A Hybrid Expert System for Generating Stock Trading Signals

Hosein Hamisheh Bahar, Mohammad Hossein Fazel Zarandi, Akbar Esfahanipour

***Abstract*—**In this paper, a hybrid expert system is developed by using fuzzy genetic network programming with reinforcement learning (GNP-RL). In this system, the frame-based structure of the system uses the trading rules extracted by GNP. These rules are extracted by using technical indices of the stock prices in the training time period. For developing this system, we applied fuzzy node transition and decision making in both processing and judgment nodes of GNP-RL. Consequently, using these method not only did increase the accuracy of node transition and decision making in GNP's nodes, but also extended the GNP's binary signals to ternary trading signals. In the other words, in our proposed Fuzzy GNP-RL model, a No Trade signal is added to conventional Buy or Sell signals. Finally, the obtained rules are used in a frame-based system implemented in Kappa-PC software. This developed trading system has been used to generate trading signals for ten companies listed in Tehran Stock Exchange (TSE). The simulation results in the testing time period shows that the developed system has more favorable performance in comparison with the Buy and Hold strategy.

***Keywords*—**Fuzzy genetic network programming, hybrid expert system, technical trading signal, Tehran stock exchange.

I. INTRODUCTION

EXPERT systems as modern technological tools having the ability to perform some specific assigned tasks, has gained more and more popularity through the time. Because of the attractiveness of stock prices, they have been under consideration in recent years, and numerous researches have been done to forecast them or to realize their behavior in markets. In this regard, technical indices and indicators have been used to generate stock trading rules based on their prices and exchange volume. On the ground that choosing an appropriate technical index or indices at the right time seemed complicated, researchers tried to apply evolutionary algorithms such as Artificial Neural Networks (ANN) [1], Genetic Algorithm (GA) [2], Genetic Programming (GP), Genetic Network Programming (GNP) etc. to overcome this challenge. GP [3] as an extension of GA [4] having a tree structure, has been widely used for generating technical trading rules [5], [6]. Furthermore, GNP, as an extension of GP [7], represents each individual as a network having three kinds of nodes named start node, processing nodes and judgment nodes which are connected to each other appropriately.

H. Hamisheh Bahar is with the Department of Industrial Engineering and Management Systems, Amirkabir University of Technology, Tehran, Iran (corresponding author; e-mail: hamish@aut.ac.ir).

M. H. F. Zarandi and A. Esfahanipour are with the Department of Industrial Engineering and Management Systems, Amirkabir University of Technology, Tehran, Iran (e-mail: zarandi@aut.ac.ir. esfahaa@aut.ac.ir).

GNP was used for extracting trading rules base on candlestick charts in [8] for the first time. In [9], [10], Reinforcement Learning (RL) was added to GNP by considering two sub nodes for each node. By adding RL to GNP, the more desirable the sub node performs in decision making, the more the probability of visiting it in the following becomes. After that in [11] control nodes were added to GNP, and made it possible to be used for portfolio optimization. In [12], [13] Chen et al. used Time Adapting GNP for the portfolio optimization. Because the trend of prices is changing, using a price window which shifts through the time during the evolution phase makes the individuals adapt to changing trends.

Fuzzy GNP for the first time was proposed by Sendari et al. [14]. In their paper, they used the fuzzy judgment nodes for deciding the next node considering that fuzzy judgment nodes which determine the next node probabilistically instead of some judgment nodes with a threshold, made the model more realistic.

In [15], Mabu et al. applied GNP for generating stock trading rules. In their research, during the evolution phase different rules were generated out of the elite individual and were accumulated in the appropriate rule pool. Different rule pools were created based on the price's trend and the generated trading signal. At last, the decision making was based on accumulated rules in the rule pools. In following years, Chen and Wang proposed a Risk-Adjusted GNP for portfolio optimization [16]. Consequently, they used conditional Sharp Ratio as the fitness function of their Risk-Adjusted model. In conditional Sharp Ratio, the deviation of return from its average value will be divided by Conditional Value-at-Risk.

