2009 International Conference on Adaptive and Intelligent Systems

A Comparison of Genotype Representations to Acquire Stock Trading Strategy Using Genetic Algorithms

Kazuhiro Matsui
Dept. of Computer Science
College of Engineering, Nihon University
Koriyama, Japan
matsui@cs.ce.nihon-u.ac.jp

Haruo Sato
Dept. of Computer Science
College of Engineering, Nihon University
Koriyama, Japan
sato@cs.ce.nihon-u.ac.jp

***Abstract*—Automatic trading methods are important issues in recent financial markets. In this paper, we compare some genotype coding methods of technical indicators and their parameters to acquire stock trading strategy using genetic algorithms (GAs). In previous works, the locus-based representation is widely used for encoding technical indicators on chromosomes in GAs, and the direct coding is also widely adopted for encoding the parameters of the indicators. However, these conventional methods are not so effective for the GA search. Therefore, we propose a new genotype coding methods, namely the allele-based indirect coding. We examine the performance of the proposed and conventional coding methods in stock trading of twenty companies in the first section of the Tokyo Stock Exchange for recent ten years. In our empirical results, the allele-based indirect coding is superior to the other ones both on the cumulative profits and the computational costs.**

***Keywords*-Stock trading; Genetic Algorithm; Algorithmic Trade;**

I. Introduction

Automatic trading methods, such as algorithmic trading, are expanding rapidly in recent financial markets. Various works are found in applications of computational intelligence methodologies in finance [1]. In these methodologies, evolutionary computation, such as genetic algorithms (GAs)[2], is promising because of their robustness, flexibility and powerful ability for search.

Some works have been done for acquiring trading strategy using evolutionary computation [3], [4], [5], [6]. Their methods are based on technical analysis, which is one of the two basic types of approaches in stock trading. Technical analysis is a technique to attempt the forecast of the future direction of prices by analyzing past market data, such as price and volume. The other type of the approaches in stock trading is fundamental analysis, which focuses on analyzing financial statements and management. The above works to acquire trading strategy adopted technical analysis because it is easily applied to automatic trading in comparing with fundamental analysis.

Various kinds of genotype-phenotype coding are proposed in these works. However, it is not clear which representation is better than others. We compare some genotype representations in terms of coding for technical indicators and their parameters in this paper. In conventional coding methods for technical indicators, locus-based representation has been widely used. This representation causes chromosomes in GAs to be too long when many technical indicators are used. The conventional coding methods also used simple binary chromosomes for parameters of technical indicators. However, the efficiency of the binary coding is low for searching spaces of parameters. Therefore, we propose a new genotype representation to solve these problems. Our representation is called the allele-based indirect coding. We compare it with some conventional methods and verify the effectiveness of our method.

This paper is organized as follows: In Section 2, we summarize the concept of technical analysis in stock trading. We describe the details of genotype representation in Section 3. Section 4 contains our trading method and the empirical results are followed in Section 5. Sections 6 and 7 are discussion and conclusions respectively.

II. Technical Analysis in Stock Trading

There are two basic approaches to analyze markets: fundamental analysis and technical one. The former is based on analyzing financial statements, management, and competitive advantages of companies. The latter is based on the past patterns of changes of share prices. In technical analysis, many indicators are used for trading. They are calculated from past share prices. Generally, technical indicators have some parameters. For example, moving average has a parameter, namely *period*, which is used as the denominator of averaging calculation. Various derived indicators, such as *10-days moving average*, *50-days moving average*, *etc.*, are defined with the period. In this paper, we use many technical indicators and their parameters for automatic trading.

Many technical indicators are known in traders, but it is difficult to select optimal indicators for trading. Furthermore, it is also hard to determine parameters for the selected indicators. In this paper, we apply GAs to this problem. Both technical indicators and their parameters are encoded on

978-0-7695-3827-3/09 $25.00 © 2009 IEEE
DOI 10.1109/ICAIS.2009.29

chromosomes of individuals in GAs and the genetic search is applied to acquire effective combinations of technical indicators and their parameters for trading. The aims of this paper are to compare various methods of genotype representation and to clarify the effectiveness of our new method, which is described in the next section.

