stock price, and P_t, P_{t+1} are the stock prices at time $t, t+1$ in the artificial market. For simplicity, the function has the same form independent from cases where the demand is greater than the supply or not[8].

3 GP procedure for learning

3.1 Basics of the GP

The GP is an evolutionary search process to generate structures of arbitrary functions, where each individual in the population (pool of individuals) is a computer program composed of the arithmetic operations, standard mathematical expressions and variables rather than binary bits in the GA (Genetic Algorithm)[6][15]. For simplicity, estimation problems of system equation generating time series are explained.

In the GP, the system equations are represented in parse tree structures. The prefix representation corresponding to the parse tree is approved due to simple operational performance rather than pointer based implementation and the postfix approach. In the parse tree, non-terminal nodes are taken from some well-defined functions such as binomial operation $+, -, \times, /$, and the operation taking the square root of variable. Terminal nodes consist of arguments chosen from set of constant and variable. The pool of variable consists of the variable x_t itself and the time lag of x_t such as x_{t-1}.

The prefix representation follows traditional representation by using the Lisp syntax. For example, if the function is given as $x_t = (3 \times x_{t-1} - x_{t-2}) \times (x_{t-3} - 4)$ then, we have the next prefix representation.

$$\times, -, \times, 3, x_{t-1}, x_{t-2}, -, x_{t-3}, 4. \tag{10}$$

The ability (called fitness) of underlying individual corresponding to a forecast equation is evaluated by comparing the the observation x_t with resulting prediction $\hat{x}_t$ obtained by substituting lagged values such as x_{t-1} in the equation. The fitness of individual is defined by the inverse of prediction error between $\hat{x}_t$ and x_t.

For checking the validity of underlying parse tree (individual), the so-called stack count (denoted as *StackCount* in the paper) is useful[25][26][27]. Assume scanning process of a prefix representation from left to right using the counter. The *StackCount* is the number of operands it places on the counter minus the number of operators taken off from it. The cumulative *StackCount* never becomes negative until we reach the end at which point the overall sum still needs to be -1. The *StackCount* is used for checking syntactical validity of initial population of individuals. Usually, we generate initial population by using random numbers, and if the final count is -1, then the individual is acceptable. The *StackCount* is also available for finding relevant crosspoints in crossover operations.

Crossover operations

Two genetic operations are used in the reproduction phase: crossover and mutation. The crossover operation selects two individuals from the population with the probability proportional to fitness. Then, choose one crosspoint on each individual, and these two subtrees (segments of individuals) are extracted and swapped with each other reproducing two offsprings. Different from the operation in GA, the crossover operation in the GP is applied to restricted cases so as to ensure syntactical validity of offsprings. Then, we can not choose arbitrary loci (crosspoints) on the individuals in swapping subtrees.

To keep the crossover operation always producing syntactically and semantically valid programs, we look for the nodes which can be a subtree in the crossover operation and check for no violation. By using the *StackCount* already mentioned, we ensure the valid crossover operation. The basic rule is that any two loci on the two parents genomes can serve as crossover points as long as the ongoing *StackCount* just before those points is the same. The crossover operation creates new offsprings by exchanging sub-trees between two parents.

Mutation

The goal of the mutation operation is the reintroduction of some diversity in an population. In the mutation operation, we replace a part of the tree by another element. Select at random a locus in a parse tree to which the mutation is applied, the we replace the place by another value (a primitive function or a variable).

3.2 Algorithm of the GP

We iteratively perform the following steps until the termination criterion has been satisfied.
(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 function.
(Step 2)
Evaluate each individual (obtaining value of forecast equation) in population by substituting values of variable and constants included in the individual. Then, assign resulted fitness value as a partial credit for getting closer solution to the correct output.
(Step 3)
Select a pair of individuals chosen with a probability p_i based on the fitness S_i. The probability p_i is defined for ith individual as follows.

$$p_i = (S_i - S_{min})/\sum^{N}(S_i - S_{min}). \tag{11}$$

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. Then, we gather these new offsprings in the pool P-B which is different from the initial pool P-A. Iterate the procedure several times, and we gather sufficient number of new offsprings necessary for the replacement of individuals. Then, we replace individuals in the pool P-A having lower fitness by individuals in the pool P-B.
(Step 5)
At a certain probability, we apply the mutation operations for the pool of individuals. 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 Multi-fractality of Stock Prices

4.1 The multi-fractality formalism

Now, we formalize the multi-fractality by following comprehensive description by Arneodo, Bacry and Muzy[1][33]. Usually, in dealing with fractal objects on which a measure μ is defined, the dimension D is introduced to describe the increase of the mass $\mu(B_t(\epsilon))$ with size ϵ. where $B_t(\epsilon)$ is the ball centered at the time t and of size ϵ.

$$\int_{B_t(\epsilon)} d\mu(y) \sim \epsilon^D. \tag{12}$$

In general, however, fractal measures display multi-fractal properties in the sense that they scale differently from point to point. Then we consider a local scaling behavior such as.

$$\mu(B_t(\epsilon)) \sim \epsilon^{\alpha(t)}, \tag{13}$$

where the exponent $\alpha(t)$ represents the singularity strength of the measure μ at the point t. The smaller the exponent $\alpha(t)$, the more singular the measure around x and the stronger the singularity. The singularity spectrum $f(\alpha)$ describes the statistical distribution of the singularity exponent $\alpha(t)$. If we cover the support of the measure μ with the balls of size ϵ, the number $N_\alpha(\epsilon)$ of such balls that scale like ϵ^α for a given α, behaves like

$$N_\alpha(\epsilon) \sim \epsilon^{-f(\alpha)}. \tag{14}$$

In the limit $\epsilon \to 0$, $f(\alpha)$ is defined as the Hausdorff dimension of the set of all points t such that $\alpha(t) = \alpha$.

$$f(\alpha) = d_H[x \in supp\ \mu, \alpha(x) = \alpha]. \tag{15}$$

The generalized fractal dimensions D_q, which correspond to scaling exponents for the qth moment of the measure μ, also provide us an alternative description of the singular measures. If we cover the support of μ with the boxes $B_i(\epsilon)$ of size ϵ, one can define a series of exponents $\tau(q)$ from the scaling behavior of the partition function.

$$Z(q,\epsilon) = \sum_{i=1}^{N(\epsilon)} \mu_i^q(\epsilon), \tag{16}$$

where $\mu_i = \mu(B_i(\epsilon))$. In the limit $\epsilon \to 0$, $Z(q,\epsilon)$ behaves as a power law.

$$Z(q,\epsilon) \sim \epsilon^{\tau(q)}. \tag{17}$$

The spectrum of generalized fractal dimensions is obtained from the knowledge of the exponents $\tau(q)$ by the following relation.

