Buying and Selling Stocks of Multi Brands Using Genetic Network Programming with Control Nodes

Etsushi Ohkawa  Non-member  (Waseda University, etsushi@fuji.waseda.jp)
Yan Chen  Non-member  (Waseda University, chenyan1513@akane.waseda.jp)
Zhiguo Bao  Non-member  (Waseda University, baozhiguo@moegi.waseda.jp)
Shingo Mabu  Member  (Waseda University, mabu@aoni.waseda.jp)
Kaoru Shimada  Member  (Waseda University, k.shimada@aoni.waseda.jp)
Kotaro Hirasawa  Member  (Waseda University, hirasawa@waseda.jp)

Keywords: evolutionary computation, Genetic Network Programming, Candlestick Chart, buying and selling stocks

Recently, Genetic Network Programming (GNP) has been proposed, which extends the functions of Genetic Algorithm (GA) and Genetic Programming (GP). GNP with control nodes (GNPcn) was also proposed recently, which extends the functions of GNP.

GNP has directed graph structures in which a number of nodes executing functions of judgment or processing are connected by directed links to each other. Its features are the repeated usage of nodes, implicit memory of the past actions and effective search of solutions.

In this paper, a new approach is proposed and verified for developing stock trading strategies based on the combinations of Candlestick Charts by GNPcn. Combined Candlestick Charts are usually used for the judgment of buying and selling of stocks. This paper determines the optimal combinations of candlestick using GNPcn, for determining the timing of buying / selling stocks of multi issues.

GNPcn has some control nodes, while a conventional GNP has just one start node and no control nodes. Therefore, GNPcn can search the solution space widely by increasing the number of control nodes and as a result can find many distinguished buying and selling rules. Each group of control nodes is assigned to each stock, so that GNPcn can find a strategy of buying and selling stocks of multi issues (shown in Fig. 1). The purchase capital can be effectively distributed to multi issues, where more capital is invested to the stock with larger profits. As a result, more profit is promised by the trading.

GNPcn distributes the purchase capital to each stock based on the distribution ratio. Distribution ratio $P(i, d)$ of stock $i \in S$ on day $d$ is calculated by the following equation:

$$ P(i, d) = \frac{\exp \left( \frac{R(i, d)}{T} \right)}{\sum_{i \in S} \exp \left( \frac{R(i, d)}{T} \right)} \quad \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cd -13-
Buying and Selling Stocks of Multi Brands Using Genetic Network Programming with Control Nodes

Etsushi Ohkawa*  Non-member
Yan Chen†  Non-member
Zhiguo Bao*  Non-member
Shingo Mabu*  Member
Kaoru Shimada**  Member
Kotaro Hirasawa*  Member

A new evolutionary method named “Genetic Network Programming with control nodes, GNPcn” has been applied to determine the timing of buying or selling stocks. GNPcn represents its solutions as directed graph structures which has some useful features inherently. For example, GNPcn has an implicit memory function which memorizes the past action sequences of agents and GNPcn can re-use nodes repeatedly in the network flow, so very compact graph structures can be made. GNPcn can determine the strategy of buying and selling stocks of multi issues. The effectiveness of the proposed method is confirmed by simulations.

Keywords: evolutionary computation, Genetic Network Programming, Candlestick Chart, buying and selling stocks

1. Introduction

Evolutionary Computation aims at producing the solutions for optimization problems or useful structures of data by change, composition and selection imitating the mechanism of life. Genetic Algorithm (GA) (1) and Genetic Programming (GP) (2) are typical evolutionary algorithms. We have proposed Genetic Network Programming (GNP) (3)–(7) recently, which extends the functions of GA and GP. Genetic Network Programming with control nodes (GNPcn) (8) was also proposed recently, which extends the functions of GNP.

GNP has directed graph structures in which a number of nodes executing a function of judgment or processing are connected by directed links to each other. Its features are the repeated usage of nodes, implicit memory of the past actions and effective search of solutions (3).

On the other hand, many approaches to the stock price prediction and buying / selling strategy using soft computing such as Genetic Algorithm, Genetic Programming, Neural Networks (NNs) (9) etc. have been proposed (10)–(14) and many of them support efficient market hypothesis (15): E. F. Fama presented the efficient market theory in terms of a fair game model, citing that investors could be confident that a current market price fully reflects all available information and the expected return based upon this price is consistent with its risk.