In brief, in this paper using fuzzy processing nodes and applying trading rules in a frame-based structure contributed performance enhancement to the conventional GNP-RL. In the following, the implementation of these contributions and their effectiveness will come under deliberation.

The rest of the paper is organized as follows. In Section II, the proposed Fuzzy GNP-RL is described. In Section III, implementation of the proposed frame-based expert system in Kappa-PC is explained, and in Section IV, results are reported. Finally, the paper closes with our conclusion.

II. FUZZY GNP

In GNP, each individual is a network having three kinds of nodes. As shown in Fig. 1, it consists of a start node and several judgment and processing nodes. The start node just specifies the first node, and does not have any other role in performance of the graph structure. In judgment nodes, a technical indicator will be used for judgments, and in processing nodes, the decision to Buy, Sell or No Trade will be made. In virtue of GNP's graph structure, it is possible to use nodes repeatedly. Consequently, this possibility culminates in compact structure of GNP, and prevents bloating phenomenon which is more prevalent in GP [9].

A. Genotype of Fuzzy GNP-RL

In Fig. 2, the structure of judgment and processing nodes and the genes corresponding to them are presented.

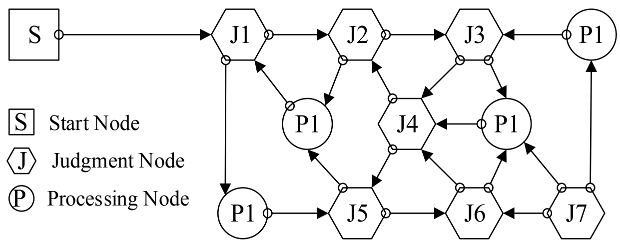

Fig. 1 Basic Structure of GNP

In gene representation, NT_i represents the node type. In fact, $NT_i = 0$ means the node is start node, and $NT_i = 1$ shows processing nodes, and $NT_i = 2$ represents judgment nodes. The d_i represents the time delay corresponding to each kind of nodes. The time delay is the time that it takes to transit from node i to node j. By setting time delay, the number of visited nodes during each trading day can be managed. The time delay for the start node is zero, and for the processing nodes is equal to 5, and for judgment nodes is considered 1 time unit. In addition, each day consists of five time units. Therefore, more than one trade cannot be executed in each trading day. For each sub node, Q represents the Q value needed for RL. The ID_i shows the identification number of each sub node. For judgment nodes, it specifies the technical index which should be used, and in processing nodes it indicates the function of the sub node which can be Buy or Sell. The α_i and β_i are parameters which are used in membership function of each sub node. In judgment of sub nodes i, C_i^A and C_i^B show the number of the next node. Based on the technical index, one of them will be chosen as the next node. If the judgment result is A, then the next node will be C_i^A and the other way around. By contrast, there is not conditional branch in processing nodes, and they always refer to C_i^A as their next node.