III. Genotype Representation

A. Related Works

Some related works has been done on automatic trading strategy using evolutionary computation methodologies. A method to search the effective combinations of technical indicators using genetic algorithms was proposed [4]. This method is similar to the locus-based representation described in Section 3.3. Its genetic search was limited to technical indicators. Their parameters were fixed through the search.

On the other hand, some methods to search the optimal parameters to calculate technical indicators using genetic algorithms were proposed [5], [6]. These methods searched only the parameters. Their technical indicators were fixed through the genetic search. These genotype representations correspond to the direct coding described in the following section.

B. Parameter Encoding

It is necessary to encode parameters on chromosomes of individuals for genetic search of trading strategy. In the related works, such as [5] and [6], binary coded GAs are mainly used for this aim. However, it involves a shortcoming. Generally, binary coded GAs divide their search-ranges at regular intervals and assign each value to each binary code. However, it is often not desirable to divide the range at regular intervals. For examples, suppose that binary coded GAs search the period of the moving average of prices. In comparing among shorter periods, such as four and five days, it may causes different profits in short-term trading. On the other hand, in comparing among longer periods, such as 99 and 100 days, it will be expected that the difference of profits is little although the difference of these periods is same as one day. Thus, simple binary coding is not always suitable for the search of parameters. We refer the conventional method of binary coding as *the direct coding* in the following sections.

In this paper, we propose a new coding method of parameters. Generally, there often are some particular values which are widely used to calculate technical indicators. Therefore, we restrict the ranges of the genetic search to the set of these values. For examples, in the above case of the moving average of prices, we are able to restrict the range to a set of values $\{5, 10, 20, 50, 100\}$. This method makes the space of the genetic search to be smaller than the conventional one. Thus, the efficiency of the genetic search is expected to be higher than the conventional one, and the computational cost is reduced. We refer the new method of coding as *the indirect coding* in this paper.

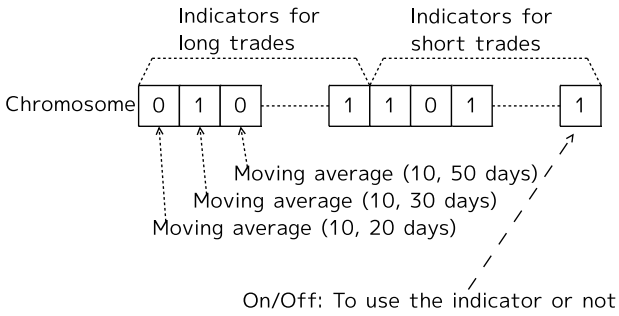

Figure 1. Locus-based representation.

C. Locus-based Representation

In this paper, we compare two types of the genotype representation of technical indicators. The first is the locus-based representation, as shown in Figure 1. This representation assigns each locus, which is a bit-position on chromosomes, to each technical indicator. For example, the first bit is assigned to the indicator "*the crossover of moving average between ten and twenty days*" in Figure 1. The indicator is used for trading when the assigned bit is "1," and it is not used in the opposite case. Note that technical indicators and their parameters are combined to single bits and the parameters are encoded in the indirect coding.

The chromosome length in the locus-based representation is equal to the total numbers of candidates of technical indicators which are combined to their parameters. Thus, the length becomes long when the numbers of the candidates increase. We can apply ordinary binary-coded GAs easily to the locus-based representation because the chromosomes are coded in binary strings.

D. Allele-based Representation

The second type of genotype coding is the allele-based representation. Figure 2 shows the concept of it. An allele is a value on a locus which is a position on the chromosome. In the allele-based representation, the allele takes various values which represent technical indicators and their combined parameters in the indirect coding. For example, Allele #1 is assigned to the indicator "*the crossover of moving average between ten and twenty days*" in Figure 2.