Supposing that this theory is the case, it means that the specific investors cannot gain the profit over the average in the market.

While, there exists several research on stock markets using behavioral finance, GA, GP, NNs, data mining and so on, some of which are negative to efficient market hypothesis.

In this paper, a new approach is proposed and verified for developing stock trading strategies based on the combinations of Candlestick Charts by GNPcn. Combined Candlestick Charts are usually used for the judgment of buying and selling of stocks. Stock market participants have proposed several effective combinations of Candlestick Charts empirically. But, because of the complicated fluctuations in stock prices, it was difficult to get the optimized combinations of Candlestick Charts. This paper describes the optimal combinations of Candlestick Charts using GNPcn, for determining the timing of buying / selling stocks of multi issues.

GNPcn has some control nodes, while a conventional GNP has just one start node and no control nodes. Therefore, GNPcn can search the solution space widely by increasing the number of control nodes and as a result can find many distinguished buying and selling rules. Each group of control nodes is assigned to each stock, so that GNPcn can find a strategy of buying and selling stocks of multi issues. The purchase capital can be effectively distributed to multi issues, where more capital is invested to the stock with larger profits. As a result, more profit is promised by the trading.

The outline of this paper is as follows: Section 2 briefly describes Genetic Network Programming with control nodes. In section 3, the proposed stock buying / selling
method of multi issues using GNPcn is described. The simulation results obtained with the model are reported in section 4. The paper concludes with a summary of the results in section 5.

2. Genetic Network Programming with Control Nodes

2.1 Structure of GNP

Fig. 1 shows the basic structure of GNP. Conventional GNP consists of the following several judgment nodes, processing nodes and one start node.

Start node: Start node indicates the start point of GNP transition.

Judgment node: Judgment node returns a judgment result on the information gained from the environments, and determines the next node to which the current activated node should move.

Processing node: Processing node works as a processing function, and the next node is uniquely determined by its single connection.

The current activated node is not compulsorily transferred to the start node in GNP and there is no terminal node in GNP. Therefore, once GNP is booted up, the successive node activation is carried out according to the network flow until the time limit.

In Fig. 1, the gene structure of node \( i \) is divided into two parts: one is the content of the node, and the other is the connection of the node. \( NT_i \) means the kind of the node: control node, judgment node, or processing node. \( ID_i \) means the function of node \( i \). \( C_{ik} \) means the node number to which the \( k^{th} \) branch of node \( i \) connects.

In GNP, it is possible to fix the number of nodes in advance, so it never cause bloat which can be seen in GP\(^{(2)}\). In addition, the past history of the node transition affects the current node because the node transition of GNP is executed according to its node connections. It means that the network structure of GNP has an implicit memory function of the past actions, and it realizes the compact structure of GNP without numerous nodes\(^{(3)}\).

2.2 Structure of GNPcn

In conventional GNP, the current activated node doesn’t need to return to the start node. This purpose is to repeatedly use most of the judgment nodes and processing nodes in GNP as much as possible. However, there is a possibility that some of the nodes are not used in the conventional GNP, therefore, GNP with Control Nodes (GNPcn) is proposed.

GNPcn starts from one of the “control nodes,” and the current activated node is transferred to one of the control nodes after executing a certain number of processing nodes. Consequently, the performance of GNPcn improves because the increase of the number of control nodes contributes to searching the solution space widely and finding many smart solutions.

Fig. 2 shows the basic structure of GNPcn. Several control nodes are added to the conventional GNP. GNPcn uses one of the groups of control nodes for one stock when dealing with multi issues in the stock market.

2.3 Node Transition of GNPcn

For example, when GNPcn deals the stock “i,” it starts its node transition from control node “\( C_1 \)," and the current activated node returns to one of the control nodes (\( C_{i1}, \ldots, C_{im} \)) successively after transiting “m” processing nodes from the last control node (this is shown in Fig. 3).

In “one day” period, the current activated node of each stock transits one node. This procedure is as fol-
Buying and Selling Stocks of Multi-brands using GNPcn

2.4 Evolution of GNPcn

Fig. 4 shows the whole flow chart of the evolution of GNPcn.

In the initialization of a GNPcn population, the connections of the nodes in each individual are randomly set. In the evaluation, the stock trading is carried out and the fitness of each individual is calculated. In our case, the fitness value is assumed to be the sum of the fitness of each stock. In each generation, the elite individuals are preserved and the rest of the individuals are replaced by the new ones generated by crossover and mutation.