$$D_q = \tau(q)/(q-1). \tag{18}$$

For certain values of q one can recognize well know quantities. D_0 corresponds to the capacity (box) dimension, and D_1 characterizes the information dimension. Moreover, for integer $q \geq 2$, the D_q can be related to the scaling behavior of $q-$point correlation integrals. At the scale ϵ if we consider that the distribution of the α is of the form $\rho(\alpha)\epsilon^{-f(\alpha)}$ and if we use the expression in equation (16), then it follows.

$$Z(q,\epsilon) \sim \int \rho(\alpha)\epsilon^{q\alpha - f(\alpha)} d\alpha. \tag{19}$$

In the limit $\epsilon \to 0$, this sum is dominated by the term $\epsilon^{\min(q\alpha - f(\alpha))}$. Then, from the definition of $\tau(q)$, one obtains

$$\tau(q) = \min_\alpha(q\alpha - f(\alpha)). \tag{20}$$

Thus, the $\tau(q)$ spectrum and in turns the D_q is obtained by the Legendre transformation of $f(\alpha)$ singularity spectrum. If $f(\alpha)$ and D_q are smooth functions, then we obtain following relations.

$$q = df/d\alpha, \quad \tau(q) = q\alpha - f(\alpha). \tag{21}$$

4.2 Wavelet Transform Modulus Maxima method

In the following discussion, we use the Wavelet Transform Modulus Maxima (WTMM) method proposed by Muzy, Bacry and Arnedo for finding the multi-fractal processes in time series[1][33]. As shown in previous section, we have a natural generalization of the classical box counting techniques to fractal signals. Then, the wavelets play the role of generalized boxes, and the wavelet transform reveals as a mathematical microscope that can be further used to extract microscopic information about the scaling properties of the fractal objects. The wavelet transform of a function consists in decomposing it into elementary space-scale contributions, associated to the so-called wavelets which are constructed from one single function, the analyzing wavelet.

At first, we calculate the wavelet transform of the time series $g(t)$ as follows.

$$z(b,a) = \frac{1}{a}\int_{-\infty}^{\infty} \phi(\frac{t-b}{a})g(t)dt, \tag{22}$$

where $\phi((t-b)/a)$ is obtained from the basic wavelet function $\phi(t)$ by using the dilation index (or called the scale parameter) a and the translation index (or called the space parameter) b. The analyzing wavelet $\phi(.)$ is generally chosen to be well localized in both space and frequency. For particular purpose of tracking singularity interested here, the function $\phi(.)$ is required to be orthonormal to some lower order polynomials.

$$\int_{-\infty}^{\infty} t^m \phi(t)dt = 0, \quad 0 \leq m < N. \tag{23}$$

$$\phi^{(N)}(t) = \frac{d^N}{dt^N} e^{-t^2/2)}. \tag{24}$$

The Hölder exponent $h(t_0)$ of a function g at t_0 is defined as the largest exponent such that there exists a polynomial $P_n(t)$ of order n that satisfies

$$|g(t) - P_n(t-t_0)| = O(|t-t_0|^h), \tag{25}$$

for t in a neighborhood of t_0. The local behavior of g is mirrored by the wavelet transform which locally behaves like

$$z(t_0, a) = O(a^{h(t_0)}), \tag{26}$$

in the limit $a \to 0$, we have $N > h(t_0)$ which satisfy the equation (26). The above equation mainly says that in investigating the local scaling behavior of the wavelet coefficients computed with an analyzing wavelet whose N first components vanish, then one can generally detect all the Hölder exponents of g that are smaller than N.

The originality of the WTMM method consists in building partition function from the modulus maxima of the wavelet transform. These maxima are defined at each scale a, as the local maxima of $z(t,a)$ considered as a function of t. These wavelet transform modulus maxima are disposed on connected curves called maxima lines. Let us define $L(a_0)$ as the set of all the maxima lines l that satisfy

An important feature of these maxima lines is that, each time the analyzed signal has a local Hölder exponent $h(t_0) < N$ at the point t_0, and there is at least one maxima line pointing towards t_0, along which the equation (26) holds. In the case of fractal signals which are typically characterized by a hierarchical distribution of singularities, we expect that the number of maxima lines will diverge in the limit $a \to 0$. The WTMM has an advantage of the space-scale partitioning given by this skeleton to define the following partition function.

$$G(a,q) = \sum_{l \in L(a)} [sup_{(t,a') \in l} |z(t,a')|]^q. \tag{27}$$

The analyzing wavelet $\phi(.)$ can be seen as a box of a particular shape where the scale a is its size, while the modulus maxima indicate how to position our special boxes to obtain a partition at the considered scale.

After these calculation of maxima, we obtain the exponent $\tau(q)$. In the limit $a \to 0$, one can again define the exponent $\tau(q)$ for the power law behavior of the partition function.

$$Z(a,q) \sim a^{\tau(q)}. \tag{28}$$

Then, we can compute the singularity spectrum $D(h)$ from the Legendre transform of $\tau(h)$.

$$D(h) = \min_q [qh - \tau(q)]. \tag{29}$$

The singularity spectrum $D(h)$ of the signal is then defined as the function that gives, for a fixed h, the Hausdorff dimension of the set of points of x where the exponent $h(x)$ is equal to h.

The value h can have also negative values under the extension of definition about the Hölder exponent. Therefore, we depict the graphs also for negative h based on the WTMM. As Muzy and Bacry showed, to figure out much stronger singularities we need to extend the definition of $D(h)$ to tempered distributions. Roughly speaking, a distribution $f(.)$ has, at x_0, a Hölder exponent $h(x_0)$, if its primitive (in the sense of distributions) has, at the same point, a local Hölder exponent $h(x_0)+1$, then the Hölder exponent can also take on negative values[1][33].

The open software reliable for the WTMM procedure described above is now available through the Internet, and see[13].

4.3 Simulation scheme

To investigate the origin of multi-fractality in the artificial stock prices, the composition or the ratio of agents of each type may play an important role. Then we prepare several cases of combination of the number of agents for each type. Table 4 summarizes these cases depending on the number of agents where N_i denote the number of agents of type i. These cases are used for the scheme of simulation studies.

Compositions and consistency

Theoretically, we have 32 cases of composition of five types of agents, however, we restricted ourselves to cases listed in Table 4 based on our previous works[10][11][20]-[22][24]. The main reason to omit other cases depends on the monotonous characteristics of generated stock price. If we exclude type 3 and 4 agents from the system, then the stock price becomes to be less fluctuating and converged to a single price. Similarly, if we use only type 3 and 4 agents, then the stock price becomes to be unrealistic. For these reasons, we assume that at least one type of agents must be included from the pool of agents using forecast equations (type 1 and 2), and the pool of agents using forecast rules (type 3 and 4). Off course, the stock price bearing monotonic characteristics is also to be discussed, however, we are now mainly interested in the reality of stock price, and the problem will be treated as another issues. Then, as in Table 4 we choose the compositions between type 1-type 3 agents, and type 2-type 4 agents, as well as the case including all types of agents.