Generally, the allele-based representation makes the length of chromosome to be shorter than the locus-based representation. The total numbers of alleles is identical to the total numbers of candidates of technical indicators which are combined to their parameters. Note that it is necessary to determine the length of chromosomes in advance.

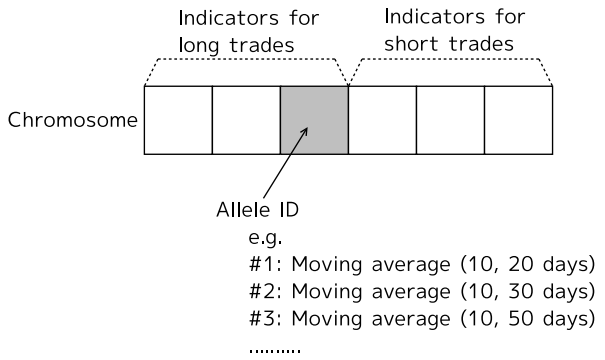

Figure 2. Allele-based representation.

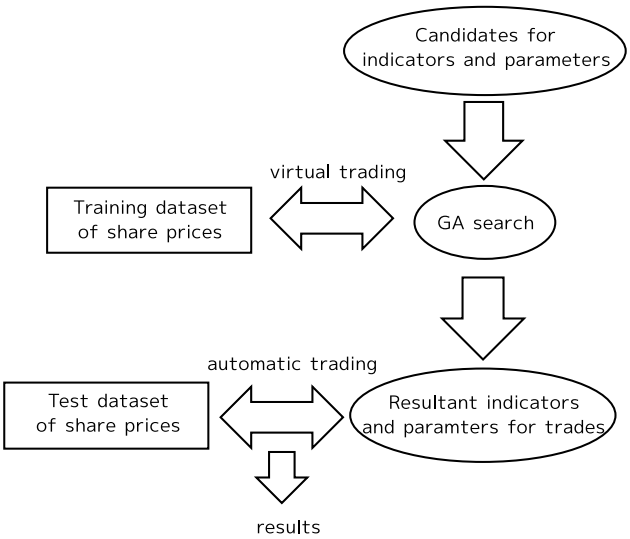

Figure 3. Overview of our system.

IV. Methods

A. Overview

We show the overview of our system in Figure 3. The flow of our method is the following:

1) Prepare a set of historic share prices and divide it to the training dataset and the test one.
2) Determine candidates of technical indicators and their parameters.
3) Apply the genetic search to find effective indicators and their parameters for stock trading on the training dataset.
4) Run the simulation of automatic trading for the test dataset with the indicators and their parameters which have been found by the previous genetic search.

The aim of this process is to maximize the profit on the test dataset, not on the training dataset. Thus, it is important to keep off overfitting on the training dataset.

Note that the GA is applied only on the training phase (Step 3), not on the test phase (Step 4).

The trading rules adopted in our system are applied *daily*. When a trading rule is matched in a day, the system opens or closes a position *at the opening price on the next day*.

B. Technical Indicators

We use the following technical indicators in this paper:

1) Simple Moving Average Crossover (SMA)
The SMA is a simple average of closing prices for the last n days. We define the SMA as follows:

$$\mathrm{SMA}_n(t) = \frac{1}{n}\sum_{i=0}^{n-1} c_{t-i}, \tag{1}$$

where c_t is the closing price at the day t, n is the parameter which determines the period to calculate the SMA.
In our experiments, we apply the following rules: For a long trade, enter when a shorter-period SMA crosses a longer-period SMA and exit when the opposite occurs. A short trade is the contrary of the long one.

2) Exponential Moving Average Crossover (EMA)
The EMA is an exponentially weighted average of closing prices for the last n days, as follows:

$$\mathrm{EMA}_n(t) = \begin{cases} \mathrm{SMA}_n(t) & (t = 0) \\ \mathrm{EMA}_n(t-1) & \\ \quad + \alpha\left(c_t - \mathrm{EMA}_n(t-1)\right) & (t \geq 1), \end{cases} \tag{2}$$

where n is the period parameter, and $\alpha\,(= 2/(1+n))$ is the weight. In our experiments, we use the same crossover rule as the above SMA for trades.