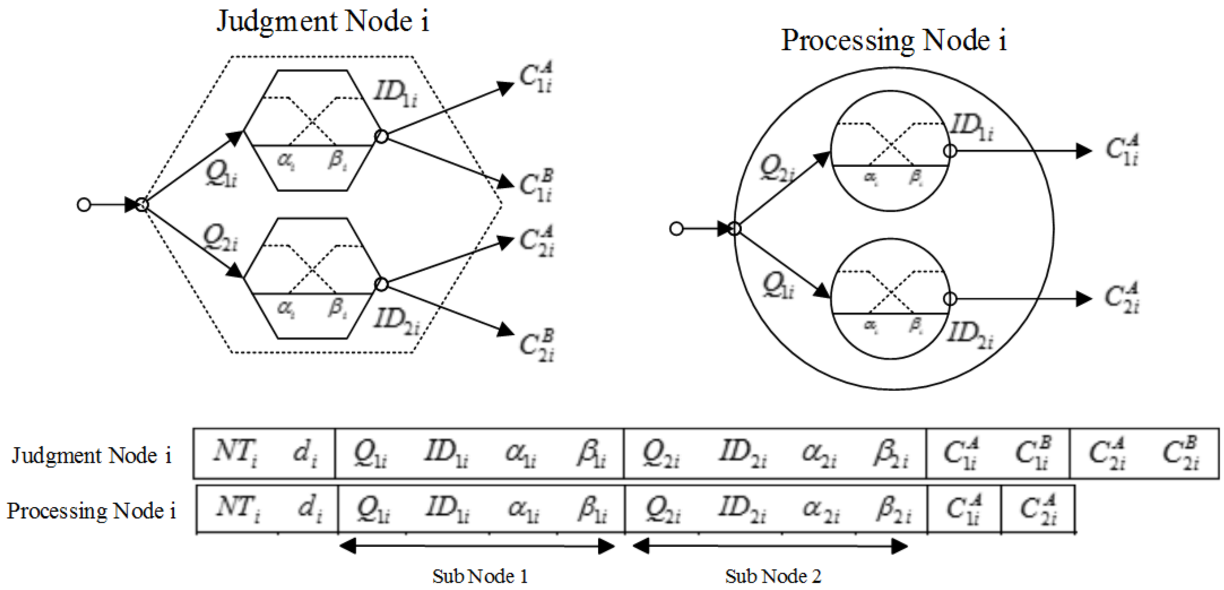

Fig. 2 Judgment and Processing Nodes' Structure

B. RL in Fuzzy GNP-RL

The learning phase of this model is based on Sarsa algorithm [17]. As shown in Fig. 2, in the proposed GNP model, each node consists of two sub nodes, and one of them is chosen according to their Q values. Here, each node is considered as "state" and the selection of the sub node is defined as "action".

The sub nodes are chosen using ϵ-greedy policy. Based on that, the sub node with the greater Q value is selected with probability of 1-ϵ and the other way around. After choosing the appropriate sub node and performing its task, the Q values are updated as:

$$Q_{ip} \leftarrow Q_{ip} + \alpha (r + \gamma Q_{jq} - Q_{ip}) \tag{1}$$

where Q_{ip} shows the Q value of the current sub node, and Q_{jq} shows the Q value of the next sub node. α is learning rate, γ is the discount rate and r is obtained reward since the previous processing node . In this regard, choosing a sub node which makes profit will increase its Q value, and consequently will increase its probability to be chosen in the following transitions.

Adding RL provides GNP with this ability to have online learning [9]. In this way, the learning is not confined to evolution phase (offline learning), and during calculation of the fitness function of each individual the system can benefit from that.

C. Node Transition in Fuzzy GNP-RL

At first, the procedure starts with the start node, and then based on the type of the next node the decision varies as explained in the following.

1) *Judgment Nodes*: if the node i is a judgment node, a technical index is selected according to its identification number (ID_i). Each technical index has its own Importance Index (IMX) which shows whether buying or selling signal is more likely to appear. The result obtained by IMX will be used later in the next processing node to generate trading signal. As an example, in Fig. 3 the IMX for Rate of Deviation (ROD) index is indicated.

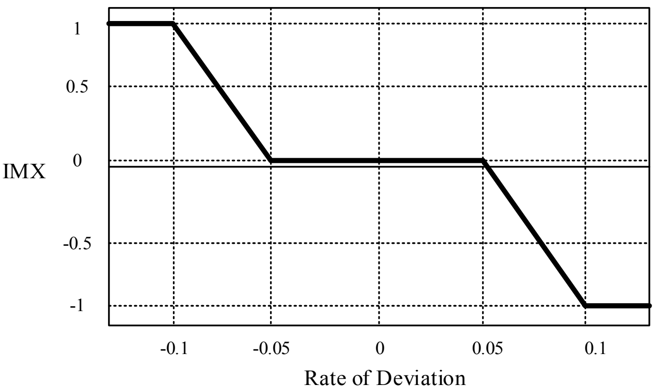

Fig. 3 IMX for Rate of Deviation (ROD) [18]

In the next step, the model should determine the next transition node using fuzzy judgment method. As shown in gene representation, each sub node has two fuzzy parameters which are used to create its membership function. By having α_i and β_i the membership function of judgment and processing nodes will be created as it can be seen in Fig. 4.