As alternatives, we can test the compositions where agents using their own pool of individuals (self-learning) and the agents using common pool of individuals (social learning). We call these cases as intermediate cases. In these cases one group of agents consist of agents using forecast equations, and another group of agents consists of agents using forecast rules. However, it is also observed that if all agents utilize their own individuals (self-learning) then the stock price changes abruptly, since they can not improve their decisions (individuals) through social learning. Then, the social learning using common pool of individuals play a role to mitigate the abrupt changes of price. From these reasoning, it is seen that these intermediate cases where agents using self-learning are mixed with agents using social learning are located between extreme cases

Table 4: Cases depending on the numbers of agents

	Case1	Case2	Case3	Case4	Case5	Case6
N_1	100	100	0	100	0	0
N_2	100	0	100	0	100	0
N_3	100	100	0	100	0	0
N_4	100	0	100	0	100	0
N_5	100	100	100	0	0	100

where all agents utilize their own individuals (self learning). In Table 4, these intermediate cases are located between Case 2 and Case 3, and also between Case 4 and Case 5. The behavior of agents in these intermediate cases will be estimated from Case 2 through Case 5, and therefore intermediate cases are excluded from Table 4.

As is seen from Table 4, the total number of agents depends on cases, and it is expected that the activity of the stock market is different from each other. Roughly speaking, if the number of agents becomes larger, then the range (difference between the maximum and minimum of the stock price) becomes larger. As far as we focus on the multi-fractality of the stock price itself, the difference of the activity of the market is not serious. However, if we use the stock price and related features for the discriminant analysis to identify the dominant agents in the market, the range of the stock price may affect the analysis. Therefore, we adjust the range of stock price in the analysis discussed later, and avoid the direct effect of ranges of stock prices.

Parameters in simulation studies

We assume following parameters for simulation studies.

- number of individuals: agents of type 1,3 have 50 own individuals, agents of type 2 and 4 have 50 shared individuals
- size of array of individual: length of individuals is selected to be 30
- market parameters:$\lambda = 0.01$, $\beta = 0.0005$ and $r = 0.01$
- record of stock prices: 10,000 samples
- random numbers: generated by using the library of computing facility

Agents are allow to make pre-processing and learning for 3,000 time periods where no stock is delivered prior to entering into the stage of active trades. After learning period, agents trade stocks based on their own forecasts, however, under the predefined budget restriction. Agents are also restricted to trade a maximum of 5 shares per trading. Moreover, besides doing trades, individuals in common forecast equations and decision rules for agents type 2 and 4 are simultaneously updated every 10 periods (trades). Similarly, agents of type 1 and type 3 prefer updating their own individuals more frequently, namely, once per 5 periods.

Besides the composition of agents of each type, we investigate the effect of restriction on the selection of forecast methods (referred as probabilistic decisions) made by agents. We define p_S as the probability in selecting an individual from the pool. We assume agents of type 1, 2, 3 and 4 use one of the individuals at random for forecast of prices (or decision making) by using one of N_S individuals having relatively large fitness which are embedded in the whole pool of individuals. The probability p_S is therefore defined as $p_S = N_S/N_G$ where N_G is the number of individuals in the pool. If $p_S = 1$ the method for the usage of individual follows random decision rather than the probabilistic decision. If $N_S = 1, p_S = 1/N_G$, then the method corresponds to the exclusion scheme in which agents use only individuals with highest fitness.

4.4 Simulation results

In the following, we show the simulation results for checking the multi-fractality of each case. Figure 3 and 4 show examples of $\tau(q)$ and $D(h)$ for each case of artificially generated stock price. Even though these figures are only typical examples for each case, but we have very similar results for another simulation studies. Therefore, we can reduce several important characteristics of multi-fractality corresponding to each case.

Since the $D(h)$ corresponds to the Household dimension of the singularity with the exponent h, and $D(h)$ is evaluated as the fractal dimension of the support of time axis with Hölder exponent h. Then, it is concluded that if the value of $D(h)$ can reach the value one at $h = h_0$, then time series possesses the singularity with a certain value of h_0 all through the time axis. Otherwise the value of $D(h)$ does not reach one.

(1) Existence of multi-fractatily seen from Figure 3

It is well known that if the lines of $\tau(q)$ are composed two straight lines connected around $q = 0$ (piecewise linear), then the time series bears multi-fractality. On the other hand, if the line of $\tau(q)$ is straight, then the time series bears mono-fractality. As is seen from Figure 3, for Case 1 through Case 5 the lines of $\tau(q)$ are composed two straight lines connected around $q = 0$ (piecewise linear). On the other hand, the line of $\tau(q)$ for Case 6 is almost straight. Then, it is concluded that stock price time series bear nonlinearity in $\tau(q)$ except for Case 6, and we see multi-fractality for Case 1 through 5. Additionally, we see the line of $\tau(q)$ of case 6 is slightly curve, and a little deviation from mono-fractality is observed.

(2) Existence of multi-fractatily seen from Figure 4

Similar results about the existence of multi-fractality are obtained from Figure 4 as well as in Figure 3. In the graphs of $D(h)$ for Case 1 through Case 5 we see relatively spread curves compared to that of Case 6. If the time series follows the Brownian motion of the fractal dimension $D = 1.5$, then the value of $D(h)$ becomes to be one (maximum value) only at $h = 0.5$ (single point). When we see $D(h)$ for Case 6 in the figure, $D(h)$ becomes to be one (maximum value) at $h \approx 0.5$. The fact implies us that the time series has the singularity with $h = 0.5$ all through the time axis, and the time series has characteristics resemble to the Brownian motion. However, the shape of $D(h)$ for Case 4 and Case 6 is not sharp enough compared to the ideal (theoretical) shapes of $D(h)$ of Brownian where the shape of $D(h)$ is reduced to a single point. The restriction comes from the fact that the volume of stock and the assets of agents in the artificial are limited, and therefore the stock price does not necessarily follow free movement which is originally expected. But, this kind of limitation is also observed in real markets, and we see no ideal Brownian motions in stock markets. Then, we can conclude that the stock price time series for Case 1, 2, 3 and 5 bear strict multi-fractality.

(3) Negative correlation of time series

As is seen from the figure, In Case 6 even though the width of spread of the Hölder exponent h in $D(h)$ is small but still observed, then the time series bears multi-fractality, however it is not strict. Also for Case 4, $D(h)$ exhibits its maximum value at $h = 0.18$, and the spread of $D(h)$ is limited less than 0.5. The fact implies us that the artificial time series in this case has a kind of negative correlation with past values.