When the current generation is not the last generation, GNPcn continues to evolve, else, it finds the best individual and terminates evolution.

2.4.1 Crossover

The crossover is operated between two parents, and two new individuals are generated. The procedure is as follows:

(1) Select two individuals as parents using tournament selection twice.
(2) Some nodes in the parents are selected as the crossover nodes with Crossover Probability \( P_c \).
(3) Two parents exchange the connection of the corresponding crossover nodes.
(4) Also the value “\( m \)” (the number of processing nodes for returning to the next control node) and the number of control nodes of each stock “\( n \)” are crossovered with \( P_m \).
(5) The two new individuals are added to the next generation.

2.4.2 Mutation

The mutation is executed in one parent and one new individual is generated.

(1) Select one individual as a parent using tournament selection.
(2) Some node connections are selected with Mutation Probability \( P_m \). The selected connections are changed randomly and the new individual is generated.
(3) Also the value “\( m \)” and “\( n \)” are changed randomly with \( P_m \). The range of change is ±2, in our case.
(4) The new individual is added to the next generation.

3. Strategy of Buying and Selling Stocks of Multi Issues Using GNPcn

3.1 Candlestick Chart

Candlestick Chart originated in a rice market in Edo period in Japan, and indicates the fluctuations in stocks in a day as a stick of the candle. Candlestick Chart has been winning international recognition for its good indication of stock prices, and it has been widely used as the means of indicating the fluctuations of the stocks \(^{(16)-(18)}\).

Candlestick Chart represents four stock prices: Opening price, High price, Low price and Closing price. The daily candle line represents the fluctuation of stock prices of a specific day, and the weekly and monthly ones represent that of a specific week and month, respectively.

In this paper, we deal with the daily Candlestick Chart, each of which represents the range between the high price and low price of the day’s trading. As shown in Fig. 5, the top of the candlestick represents the high price of the day, and the bottom of it represents the low price of the day. The box is called the body of a candlestick and the height of the body represents the range between the opening price and closing price of the day. When the body is black, it means that the closing price is lower than the opening price, and when white, it means the opposite: the closing price is higher than the opening price.

Therefore, we can know four prices from a Candlestick Chart. However, traders usually judge the chances of trading by combining Candlestick Charts because of the insufficient information in a single day’s chart.

In the Candlestick Chart patterns, the existence of the window indicates the additional chance of trading. As
shown in Fig. 6, the window is the range between the high price and the low price during the two days. The existence of a window and the kind of the body result in many kinds of Candlestick Chart patterns.

### 3.2 Strategy of Buying and Selling Stocks

In GNPcn, judgment nodes check Candlestick Chart patterns, and processing nodes work for buying or selling stocks. Therefore, a chain of judgment nodes and the processing nodes expresses the buying / selling strategy. The concrete explanation of trading rules on stock markets using Candlestick Chart with GNPcn is as follows.

Table 1 shows the node genes of control node, judgment node and processing node of GNPcn.

One of the groups of control nodes is assigned to each stock \( i \in S \) (\( S \): set of stocks). GNPcn starts at \( C_1 \) and the current activated node returns to one of the control nodes of each stock after transiting \( m \) processing nodes from the last control node. That is, the current activated node returns to \( C_{i+1} \) firstly, and next \( C_{i+2}, C_{i+4}, \ldots, C_m, C_1, \ldots \) consecutively. This is shown in Fig. 3.

Trading timing is determined by the evolution. The simulation result was evaluated by the following Fitness, which shows the sum of Fitness \((i)\) of stock \( i \).

\[
\text{Fitness} = \sum_{i \in S} \text{Fitness}(i) \quad \text{................. (1)}
\]

\[
\text{Fitness}(i) = \text{Sell}(i) + \text{Price}(i) \times \text{Unit}(i) - \text{Buy}(i) \quad \text{................. (2)}
\]

where,

- \( \text{Sell}(i) \): sum of the money earned by selling stock \( i \).
- \( \text{Price}(i) \): price of stock \( i \) at the end of the trading.
- \( \text{Unit}(i) \): amount of stocks of stock \( i \) in hand at the end of the trading.
- \( \text{Buy}(i) \): sum of the money for buying stock \( i \).
- \( S \): set of stocks.