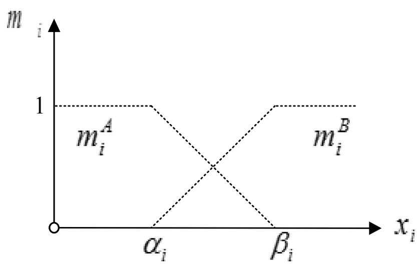

Fig. 4 Membership Function

The procedure of determining the next node using the membership function is as:

- If index value < α_i then the next node would be C_i^A
- If index value > β_i then the next node would be C_i^B
- If α_i< index value < β_i then the next node will be chosen probabilistically. So, the next node would be C_i^A with the probability of $m_i^A(x_i)$, and C_i^B will be selected with the probability of $m_i^B(x_i) = 1 - m_i^A(x_i)$, where $m_i^A(x_i)$ and $m_i^B(x_i)$ show index value's membership degree in A and B.

2) *Processing Nodes*: in processing nodes, it is necessary to calculate A_t as presented in (2) in order to generate trading signal.

$$A_t = \frac{1}{|I'|} \sum_{i' \in I'} IMX\,(i') \tag{2}$$

where I' is the set of judgment nodes being visited until the last processing node. In this paper, inspired by [14], we extended the GNP-RL and applied fuzzy processing nodes. In this case, we can generate trading signal using membership function of processing nodes which are exactly similar to Fig. 4. Using fuzzy processing nodes obliges by providing ternary signals. In fact, as it can be seen in following, using fuzzy processing node makes it possible to generate No Trade signal. This possibility helps the model seem more realistic on the ground that in a lot of situations not trading maybe more beneficial in comparison with buying or selling. To summarize, generating signal in fuzzy processing nodes is as:

In processing nodes with $ID_i = 1$ (Buying Nodes):

- If $A_t < \alpha_i$ then No Trade
- If $A_t > \beta_i$ then Buy
- If $\alpha_i < A_t < \beta_i$ then No Trade with the probability of $m_i^A(x_i)$ and Buy with probability of $m_i^B(x_i) = 1 - m_i^A(x_i)$

In processing nodes with $ID_i = 2$ (Selling Nodes):

- If $A_t < \alpha_i$ then Sell
- If $A_t > \beta_i$ then No Trade
- If $\alpha_i < A_t < \beta_i$ then Sell with the probability of $m_i^A(x_i)$ and No Trade with probability of $m_i^B(x_i) = 1 - m_i^A(x_i)$

After generating the signal, the model transits to C_i^A

D. Operators of Fuzzy GNP-RL

Fuzzy GNP-RL uses genetic operators such as Crossover and Mutation in the evolution phase. For executing operations, the individuals are selected as parents by Tournament Selection method. The Crossover operator is done as explained below.

- Two parents will be selected using tournament selection.
- One node is selected from the first parent with the probability of p_c.
- Selected node will be exchanged with its corresponding node in the other parent.
- Two obtained offspring will be considered as individuals of next generation.

The Mutation operator is executed on one individual. The parameters ID_i, C_i^A and C_i^B are mutated as follows.

- One individual is selected using Tournament selection as a parent
- Each parameter of selected individual is chosen for mutation with the probability of p_m
- Selected parameters are changed randomly
- New individual will be considered as an individual of next generation

For mutating α_i and β_i, we use the method proposed in [14]. In this method, these parameter are not mutated uniformly, but by passing through generation they change less. The process of mutating fuzzy parameters is explained below.