It is proved if the point of h where the value of $D(h)$ becomes to be maximum is small enough, then the time series is regarded to have a kind of negative correlation with past values. By looking corresponding figures of time series generated for Case 1 through Case 5, negative

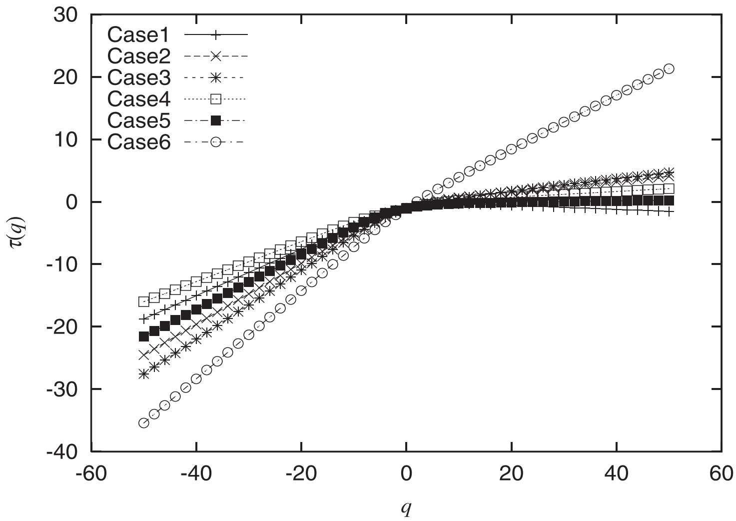

Figure 3: An example of $\tau(q)$

correlations are found in each case. One reason for the existence of negative correlations comes from the agents' behavior where agents using forecasts tend to expect future prices in inverse direction. With respect to current movement, namely, current rise (fall) of price induces agents' forecasts of fall (rise) of price. Additionally, if the system includes agents of type 2 and 4 who use common forecast equations (rules), then there are many agents taking a similar decision and behavior.

(4) Multi-fractality and agents' behavior

In summary, in Case 6 where agents take random decision for sell/buy decision the stock price even bears multi-fractality, however it is not strict, and is resemble to a Brownian motion. Similarly, this kind of deviation form multi-fractality is also observed in stock prices of Case 4 where all agents make decision by using the prediction functions. The fact means that even though agents follow a kind of "rational" decision and behavior, but the randomness of stock prices is also found as a result of interactions among agents. Then, these rational behaviors also contribute to reproducibility of artificial stock prices which are statistically resemble to real ones (following Brownian motions).

4.5 Multi-fractality under probabilistic decision

Now, we explain the simulation results for checking the multi-fractality for the cases of probabilistic decision within the Case 1.

As is referred as probabilistic decision made by agents, we define p_S as a probability in selecting an individual from the pool. We assume agents of type 1,2,3,4 use one of the individual at random from N_S individuals which are selected from N_G members in the pool. The probability p_S is therefore defined as $p_S = N_S/N_G$. If $p_S = 1$ the method for the usage of individual is a random decision. If $N_S = 1, p_S = 1/N_G$, then the method corresponds to the exclusion scheme in which agents use only individuals with highest fitness. We consider the cases for $p_S = 1/N_G, 0.1, 0.2, 0.3, 0.4, 0.5$.

Figure 5 and 6 show examples of $\tau(q)$ and $D(h)$ for each case of artificially generated stock

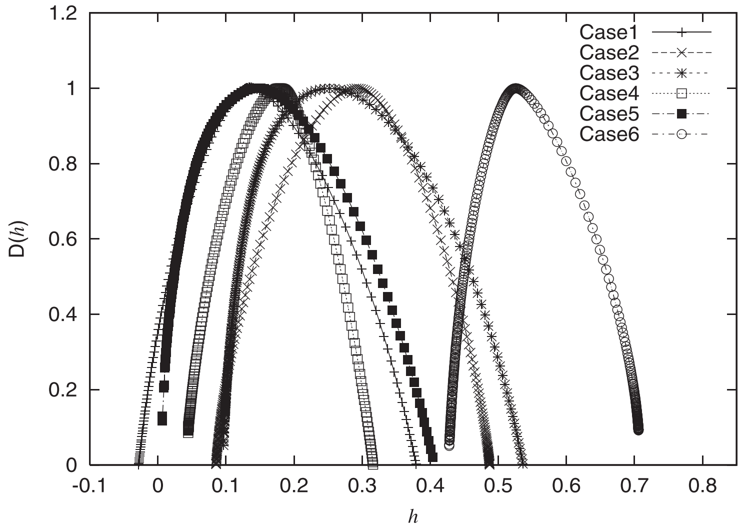

Figure 4: An example of $D(h)$

price. Even though these figures are only typical examples for the cases, but we have very similar results for another simulation studies[23][24]. Therefore, we can reduce several important characteristics of multi-fractality corresponding to each case.

(1) Existence of multi-fractatily seen from Figure 5

As is seen from Figure 5, for each case the lines of $\tau(q)$ is composed two straight lines connected around $q = 0$ (piecewise linear). Then, it is formally concluded that stock price time series bear multi-fractality.

(2) Existence of multi-fractatily seen from Figure 4

In the graphs of $D(h)$ for each case we see relatively spread curve. However, according to the analysis based on the MTMM, if the curve of $D(h)$ does not reach to the value one as the maximum, it is concluded that the time series is characterized as the nonsingular one. Therefore, if $Ps = 1/N_G, 0.1, 0.2, 0.3$ the agents' behavior is restricted in some range, and the generated time series bears no more fractality (multi-fractality as well as mono-fractaility).

(3) negative correlations of time series

The time series generated for cases $p_S \geq 0.3$ have a kind of negative correlations. However, the spread of corresponding curve of $D(h)$ is not large, then it is concluded that even though the time series are regarded as a multi-fractal time series, but they deviate from strict multi-fractality.

(4) Multi-fractality and agents' behavior

In summary, even though we assume the random behavior of agents, for small p_S where the range of decision making of agents is restricted, the time series becomes to be unusual pattern. Then, the simulation scheme in these cases are not relevant to examine the artificial market. Additionally, since agents make their decision on a deterministic prediction function or production rules for $p_S \leq 0.2$, the time series reveals as a monotonic trend or the periodical change. On the other hand, if we choose large p_S, then the agents' behavior become to be random enough and the time series is still mono-fractal, but they deviate from strict multi-fractality. In this case there is no meaning to organize artificial market.