In Eq. (1) and (2), Fitness means the total capital gains and Fitness \((i)\) means the capital gains of stock \( i \) obtained during the trading.

3.3 Judgment Node of GNPcn

The number of kinds of the judgment nodes is only one, and a judgment node has eight branches. One branch is selected depending on the color of the Candlestick body, whether there exists a window or not, and whether or not yesterday’s closing price is higher than the opening price of the day before yesterday. The eight patterns are shown in Fig. 7.

3.4 Processing Node of GNPcn

When GNPcn determines to buy stock \( i \) on day \( d \), the stock \( i \) is bought as much as possible using \( P(i,d) \), where \( P(i,d) \) is the ratio to be used for stock \( i \) out of the total fund in hand on day \( d \). \( P(i,d) \) is explained in the next subsection. When GNPcn determines to sell stock \( i \), all of the stocks of issue name \( i \) in hand are sold off.

3.5 Distribution Ratio to Each Stock

GNPcn distributes the purchase capital to each stock based on the distribution ratio. Distribution ratio \( P(i,d) \) of stock \( i \) on day \( d \) is calculated by the following equation:

\[
P(i,d) = \frac{\exp \left( \frac{R(i,d)}{T} \right)}{\sum_{i \in S} \exp \left( \frac{R(i,d)}{T} \right)} \quad \text{................. (3)}
\]

where,

\[
R(i,d) = \begin{cases} 
0 & \text{Buy}(i,d) = 0 \\
\frac{G(i,d)}{\text{Buy}(i,d)} & \text{otherwise}
\end{cases} \quad \text{............. (4)}
\]

\[
G(i,d) = \text{Sell}(i,d) + \text{Price}(i,d) \times \text{Unit}(i,d) - \text{Buy}(i,d) \quad \text{................. (5)}
\]
Buy(i, d): sum of the money for buying stock i until day d.
Price(i, d): price of stock i on day d.
Unit(i, d): amount of stocks of stock i in hand on day d.
Sell(i, d): sum of the money earned by selling stock i until day d.

The purchase capital is evenly distributed to each stock on an initial day, but the distribution ratio would be changed after the capital gains of each stock are obtained.

3.6 Procedure of Trading GNPcn individual starts its operation from one of the control nodes, the activated node is transferred to a processing node or a judgment node, the judgment or processing is executed in the current day, and the next node for the next day is determined. At the processing node, the trading is executed using the opening price of the day.

The concrete procedure of trading is as follows:
(1) Calculate the distribution ratio of each stock whenever the buying signal occurs in the processing nodes during the transition of each stock.
(2) Do the following for each stock until the end of the trading.
   • If the activated node is a judgment node, it determines the next node for the next day depending on the judgment result.
   • If the activated node is a processing node, firstly buys or sells stocks, then determines the next node for the next day.
   • If the processing node is executed m times from the last control node, then the next node is determined by the next control node.
(3) Calculate the fitness of each stock at the end of the trading.

4. Simulations

4.1 Simulation Conditions

4.1.1 Training and Testing Data The data used in the simulations are the stock data of 10 issues from A to J, all of which are listed in the First Section of the Tokyo Exchange market (see the appendix).

In the simulations, the following stock prices in 2001 through 2003 are used for training, and those in 2004 for testing.


4.1.2 Simulation Parameters Table 2 shows the parameters of the evolution of GNPcn. Four control nodes are assigned to each stock initially and we tested 10 stocks, i.e., the total number of control nodes is 40.

And we test the various value of T in Eq. (3), i.e., parameter of the distribution ratio \( P(i, d) \). \( T \) defines the strength of the distribution. If \( T \to \infty \), GNPcn distributes the fund evenly to each stock. And if \( T \to 0 \), the money is distributed more to the stock which can obtain a good mark. So \( T \) shows the distribution characteristic.

4.1.3 Study of Comparison In the simulations, GNPcn is compared with the conventional GNP, where, one GNP individual deals with only one stock, so we evolved 10 different GNP populations for 10 different stocks in our case. On the other hand, one GNPcn has 10 groups of control nodes, each of which deals with each stock, so one GNPcn can deal with 10 stocks as shown in Fig. 8. This is an advantage over the conventional GNP.

We also compared the proposed method with “Buy and Hold.” In GNPcn, capital gains obtained during the deal are also available for other stocks. But in conventional GNP, capital gains can be used for itself.