- Fuzzy parameters of each sub node is selected with the probability of p_m
- Selected parameters are mutated using (3).

$$x_i' = \begin{cases} x_i + \Delta(t, UB - x_i), & if\ \xi = 0 \\ x_i - \Delta(t, x_i - LB), & if\ \xi = 1 \end{cases} \tag{3}$$

where x_i' and x_i are fuzzy parameters after and before mutation, and ξ is a random digit. UB and LB are upper bound and lower bound of the parameter, and t is the current generation number. In addition, $\Delta(t, y)$ is calculated using (4).

$$\Delta(t, y) = y\left(1 - r^{\left(1-\frac{t}{T}\right)^2}\right) \tag{4}$$

where T is the maximum number of generations.

- New parameters constitute the parameters of mutated individual which transfers to new generation.

Finally, the flowchart of evolution phase and extracting trading rules is indicated in Fig. 5.

E. Designing the Frame Structure

After generating trading rules obtained by Fuzzy GNP-RL, we use them in a frame structure to develop our system. To design the frame-based system, we have used Kappa-PC software. For each stock, a frame is designed having slots which consists of price's time series and extracted rules. Additionally, there is a frame named "System" which starts the system and exposes the generated signal. The procedure of generating trading signals in this frame-based system is shown in Fig. 6.

III. Application of the System In Tehran Stock Exchange

In this section, we have used the proposed Fuzzy GNP-RL to extract trading rules for ten stocks traded in TSE. These stocks are selected among stocks with the highest liquidity in the recent years, and are chosen from different sectors of the market in order to keep the portfolio well-diversified.

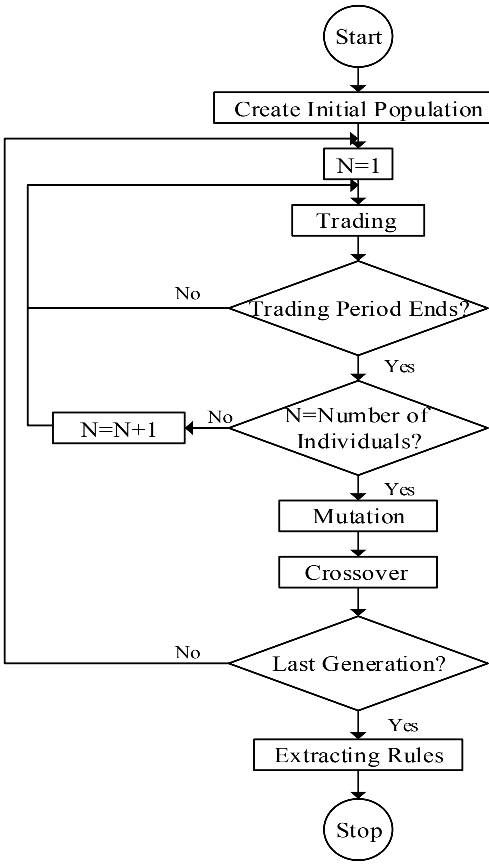

Fig. 5 Flowchart of Evolution Phase and Extracting Rules [18]

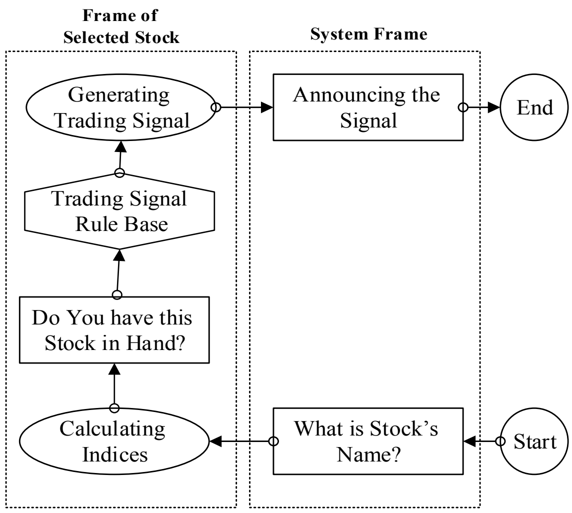

Fig. 6 Procedure of Generating Signals in the Frame Structure

In evolution phase of the Fuzzy GNP-RL, we use a window with the length of 25 days to evaluate the performance of individuals in this time period by calculating their fitness function. To make the proposed model more realistic, this time window shifts through time. In this case, the individuals will adapt to price's trend changes. In detail, every two iterations the time window shifts one day ahead.