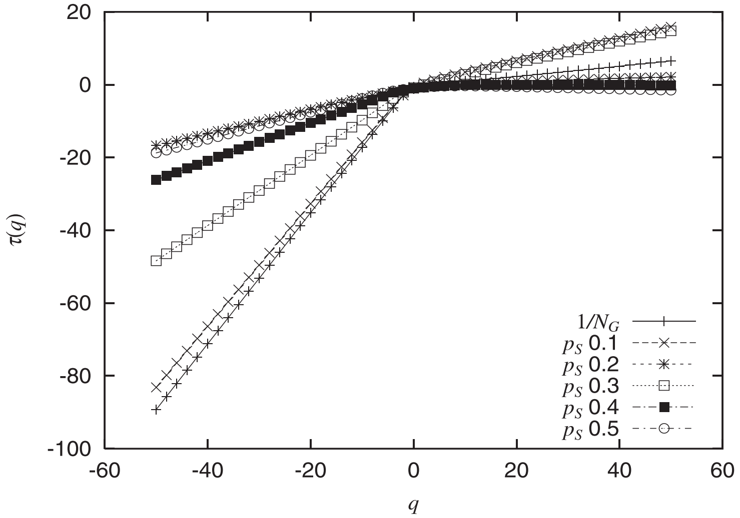

Figure 5: example of $\tau(q)$ for probabilistic decision

5 Feature extraction of multi-fractal time series

5.1 Feature extraction using prediction

Contrast to the WTMM, there exists no way to define the multi-fractality in the time domain while the origin of multi-fractality is still unknown. It is seen from conventional works related to the multi-fractality of the time series, there exit no definition of the multi-fractality in the time domain itself. As an alternative, we propose here a method to extract features from the multi-fractal time series based on the approximation of mono-fractal time series derived from corresponding original one[16][17][37][38][39]. As is seen in conventional works, slight changes of parameters in generating fractal time series show fluctuations and sometime jumps in the time series[7][35]. Therefore, we relay on the assumption that we can approximately estimate mono-fractal counterpart from the original multi-fractal time series. Then, the difference between multi-fractal and corresponding mono-fractal time series result in feature extraction [28]-[30].

At first, we show the prediction (smoothing) method of mono-fractal time series by specifying scaling of time axis.

Model Fitting

Assume a general linear input-output system

$$y(t) = \int_0^{t_0} h(t, t-\tau)x(\tau)d\tau, \quad t > t_0. \tag{30}$$

whereby, we treat the case of system identification of the time series having the same input and output expressed as $y(t) = x(t)$.

Then, we specify the model by introducing the impulse response function $h(t, \tau)$ of linear time-variant system represented by using a set of scaling functions $\phi(t)$ as follows.

$$h(t, \tau) = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} h_{ij}\phi_{Ni}(t)\phi_{Nj}(\tau), \tag{31}$$

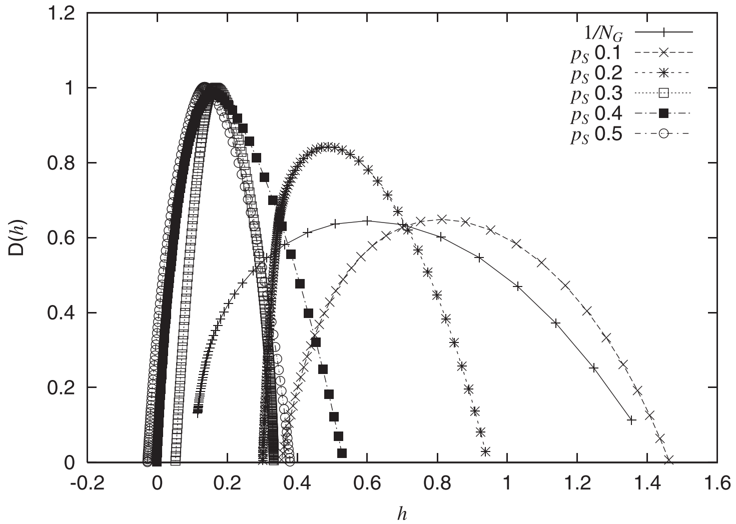

Figure 6: An example of $D(h)$ for probabilistic decision

where

$$\phi_{ij}(t) = \phi(2^i t - j). \tag{32}$$

$$\phi(t) = \begin{cases} 1, & 0 < t \le 1, \\ 0, & \text{otherwise.} \end{cases} \tag{33}$$

Since the coefficients h_{ij} of the impulse response function are unknown, then they are determined so that the mean square error $E_t = (y(t) - x(t))^2$ between observed value $x(t)$ and fitted value $y(t)$ obtained by using equation (30) is minimized. Since the derivatives of the evaluating function with respect to h_{ij} are easily estimated by using the property of scaling function, the minimization task using steepest descent algorithm is not hard.

Prediction

Let $T_s(T_e)$ be defined as the beginning (end) of the time period where the time series is observed ($T_s < t < T_e$). For simplicity we set $T_s = 0, T_1 = T_e$. Then, the time series $y(t)$ is predicted in the period $0 < t < T_2, T_2 > T_1$ by using the time scale expansion. Letting

$$b = a^D, \quad a = T_2/T_1, \quad T_2 > T_1. \tag{34}$$

where, the constant D is the mono-fractal dimension of the time series $x(t)$. The fractal dimension D is limited in a range so that the time series bear the fractal geometry.

$$1 < D < 2, \quad H = 2 - D. \tag{35}$$

where H is the Hurst parameter.

Then, by using the property of fractality embedded in $x(t)$, the following equation is approximately held in $0 < t < T_2$.

$$y(t) = b^{-1} \int_0^{bt_0} h\left(\frac{t}{b}, \frac{t-\tau}{b}\right) x(\tau) d\tau. \tag{36}$$

In this formulation, if we expand the time axis by a times, then we observe b pieces of segments (fractal geometry) in $0 \leq t \leq T_2$.

In other words, the same relation of linear prediction is held in the period $0 < t < T_2$, if we use $h(t,\tau)$ estimated in $0 < t < T_e$ which is b times expanded in the time scale.

The geometry included in equation (36) is explained by using the self-similarity in the Koch curve. In the Koch curve the curve spread in two-dimensional direction, we find b pieces of basic pattern if we expand the abscissa by a times . Since we have no enough space to explain the prediction method, please see[16][17][37]-[39].

It is approved to take the range of prediction $T_1 < t < T_2$ comparable to the observation period $0 < t < T_1$.

5.2 Estimating mono-fractal counterpart

Salient property of mono-fractal time series exists in small number of parameters for reproductions. Even though no universal framework for characterizing the spectrum of mono-fractal time series, we find that measured spectra are roughly of the following form over several frequency decades.

$$S(\omega) = \sigma^2 \omega^{-\gamma}, \quad \gamma = 5 - 2D, \tag{37}$$

where D and σ^2 are the mono-fractal dimension and variance of the time series, respectively. We only need these two parameters for reproduction and prediction of time series as is described in pervious section.

It is demonstrated that orthonormal Wavelet basis expansions in terms of collections of uncorrelated wavelet coefficients lead us to much simpler formula for fractality measure [16][17][37]-[40]. Assuming that the time series is represented by using the orthogonal Wavelet transform.

$$x(t) = \sum_n \sum_m x_n^m \psi_n^m(t). \tag{38}$$

$$x_n^m = \int_{-\infty}^{\infty} x(t)\psi_n^m(t)dt, \tag{39}$$

where $\psi_n^m(t)$ is defined by using the scale and shift transform of the basic function $\psi(t)$. The numbers n, m mean the indices of the scale and shift transform (dilation and transform), respectively. Then, we obtain following relation for coefficient x_n^m derived from the characteristics of the Wavelet function.

$$\mathrm{var}(x_n^m) = \sigma^2 2^{-\gamma m}. \tag{40}$$

By taking the logarithm of equation (40), we have a linear regression of $\log[\mathrm{var}(x_n^m)]$ with respect to the index m. Then, theoretically we have

$$\log[\mathrm{var}(x_n^m)] = 2\sigma - \gamma m \log 2. \tag{41}$$

If we adopt a linear regression to the logarithm of variance $\log[\mathrm{var}(x_n^m)]$ calculated from the time series, then we can estimate mono-fractal dimension D and variance σ^2.