Other conditions for simulations are as follows:
• The program used in the testing is the best one at the last generation in the training.
• The initial fund for GNPcn is 5 million JPY. For “Buy and Hold” and GNP, the initial fund is also 5 million JPY for each stock trading.

4.2 Simulation Results

4.2.1 Parameter \( T \) in Distribution Ratio Table 3 shows the average fitness values over 30 independent trials in the training when the parameter \( T \) is set to different values. It is found from the results that
Fitness is fairly good in the training period when \( T \) is set to 0.005.

Therefore, we used the value of 0.005 for the parameter \( T \) in our simulations.

4.2.2 Comparison with Other Methods

Table 4 shows the profit ratios of GNP and GNPcn averaged over 30 independent trials.

It is clarified from the simulation results that the proposed method can obtain larger profit than GNP and “Buy and Hold.”

Fig. 9 shows an example of the changes of the distribution ratio \( P(i, d) \) of 10 stocks when GNPcn is evaluated using the testing data in 2004. We can see from Fig. 9 that stock \( H \) gained a high distribution ratio at early days of the testing, then GNPcn distributes the capital to stock \( A \) mainly after the middle days of the testing.

Fig. 10 shows an example of the changes of the capital gains \( G(i, d) \) of each stock on the \( d^{th} \) day when GNPcn is evaluated using the testing data in 2004. From the simulation results, we can see that the capital gains are increasing during the testing period.

It is found that the profit obtained by GNPcn is mainly from the stock \( H \) and \( A \), while other stocks do not gain profit so much, actually, some stocks lose money. But it is not fatal, because the losses are kept to a minimum.

The advantage of the proposed method is to determine the distribution ratio of the purchase capital to each stock automatically. As a result, it is clarified that GNPcn can distribute the purchase capital to the stocks with larger profits.

4.3 Parameters \( m \) and \( n \)

The number of the transitions of the processing nodes \( m \) and the number of control nodes \( n \) are changed by the evolution. In our simulations, \( m \) is changed between the range from 6 to 20 and \( n \) is changed from 2 to 7.

It is found that as a general trend, \( m \) and \( n \) do not take large values. This means that if \( m \) or \( n \) is very large, GNPcn can only use the local part of its structure and some nodes may be unused causing degraded performances of GNPcn.

4.4 Combinations of Candlestick Patterns

Fig. 11 shows several example of the effective candle-
Buying and Selling Stocks of Multi-brands using GNPcn

Fig. 10. Capital gain $G(i, d)$ of each stock in the testing period (2004).

Fig. 11. Examples of effective combinations of candlestick for buying stocks.

Fig. 12. Frequency of the usage of the nodes.

Table 5. Fitness and Fitness$(i)$ of 10 stocks (training in 2001 through 2003).

<table>
<thead>
<tr>
<th>Stock</th>
<th>Fitness$(i)$ [million JPY]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>$-3.27$</td>
</tr>
<tr>
<td>B</td>
<td>$3.12$</td>
</tr>
<tr>
<td>C</td>
<td>$-0.04$</td>
</tr>
<tr>
<td>D</td>
<td>$-0.04$</td>
</tr>
<tr>
<td>E</td>
<td>$-0.06$</td>
</tr>
<tr>
<td>F</td>
<td>$11.26$</td>
</tr>
<tr>
<td>G</td>
<td>$2.24$</td>
</tr>
<tr>
<td>H</td>
<td>$-1.40$</td>
</tr>
<tr>
<td>I</td>
<td>$-0.46$</td>
</tr>
<tr>
<td>J</td>
<td>$5.74$</td>
</tr>
<tr>
<td></td>
<td>Fitness $17.05$</td>
</tr>
</tbody>
</table>

of nodes and judgments. We can see that GNPcn uses selling processing node the most. However, when there exists no stocks of stock $i$ in hand but GNPcn decides selling stock $i$, the actual trading isn’t carried out. As shown in “Usage of judgment nodes,” GNPcn mainly uses the successive fall and rise.