A. Fitness and Reward

The reward that is used for updating Q values in RL is calculated by (5). In that the Reward needs selling price to be calculated, it occurs only in selling processing nodes.

$$\text{Reward} = \text{Selling Price} - \text{Buying Price} \qquad (5)$$

Moreover, the fitness function which is used to evaluate individuals in the evolution phase is as follows.

$$\text{Fitness} = \text{Total Profit in the Time Window} \qquad (6)$$

B. Technical Indices

Technical indices are used in judgment nodes to specify the next node, and to estimate the IMX value. In this paper, we used eight kinds of technical indices which were:

- Relative Strength Index (RSI),
- Rate of Deviation (ROD),
- Rate of Change (ROC),
- Volume Ratio (VR),
- Stochastics,
- Ranked Correlation Index (RCI),
- Golden/Dead Cross,
- Moving Average Convergence Divergence.

The first six of them are calculated in three different time periods with the length of 5, 10 and 15 days. As a result, totally we had 20 (6*3+2=20) technical indices for the judging task. Furthermore, MACD and Golden/Dead Cross indicators are calculated by using 5-day Moving Average and 25-day Moving Average of prices.

C. Data

In order to generate trading rules, the model evolves through training time period, and then the performance of the system is evaluated in testing time period. The training and testing time periods' length for selected stocks are clarified in the following.

- Training Period: consists of 250 trading days
- Testing Period: consists of 125 trading days ending in January 2, 2016.

D. Parameter Settings

The parameters used in the proposed Fuzzy GNP-RL model are determined through related studies [15], [19] and shown in Table I.

The initial connection between nodes, the identification number of sub nodes and fuzzy parameters are determined randomly in first generation. Moreover, the initial Q values are set as zero.

E. Results

As it can be seen in following, the proposed model is applied to generate trading signal for 10 companies listed in Table II. At first, the trading rules are extracted from the elite individual evolved over the training time period, and then the rules are applied to generate trading signal in testing period. The results show that the proposed model outperforms the Buy and Hold (B&H) strategy.

TABLE I
PARAMETER SETTING

Population size	301	Number of generations	500
Mutation	180	Number of nodes	61
Crossover	120	Number of judgment nodes	40
Elite	1	Number of processing nodes	20
Mutation rate (p_m)	0.02	Learning rate (α)	0.1
Crossover rate (p_c)	0.1	Discount rate (γ)	0.4
Tournament size	2	Є-greedy parameter	0.1

TABLE II
RETURNS OBTAINED BY USING FUZZY GNP-RL

Company	B&H	B&S
Pars Khodro Co.	-11.91%	-5.56%
Telecommunication Company of Iran	14.07%	5.07%
Shazand Petrochemical Co.	-3.59%	2.81%
Ghadir Investment Co.	-26.65%	1.7%
Bank Saderat Iran	-7.58%	-6.83%
Iran Construction Investment Co.	-38.36%	-25.01%
Azarab Industries Co.	41.25%	21.26%
Darou Pakhsh Pharmaceutical Manufacturing Co.	-9.35%	3.91%
Islamic Republic of Iran Shipping Lines (IRISL)	2.43%	7.49%
National Iranian Lead & Zinc Co.	26.72%	39.39%
Average	**-1.30%**	**4.42%**

The results show that applying developed system makes a positive return in this specific time period although the prices have a downward trend.