The salient property, however, is not direct applicable to the multi-fractal time series where we need various parameters for reproducing time series. Fortunately, in almost all cases in our simulation studies, the spectrum of multi-fractal time series is seem to be resemble to that of mono-fractal time series. The curve $\log[\mathrm{var}(x_n^m)]$ along m obtained from multi-fractal time series still follows smooth decay from $m = 0$ to the maximum index of m.

Then, it would appear reasonable to neglect the deviation of $\log[\mathrm{var}(x_n^m)]$ from a regression line so that we can estimate approximated parameters (D and σ^2) of corresponding mono-fractal.

For simplicity, we depict a straight regression line on the data $\log[\mathrm{var}(x_n^m)]$ obtained from the original time series to get parameters of mono-fractal counterparts[27]-[30].

5.3 Simulation results

In this section, we present simulation results for feature extraction based on the validity and functionality of prediction (smoothing) of mono-fractal counterparts corresponding the original multi-fractal time series. Since our main interest exists in finding the features of multi-fractal time series in the time domain, then we skip the process to compare and validate the results of feature extraction using obtainable results by other conventional methods.

Figure 7, 8 and 9 show examples of feature extraction for artificial stock price of Case 1, Case 5 and Case 6, respectively by depicting original time series, prediction and mean square error of detected features. These figures consist of two parts. In the upper part, the original time series is depicted by the solid line, and the prediction is depicted by the dashed line. In the lower part of figures, the mean square error corresponding to detected features is depicted by the solid line. In these figures, the left vertical axis denotes the scale for the original and predicted stock prices, and the scale for the square error of detected features.

There are a variety of characteristics of artificially generated stock prices in each simulation scenarios considered in studies, then extracted features may be affected by the time series themselves. A kind of categorization of time series may lead us to a comprehensive investigation. As approved in the recognition of real world stock prices, it is helpful to categorize generated stock prices by distinguishing the ranges of prices, fluctuations (whether various changes of prices are observed), and trends (whether monotonic increase of decrease of prices like salient regression lines are observed). Then, artificially generated stock prices are categorized into following three groups.

Group V: Various changes of patterns

In Case 1 and Case 2, we see relatively large range of prices, and also various changes of patterns. Then, these prices are seemed to be resemble to real stock prices. Especially, in Case 1 there are many fluctuation and changes in prices.

Group M: Monotonic trend and small changes

In Case 3,4 and 5, we find dominant increasing or decreasing trends on the all region of time axis. Then, even though there are various and small-sized changes in a short period of time, these fluctuation are not regarded to be dominating whole of stock prices.

Group S: Small range and small-sized variations

In Case 6, there are many changes (fluctuations) of prices but they are small-sized, and prices move upward and then downward in a limited range. Moreover, we find no dominant trends.

Even though there exist still others which cannot be addressed so directly, but characteristics above are representative and dominant in these groups. Then, we summarize the result of feature description in the time domain along the categorization of these groups. In Group V, there are many feature points where a kind of small spikes are observed. At the same time, we see several time points where relatively large spikes are found, but the total number of feature points are not large. By examining the original time series, it is found that these feature correspond to rapid changes of prices in a short time period.

In Group M, there are many feature points with small spikes having similar amplitude, and these time points are relatively dense. At the same time, we observe relatively small number of large spikes. Originally, the time series of Group M have dominant and monotonic trends in prices, and then it is imagined that the time series is hard to be approximated by a mono-fractal time series. As a result, the approximation error reveals as a large spike. These phenomena correspond to large jumps of real stock prices under stagnated trades. These fluctuations and

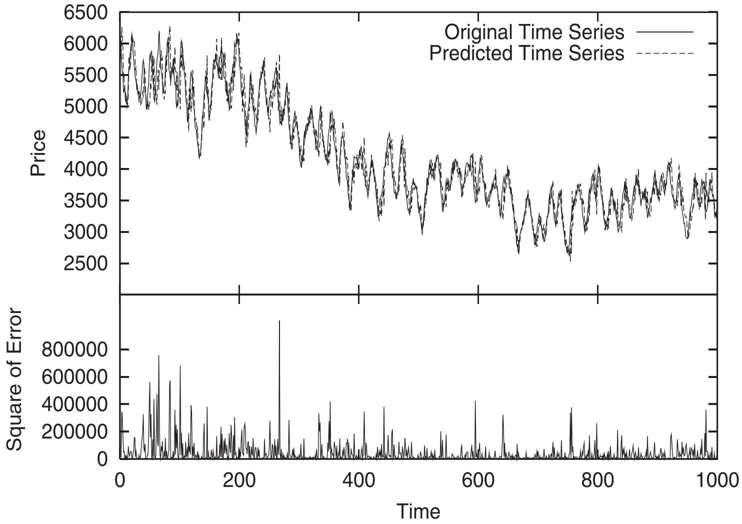

Figure 7: An example of stock price of Case 1: Group V (original, prediction and detected)

jumps are also characterized in conventional works as typical shapes of stock prices exhibiting multi-fractality.

In Group S, there are few time points of features with large square of errors, and small number of features with large amplitude are observed. The fact reflects that the approximated mono-fractal time series is very resemble to the original multi-fractal time series. It is also found in the analysis of $\tau(q)$ and $D(h)$ for Case 6 where the time series is considered to bear multi-fractality, however, they deviate from strict multi-fracatility.

5.4 Estimation of agents' behavior using Discriminant Analysis

We can see various characteristics of multi-fractality for each cases of composition of agents in artificial stock market through $\tau(q), D(h)$ and other features obtained from prediction for mono-fractal time series which is the counterpart of multi-fractal time series. Then, it is possible to estimate composition of dominant agents based on the analysis of features obtained from results such as $\tau(q), D(h)$. As is shown in Fig.1, our final purpose of the method is the estimation of composition of agents who are seemed to be major members of stock market and to contribute to generate stock prices. If we can formalize the estimation of the composition of dominant agents (traders) in the market by the multi-fractality analysis of the stock price, then it helps us to improve studies in real market for further investments.

In the following, we use conventional Multi-variate Discriminant Analysis (MDA) by employing statistical package to identify classes. The inputs (discriminant variables) for the MDA are composed of following observations. So as to clarify the definition of these variables, the outlines of definitions are depicted in Figure 10.

(1) value of $h = h_0$ where $D(h)$ reaches its maximum (p).

(2) spread of $D(h)$ by defining d_1, d_2 for which $D(h_0 - d_1) = D(h_0 + d_2) = 0.5$

(3) statistics (histogram having 4 same ranges) of spikes observed in extracted features (prediction error) obtained by fitting mono-fractal counter parts to the original multi-fractal time series.