3) Bollinger Band (BB)
The BB is an indicator based on the standard deviation of the change of prices, as follows:

$$\mathrm{BB}_n(t) = \mathrm{SMA}_n(t) \pm \beta\sigma, \tag{3}$$

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=0}^{n-1}\left(c_t - \mathrm{SMA}_n(t)\right)^2}, \tag{4}$$

where β is the factor of the standard deviation. The parameters of the BB are the period n and the factor β.
In our experiments, we apply the following rules: For a long trade, enter when the closing price crosses the upper line of the BB and exit when the closing price crosses $\mathrm{SMA}_n(t)$. For a short trade, enter when the closing price crosses the lower line of the BB and the exit rule is the same as the long one.

4) Price Channel Breakout (PCB)
The PCB is an indicator based on the trading range of

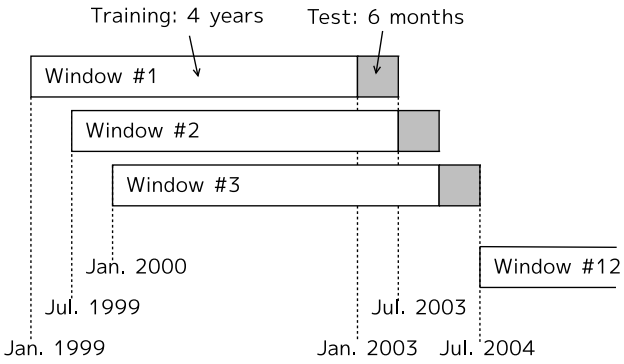

Figure 4. Concept of sliding window.

the last n days, as follows:

$$\mathrm{U}_n(t) = \max\{h_{t-i} | 1 \leq i \leq n\}, \quad (5)$$

$$\mathrm{L}_n(t) = \min\{l_{t-i} | 1 \leq i \leq n\} \quad (6)$$

where $\mathrm{U}_n(t)$ is the upper bound of the PCB, $\mathrm{L}_n(t)$ is the lower bound, h_t is the highest price and l_t is the lowest one at Day t. The PCB has one parameter, namely the period n.

In our experiments, we apply the following rules: For a long trade, enter when $c_t > \mathrm{U}_n(t)$ and exit when $l_t < \frac{1}{2}(\mathrm{U}_n(t) + \mathrm{L}_n(t))$. For a short trade, enter when $c_t < \mathrm{L}_n(t)$ and exit when $h_t > \frac{1}{2}(\mathrm{U}_n(t) + \mathrm{L}_n(t))$.

V. Experiments

A. Setups

We applied our method on ten years of price data from the first trading day of 1999 to the last trading day of 2008. We selected twenty companies at random from the components of the Nikkei 225, which is a stock market index for the Tokyo Stock Exchange, and these issues are used for trading simulation.

A sliding-window technique is applied for our experiments, as shown in Figure 4. A window consists of training and test. The former is a genetic search for four years. This search is applied to find effective indicators and their parameters for stock trading. The latter is a trading test for six months just after the training period. This test is applied to evaluate the indicators and their parameters which are found by the genetic search. We have 12 windows whose start days slide six months in the ten years.

The initial principal is 5,000,000 JPY, the trading unit is minimal, *i.e.*, a round lot of each issue. A commission of one trade is assumed at 1,000 JPY.