IV. CONCLUSION

In this paper, a hybrid expert system to generate stock trading signals is developed applying trading rules in the frame-based structure designed for the system. These trading rules are extracted by using Fuzzy GNP-RL. Extracted rules are based on technical indices of the stock prices. Applying fuzzy processing nodes provides the proposed model with the ability to generate ternary signals. In order to evaluate the performance of the developed system, it is used to generate signals for 10 stocks which are traded in TSE. The results indicated that the developed system outperforms the Buy and Hold strategy. In fact, using fuzzy nodes along with RL contributes to generate more accurate trading signals.

REFERENCES

[1] Seng-cho, T.C., et al., A stock selection DSS combining AI and technical analysis. Annals of Operations Research, 1997. 75: p. 335-353.
[2] Bauer, R.J., Genetic algorithms and investment strategies. Vol. 19. 1994: John Wiley & Sons.
[3] Koza, J.R., Genetic programming: on the programming of computers by means of natural selection. Vol. 1. 1992: MIT press.
[4] Holland, J., Adaption in natural and artificial systems. Ann Arbor MI: The University of Michigan Press, 1975.
[5] Mousavi, S., A. Esfahanipour, and M.H.F. Zarandi, A novel approach to dynamic portfolio trading system using multitree genetic programming. Knowledge-Based Systems, 2014. 66: p. 68-81.
[6] Esfahanipour, A. and S. Mousavi, A genetic programming model to generate risk-adjusted technical trading rules in stock markets. Expert Systems with Applications, 2011. 38(7): p. 8438-8445.
[7] Hirasawa, K., et al. Comparison between genetic network programming (GNP) and genetic programming (GP). in Evolutionary Computation, 2001. Proceedings of the 2001 Congress on. 2001. IEEE.

[8] Izumi, Y., et al. Trading rules on the stock markets using genetic network programming with candlestick chart. in Evolutionary Computation, 2006. CEC 2006. IEEE Congress on. 2006. IEEE.
[9] Mabu, S., K. Hirasawa, and J. Hu, A graph-based evolutionary algorithm: genetic network programming (GNP) and its extension using reinforcement learning. Evolutionary Computation, 2007. 15(3): p. 369-398.
[10] Mabu, S., et al. Stock trading rules using genetic network programming with actor-critic. in Evolutionary Computation, 2007. CEC 2007. IEEE Congress on. 2007. IEEE.
[11] Chen, Y., et al., A portfolio optimization model using Genetic Network Programming with control nodes. Expert Systems with Applications, 2009. 36(7): p. 10735-10745.
[12] Chen, Y., S. Mabu, and K. Hirasawa, A model of portfolio optimization using time adapting genetic network programming. Computers & operations research, 2010. 37(10): p. 1697-1707.
[13] Chen, Y., et al. Constructing portfolio investment strategy based on time adapting genetic network programming. in Evolutionary Computation, 2009. CEC'09. IEEE Congress on. 2009. IEEE.
[14] Sendari, S., S. Mabu, and K. Hirasawa. Fuzzy genetic Network Programming with Reinforcement Learning for mobile robot navigation. in Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on. 2011. IEEE.
[15] Mabu, S., et al., Enhanced decision making mechanism of rule-based genetic network programming for creating stock trading signals. Expert Systems with Applications, 2013. 40(16): p. 6311-6320.
[16] Chen, Y. and X. Wang, A hybrid stock trading system using genetic network programming and mean conditional value-at-risk. European Journal of Operational Research, 2015. 240(3): p. 861-871.
[17] Chen, Y., et al. Trading rules on stock markets using genetic network programming with sarsa learning. in Proceedings of the 9th annual conference on Genetic and evolutionary computation. 2007. ACM.
[18] Chen, Y., et al., A genetic network programming with learning approach for enhanced stock trading model. Expert Systems with Applications, 2009. 36(10): p. 12537-12546.
[19] Mabu, S., et al. Generating stock trading signals based on matching degree with extracted rules by genetic network programming. in SICE Annual Conference 2010, Proceedings of. 2010. IEEE.