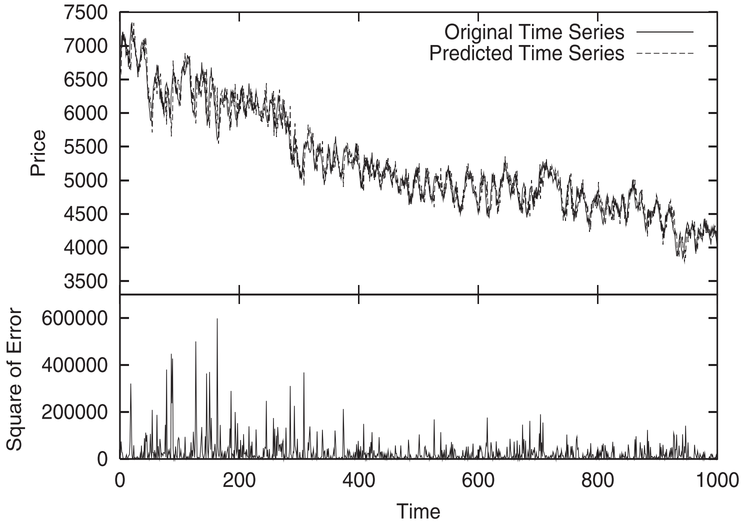

Figure 8: An example of stock price of Case 5: Group M (original, prediction and detected)

Table 5: Result of classification by MDA for Case 1 through 6.

	Case1	Case2	Case3	Case4	Case5	Case6
Case 1	1.0	0	0	0	0	0
Case 2	0	0.90	0	0	0.05	0.05
Case 3	0.1	0	0.9	0	0	0
Case 4	0.1	0	0	0.9	0	0
Case 5	0.2	0	0	0	0.80	0
Case 5	0	0	0.05	0	0	0.95

We simply prepare 20 samples of generated time series for each group (cases) under the same conditions of simulation studies, and we obtain 120 samples as a whole. And we use leave-one-out cross-validation ,that is, a discrimination function is created by 119 samples in 120 samples by considering the variable described in the above as an input, and it applies using the one remaining samples as test data. This is applied to all 120 samples.

Table 5 shows the result of the classification (recognition) by the MDA for Case 1 through 6 by showing the rate of recognition. Vertical and the horizontal column correspond to the original cases (classes) and estimated cases, and the number in each lattice means the rate of recognition. As is seen from the result, we can identify the cases of composition of agents at the rate about 90.8%. The fact means that the estimation of more accurate composition of agents dominating the market may be possible by using the method proposed by the paper.

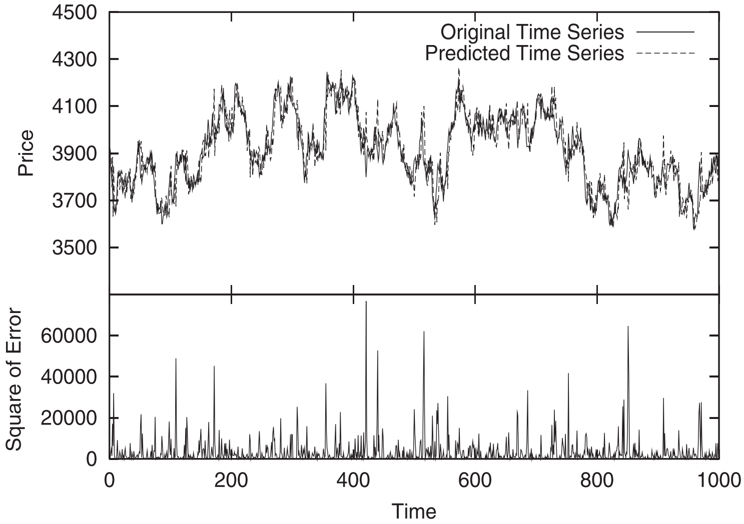

Figure 9: An example of stock price of Case 6: Group S (original, prediction and detected)

6 Conclusion

This paper treated the multi-fractality analysis of time series in artificial stock market generated by multi-agent systems based on the GP and its applications. We assumed five types of agents partly preferring forecast equations or forecast rules to support their decision making, and another type agents with random behavior. We defined six cases by changing the composition of agents. Then, the time series of artificial stock price revealed as a multi-fractal signal for these cases. By applying prediction method for mono-fractal time series, the features of the multi-fractal time series were extracted, and the origin of multi-fractal processes in artificial was discussed. Then, a backward method was applied to estimate the composition of agents who are seemed to be major members of stock market by examining the features of multi-fractality for stock price time series. As a result, the effect agents of each type on stock prices was comprehensively discussed using $\tau(q)$ and $D(h)$ which are common tools to investigate multi-fractality. Different from conventional works, our work will contribute to figure out the behavior of institutional investors whose activities are well reflected to stock prices through markets.

For future works, it is necessary to apply the method of the paper to real stock prices. Moreover, the cases where nonsingularity is observed in $D(h)$ and the stock price bearing monotonic characteristics also will be examined. Further works by the authors will be continued.

Acknowledgement

This research was partly supported by the Japan Society for the Promption of Science under the grant number (B)19310099, and authors would like to thank the organization.

References

[1] J.Alvarez-Ramirez,M.Cisneros,C.Ibarra-Valdez and A.Soriano: Multifractal Husrt analysis of crude oil prices. *Physica A*,312 (2002),651-670.

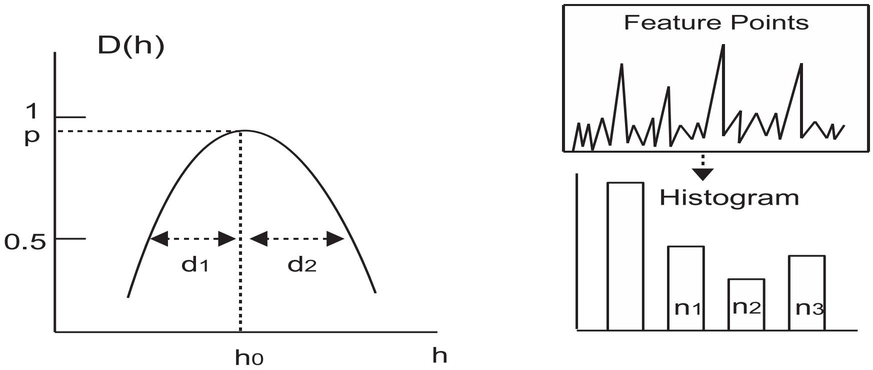

Figure 10: Outline of definition of discriminant variables

[2] Arneodo,A.,Bacry,E., and Muzy,J.F.:The thermodynamics of fractals revisited with wavelets,Physica A,213,(1995),232-275.

[3] W.B.Arthur,J.Holland.,B.LeBaron,R.Palmer and P.Taylor: Asset pricing under endogenous expectations in an artificial stock market, in Authur,W.B et.al (eds) *The Economy as a Evolving Complex System II*, Addison-Wesley, Reading MA,pp.15-44,1997.