In Table I, we show the technical indicators used in our experiments. Our experiments use daily price data and Each technical indicators are calculated from daily prices. The period for each indicator in the indirect coding takes a value from a set of $\{5, 10, 15, 20, 25, 30, 50, 75, 100, 200\}$, and the factor for the Bollinger band also takes a value from a set of $\{1.0, 1.5, 2.0, 2.5, 3.0\}$. Since SMA has two parameters, 45 indicators are defined as *5-and-10 days SMA, 5-and-15 days SMA, ..., 100-and-200 days SMA*. In same ways, 45 EMAs, 50 BBs and 10 PCBs are also defined. Thus, the total number of indicators with their parameters is 150. In our genotype representation, each indicators can be applied for *long* and *short* positions. Therefore, the total size of indicators is 300 finally in the indirect coding.

Table I
Technical Indicators.

Indicators	Parameters
Simple Moving Average (SMA)	shorter period, longer period
Exponential Moving Average (EMA)	shorter period, longer period
Bollinger Band (BB)	period, factor
Price Channel Breakout (PCB)	period

On the other hand, in the direct coding, we use 8-bits binary coding method. The period for each indicators can take a value from 1 to 256 days and the range of the factor for BB is $[1.0, 3.0]$.

The setups of our GA are the following: the population size is 50 and the searching generation is 5000. The minimal generation gap model [7] is used for selection strategy. The uniform crossover is applied for recombination of individuals with 100% crossover probability. The random-replace mutation is used for the allele-based coding and the bit-flip mutation is done for the locus-based coding. The mutation probability is $1/L$, where L is the length of chromosomes, in both mutation operators. The fitness of each individual is the total interest obtained in the trading period.

In our experiments, we compare three methods: the allele-based indirect coding, the locus-based indirect coding, and the locus-based direct coding. The third one is the representative method in previous related works.

B. Results

We obtained the results from our experiments through the 12 windows, as shown in Table II. The profit and the draw down are represented in 1,000 JPY. The worst draw down is the worst loss from a series of loss trades. The average CPU time is the average of computational time on the 12 windows of experiments in which we used Intel Xeon 3.0GHz.

In this table, the profit from the allele-based indirect coding is largest and the computational cost of this method is lowest in the three methods. The difference of the CPU time between indirect and direct coding is very large.

We show the progress of cumulative profits of the three method in Figure 5. Although all the methods suffered loss in earlier days, the loss was recovered later.

In Tables III, IV, V, we summarized the comparison of results between training and test. Since the training was applied for four years and the test was done for six months,

Table II
EXPERIMENTAL RESULTS.

parameter coding	indirect	indirect	direct
indicator coding	allele-based	locus-based	locus-based
Num. of trades	266	217	186
Total profit	2,370	1,319	1,628
Worst draw down	354	468	285
Avg. CPU time	1min. 09sec.	1min. 54sec.	16min. 12sec.

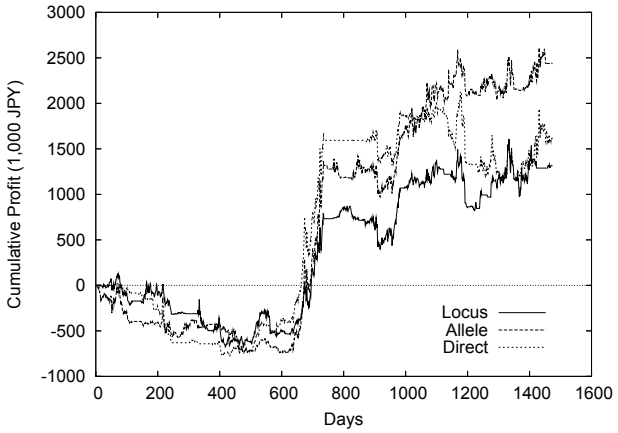

Figure 5. Cumulative profit.

the both results were converted into the value of one year and averaged over the 12 windows. The ratio in these tables is the ratio of the test to the training. In the three tables, the profits in the test were much reduced from that in the training.

VI. DISCUSSION

In our experiments, the allele-based indirect coding obtained the largest profit and its computational cost was lowest in the three methods. From this result, it turned out that the allele-based indirect coding was very effective for automatic stock trading using genetic algorithms.