4.5 Generalization of GNP Individuals

Table 5 shows the total capital gains Fitness and capital gains Fitness$(i)$ of stock $i$ in the training. As shown in Table 5, the trained GNPcn seems to be specialized in stock $F$ and $J$, but Fig. 10 shows GNPcn can handle the other stocks $A$ and $H$ in the testing period. So, the proposed method has better generalization ability than GNP because GNPcn can be trained by using various price patterns of all stocks, while the conventional GNP can only deal with one stock.
But, when the testing data and training data differ completely, GNPcn can not handle the stock trading well. We need to pay attention that the training data contain a modest amount of length and various stock price trends.

5. Conclusions

This paper proposed a stock trading model of multi issues to support traders by identifying Candlestick Chart patterns using GNPcn. GNPcn has some control nodes, and each group of control nodes is assigned to each stock, so GNPcn can create trading strategies of multi issues. The advantage of the proposed method is that it can calculate the distribution ratio of the purchase capital to each stock, then the purchase capital is distributed mainly to the stocks with larger profits.

In the simulations, GNPcn is compared with GNP using the stock data of 10 stocks. It is clarified from the simulation results that GNPcn performs better in terms of the profits than GNP. It means that the proposed method can be effective in dealing with real complicated stock markets.

In the future, GNPcn will be combined with on-line learning for creating more effective strategies of buying and selling stocks in the markets. In this paper, the Candlestick Chart body was classified into two categories according to the color of the body: black or white. For more accurate decision making in the trading, it is necessary to consider other ways of classifying the Candlestick Chart body type and determine the number of branches of the judgment nodes because there are many kinds of Candlestick Chart patterns.

(Manuscript received Jan. 16, 2008, revised May 9, 2008)

References


(9) C. M. Bishop: “Neural Networks for Pattern Recognition”, Oxford University Press (1994)


Appendix

1. 10 stocks

The list of 10 companies is as follows.

(A) Obayashi Corporation

(B) Meiji Seika, Ltd.

(C) Toray Industries Inc

(D) Asahi Kasei Corporation

(E) Takeda Pharmaceutical Company Limited

(F) Showa Shell Sekiyu KK

(G) Nippon Steel Corporation

(H) Hitachi, Ltd

(I) NEC Corporation

(J) Nissan Motor Co., Ltd.

Etsushi Ohkawa (Non-member) He graduated from Ibaraki National College of Technology, Japan and received the B.E. degree from National Institution for Academic Degrees and University Evaluation, Japan in 2007. He has been a master student in Graduate School of Information, Production and Systems of Waseda University, since April 2007.
Yan Chen (Non-member) She received the B.E. degree from Sichuan Normal University, China in 2005. From 2005 to 2006, she studied in the Business School of Sichuan University as a Master student. Since September 2006, she was a Master student in Graduate School of Information, Production and Systems of Waseda University. She received the M.E. degree from Waseda University, Japan in 2007. Since September 2007, she has been a doctor candidate at Waseda University.

Zhiguo Bao (Non-member) He received the B.E. degree in Department of Computer Science and Technology from Beijing Information Science Technology University, China in 1999. From 1999, he was with Beijing Hitachi Huasun Information Systems Co., Ltd. He received the M.E. degree from Graduate School of Information, Production and Systems, Waseda University, Japan in 2008. Since April 2008, he has been a doctor candidate at Waseda University.

Shingo Mabu (Member) He received the B.E. and M.E. degree in Electrical Engineering from Kyushu University, Japan in 2001 and 2003, respectively, and Ph.D. degree from Waseda University, Japan in 2006. Since April 2007, he has been an assistant professor in Graduate School of Information, Production and Systems of Waseda University. Dr. Mabu is a member of the Society of Instrument and Control Engineers and IEEE.

Kaoru Shimada (Member) He received the M.Med.Sci. degree from University of Tsukuba in 1999, and Ph.D. degree from Waseda University, Japan in 2007. Since 2007, he has been a visiting lecturer of Information, Production and Systems Research Center of Waseda University. Dr. Shimada is a member of the Society of Instrument and Control Engineers and IEEE.

Kotaro Hirasawa (Member) He received the B.E. and M.E. degree from Kyushu University, Japan in 1964 and 1966, respectively. From 1966 to 1992, he was with Hitachi Ltd., where he served as a vice president of Hitachi Research Laboratory. From December 1992 to August 2002, he was a Professor in Graduate School of Information Science and Electrical Engineering of Kyushu University. Since September 2002, he has been a Professor in Graduate School of Information, Production and Systems, Waseda University. Dr. Hirasawa is a member of the Society of Instrument and Control Engineers and IEEE.