[4] E.Barcy,J.Delour and J.F.Muzy: Modeling financial time series using multifractal random walks. *Physica A*,299 (2001),84-92.

[5] R. A. Brealey, *An introduction to risk and return from common stocks*, Addison-Wesley, 1986. 1989.5

[6] L. B. Brooker, D. E. Goldberg and J. H. Holland: Classifier system and genetic algorithm. *Artificial Intelligence* , 40 (1989), 235–282.

[7] L. Calvet and A. Fisher: Forecasting multifractal volatility. *Journal of Econometrics*, 115 (2001), 27–58.

[8] 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.

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

[10] X. Chen and S. Tokinaga: Synthesis of multi-agent systems based on the co-evolutionary Genetic Programming and its applications to the analysis of artificial market. *IEICE Trans.Fundamentals*, J86–A (2003), 1038–1048.

[11] X. Chen and S. Tokinaga: Multi-agent-based modeling of artificial stock markets by using the co-evolutionary GP approach. *Journal of Operations Research Society of Japan* , 47 (2004), 163–181.

[12] J. D. Farmer and S. Joshi: The price dynamics of common trading strategies. *Journal of Economic Behavior & Organization*, 49 (2002), 149–171.

[13] A. L. Goldberger, L. A. n. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Moody, C. K. Peng and H. E. Stanley: PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. Circulation 101(23): e215–e220 http://www.physionet.org/physiotools/multifractal/

[14] S. J. Grossman and J. Stiglitz: On the impossibility of informationally efficiency markets. *American Economic Review*, 70 (1980), 393–408.

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

[16] Y. Ikeda and S. Tokinaga: An estimation method for fractal surfaces by using scale expansion and its application(in Japanese). *IEICE Trans. Fundamentals*, J81–A (1998), 1127–1133.

[17] Y. Ikeda and S. Tokinaga: Evaluation of stock option prices by using the prediction of fractal time-series (in Japanese). *Journal of Operations Research Society of Japan*, 42 (1999), 18–31.

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

[19] Y. Ikeda and S. Tokinaga: Controlling the chaotic dynamics by using approximated system equations obtained by the genetic programming. *IEICE Trans.Fundamentals*, E84–A (2001), 2118–2127.

[20] Y. Ikeda and S. Tokinaga: Synthesis of multi-agent systems based on the co-evolutionary Genetic Programming by considering social learning and its applications to the analysis of artificial stock market. Journal of the Japan Society for Management Information, Vol.12, No.3, (2003), 17–35.

[21] Y. Ikeda and S. Tokinaga: Chaoticity and fractality analysis of an artificial stock market by the multi-agent systems based on the co-evolutionary Genetic Programming. *IEICE Trans.Fundamentals*, E87–A (2004), 2387–2394.

[22] Y. Ikeda, X. Chen and S. Tokinaga: Analysis of chaotic behavior of input pricing realized by the multi-agents systems based on the Co-evolutionary Genetic Programming and its applications (in Japanese). *IEICE Trans.Fundamentals*, J89–A (2006), 298–307.

[23] Y. Ikeda and S. Tokinaga: On the multifractal processes of stock prices in artificial markets consisting of multi-agents with GP-learning. *Technical Report of IEICE* , CAS2005–118 (2006), 7–12.

[24] Y. Ikeda and S. Tokinaga: Analysis of multi-fractal signals in artificial markets generated by multi-agents using the Genetic Programming for learning and its applications. *Proc. of NOLTA2006* (2006), 135–138.

[25] M. J. Keith and M. C. Martin:Genetic programming in C++: Implementation issues. in (ed) K. E. Kinnerar, Jr., *Advance in Genetic Programming* (MIT Press, 1994).

[26] J. R. Koza: *Genetic Programming*, MIT Press, 1992.

[27] J. R. Koza: *Genetic Programming II:Automatic Discovery of Reusable Programs*, MIT Press, 1994.

[28] B. LeBaron, W. B. Arthur and R. Palmer: Time series properties of an artificial stock market, Journal of Economic Dynamic and Control, Vol.23, pp.1487–1526, 1999.

[29] B. LeBaron: Agent-based computational finance:suggested readings and earlyresearch. *Journal of Economic Dynamic & Control*, 24 (2002), 679–702.

[30] J. Lu, S. Tokinaga and Y. Ikeda: Explanatory rule extraction based on the trained neural network and the genetic programming. Journal of Operations Research Society of Japan, 43 (2006), 66–82.

[31] J. Lu and S. Tokinaga: An Aggregated Approximation for modeling of time series based on the Genetic Programming and its application to Clustering (in Japanese). *IEICE Trans.Fundamentals*, J88–A (2005), 803–813.

[32] J. Lu and S. Tokinaga: Two-stage recognition method of time series based on symbolic representation of categories obtained by segment classification using the Genetic Programming and its applications to prediction(in Japanese). *IEICE Trans.Fundamentals*, J88–A (2005), 803–813.

[33] J. F. Muzy, E. Barcy and A. Arneodo: Multifractal formalisum for fractal signals:The structure function approach versus the wavelet-transform modulus-maxima method. *Physical Review E* (1993), 47 875-884.

[34] R. Pastor-Satorras: Multifractal properties of power-law time sequences:Application to rice piles. *Physical Review E*, 56 (1997), 5284–5294.

[35] R. H. Riedi, M. S. Crouse, V. J. Ribeiro and R. G. Baraniuk: A multifractal wavelet model with application to network traffic. *IEEE Trans. IT*, 45 (1999), 992–1018.

[36] 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 (201), 321–361.

[37] S. Tokinaga: Fractal theory and its applications to the prediction of economic time series. *Communications of the Operations Research Society of Japan*, 46 (2001), 640–645.

[38] S. Tokinaga, H. Moriyasu, A. Miyazaki and T. Shimazu: A forecasting method for time series bearing fractal geometry and its application(in Japanese). *IEICE Trans.Fundamentals*, J79–A (1996), 1793–1800.

[39] S. Tokinaga, H. Moriyasu, A. Miyazaki and T. Shimazu: A forecasting method for time series bearing fractal geometry and its application(in Japanese). *IEICE Trans.Fundamentals*, J79–A (1996), 1793–1800.

[40] G. W. Wornell and A. V. Oppenheim: Estimation of fractal signals from noisy measurement using Wavelets. *IEEE Trans. Signal Processing*, 40 (1992), 611–623.

[41] M. Yababe and S. Tokinaga: Applying the genetic Programming to modeling of diffusion processes by using the CNN and its applications to the synchronization (in Japanese). *IEICE Trans.Fundamentals* , J85–A (2002), 548–559.

Shozo Tokinaga〔Professor, Graduate School of Economics, Kyushu University〕
Seigo Matsuno〔Associate Professor, Department of Business Administration, Ube National College Technology〕