The difference of the computational cost was very large because the indirect coding only needs restricted sets of values for parameters whereas the direct coding needs very large sets of parameter values.

The number of trades in the direct coding was less than that in the indirect coding. This indicates that the obtained indicators and their parameters by GA were overfitted to the training datasets and it was hard to fit the test datasets. Tables III, IV, V also imply overfitting tendency in the training.

From Figure 5, it turned out that the profit in each window of experiments changed considerably. In particular, some earlier sets suffered losses. For example, Table VI shows the results in the latter half of 2003. In this table, all the methods underwent losses. Our methods, however, obtained considerable profit in the final set, which is the latter half of 2008 and in which the global financial crisis has occurred, as shown in Table VII.

Table III
COMPARISON BETWEEN TRAINING AND TEST: ALLELE-BASED CODING.

	Training	Test	Ratio
Avg. Num. of trades	52.5	44.3	84%
Avg. Profit	104.7	39.5	38%

Table IV
COMPARISON BETWEEN TRAINING AND TEST: LOCUS-BASED CODING.

	Training	Test	Ratio
Avg. Num. of trades	51.0	36.2	71%
Avg. Profit	98.2	22.0	22%

Table V
COMPARISON BETWEEN TRAINING AND TEST: DIRECT CODING.

	Training	Test	Ratio
Avg. Num. of trades	40.2	31.0	77%
Avg. Profit	106.5	27.1	25%

Table VI
RESULTS IN THE LATTER HALF OF 2003.

parameter coding	indirect	indirect	direct
indicator coding	allele-based	locus-based	locus-based
Num. of trades	21	37	37
Total profit	-166	-144	-545
Worst draw down	112	195	235

VII. CONCLUSIONS

We proposed the allele-based indirect representation to acquire stock trading strategy using genetic algorithms and we compared three types of genotype representation. In our experiments, the allele-based indirect coding outperformed the other ones. In particular, the indirect coding is superior to the direct coding in computational costs.

Future problems are the following: To keep off the overfitting in the training and to reduce the fluctuations of profits through windows of experiments. Also, it is important to add other technical indicators since we used only four types of indicators in our experiments.

Table VII
RESULTS IN THE LATTER HALF OF 2008.

parameter coding	indirect	indirect	direct
indicator coding	allele-based	locus-based	locus-based
Num. of trades	6	13	16
Total profit	281	153	461
Worst draw down	154	282	282

ACKNOWLEDGMENT

This work was partially supported by the Grant-in-Aid for Scientific Research (C) 20500215, Japan Society for the Promotion of Science.

REFERENCES

[1] A. Brabazon and M. O'Neill, "An Introduction to Evolutionary Computation in Finance," *IEEE Computational Intelligence Magazine*, pp. 42–55, Nov., 2008.

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

[3] A. Hryshko and T. Downs, "An Implementation of Genetic Algorithms as a Basis for a Trading System on the Foreign Exchange Market," *Proc. Congress of Evolutionary Computation*, pp. 1695–1701, 2003.

[4] M.A.H. Dempster and C.M. Jones, "A Real-time Adaptive Trading System Using Genetic Programming," *Quantitative Finance*, 1, pp. 397–413, 2001.

[5] D. de la Fuente, A. Garrido, J. Laviada and A. Gomez, "Genetic Algorithms to Optimise the Time to Make Stock Market Investment," *Proc. of Genetic and Evolutionary Computation Conference*, pp. 1857–1858, 2006.

[6] A. Hirabayashi, C. Aranha and H. Iba, "Optimization of the Trading Rule in Foreign Exchange using Genetic Algorithms," *Proc. of the 2009 IASTED Int'l Conf. on Advances in Computer Science and Engineering*, 2009.

[7] H. Satoh, M. Yamamura and S. Kobayashi, "Minimal Generation Gap Model for GAs Considering Both Exploration and Exploitation," *Proc. of 4th Int'l Conf. on Soft Computing*, pp. 494–497, 1996.