Received August 4, 2014, accepted August 19, 2014, date of publication August 26, 2014, date of current version September 5, 2014.

Digital Object Identifier 10.1109/ACCESS.2014.2352261

A Rule-Based Dynamic Decision-Making Stock Trading System Based on Quantum-Inspired Tabu Search Algorithm

YAO-HSIN CHOU[1], SHU-YU KUO[1], CHI-YUAN CHEN[2], AND HAN-CHIEH CHAO[2], (Senior Member, IEEE)
[1]Department of Computer Science and Information Engineering, National Chi Nan University, Nantou 545, Taiwan
[2]Department of Computer Science and Information Engineering, National Ilan University, Yilan 260, Taiwan

Corresponding author: H.-C. Chao (hcc@niu.edu.tw)

ABSTRACT Heuristic methods or evolutionary algorithms (such as genetic algorithms and genetic programs) are common approaches applied in financial applications, such as trading systems. Determining the best time to buy or sell stocks in a stock market, and thereby maximizing profit with low risks, is an important issue in financial research. Recent studies have used trading rules based on technique analysis to address this problem. This method can determine trading times by analyzing the value of technical indicators. In other words, we can make trading rules by finding the trading value of technique indicators. An example of a trading rule would be, if one technical indicator's value achieves the setting value, then either buy or sell. A combination of trading rules would become a trading strategy. The process of making trading strategies can be formulated as a combinational optimization problem. In this paper, we propose a novel method for applying a trading system. First, the proposed method uses the quantum-inspired Tabu search algorithm to find the optimal composition and combination of trading strategies. Second, this method uses a sliding window to avoid the major problem of over-fitting. The experiment results of earning money show much better performance than other approaches, and the proposed method outperforms the buy and hold method (which is a benchmark in this field).

INDEX TERMS Trading system, quantum-inspired algorithm, Tabu search, decision making.

I. INTRODUCTION

Stock markets are uncertain, complicated and difficult to predict because both price and volume of stock are affected by many factors, such as market internal, fundamental and policy factors. So when trading in a stock market, there are important issues to address in order to avoid risk and maximize profit.

Investors can develop investment strategies through two kinds of stock analysis. One is fundamental analysis, and the other is technical analysis. Fundamental analysis is based on economics and financial science. It uses many economic and financial data as fundamental indicators, such as macroeconomic indicators, current ratio, foreign exchange rates, etc. On the other hand, technical analysis is based only on stock price and transaction volume. The rationale of technique analysis is that history will continue to repeat itself through the variation of price and volume, which can reflect all of the factors affecting the stock market. Many well-known technical indicators made by price and volume, like moving average (MA), stochastic oscillator (KD) and relative strength index (RSI), are commonly used in stock markets.

In this paper, we focus on technical analysis. Financial market studies are able to use technical analysis to forecast the future price direction [19], [20]. It can also generate trading rules to provide buying and selling signals. In this work, the Quantum-inspired Tabu Search Algorithm (QTS) [1] is proposed to find the optimal, or near optimal, combination of technical trading rules. So, the purpose of this research is to prove that QTS can generate trading rules, and that these rules perform well in financial markets.

The rest of this paper is organized as follows. Section II briefly reviews previous literature. Section III provides background information of technical indicators, sliding window and the QTS Algorithm. Section IV presents the details of this trading system. Section V discusses the experiment results, and we draw our conclusions in section VI.

II. RELATED WORK

Recent studies, such as evolutionary algorithms, fuzzy systems, and artificial neural networks have attempted to determine the optimal time to trade in a stock market. Two main methods have emerged: one is to forecast the price of stock and then make a buy or sell decision, and the other is to use a combination of trading rules through technical analysis to create a buy or sell signal also there are hybrid method of those two [2]–[4].

The difference between these two methods is that the first may predict the point of price, while the second will find rising or falling fluctuations of the stock price. Related researches of the first method include using a hybrid method [20] of fuzzy [31] and PSO [32] for prediction, and using root mean square error as a reference value of performance evaluation. Researches related to the second method include IC-Yeh's [18] proposed fuzzy rule-based system (FRBS). This system uses technical indicators such as moving average indicators and moving average volume indicators as trading rules, and tunes parameters like membership function shape parameters, rule trigger thresholds and weights of conditions. It is a trading decision-making system. Some researches use evolutionary computation, such as: Genetic Algorithms (GA) [5]–[7], Genetic Programming (GP) [8]–[11], Particle Swarm Optimization (PSO) [14], [16], [17], Ant Colony Optimization (ACO) [22] and other algorithm [12], [13]. Wang uses GA [5] to select trading rules. In other words, GA is used to find the optimal combination of trading rules. In that research, spatial indexes were chosen, i.e., BIAS, RSV and WMS as its composition of rules. It also uses the concept of sliding window. As an extension of GA, GP uses a tree data structure. GP is frequently applied to this financial field. It uses its tree structure to produce profit rules. For example, standard GP [10] will generate real functions like technical indicators, price or volumes, and terminals such as price or volumes, and compare them. Repeating the above action will yield a trading rule, and its result can determine when to buy, sell, or hold stocks. The PSO [14] method tunes the parameters of the MA trading rule. In other words, particle swarm optimization is applied to optimize trading rules and maximize the trading profit by determining appropriate average durations. ACO [22] system uses technical indicators such as: 20-day moving average, KD line, stock price and trading volume as their determining factors. Specifically, in that system the trading volume is the phenomenon, the data of stochastic line KD is the visibility.

The above studies present encouraging results. However, we have noted that most researches still have some limitations. For example, they cannot outperform the simple Buy-and-Hold strategy, which is a benchmark in this financial field; in some years, they still have negative returns, and some will have good performance in training periods, but poor performance in testing periods because of the problem of over-fitting. Hence, to improve the performance results, and to avoid the problem of over-fitting, we propose a rule-based system using sliding window through a Quantum-inspired tabu search algorithm. QTS is based on the classical Tabu search and the characteristics of quantum computation [29], [30]. Our reason for choosing the QTS-algorithm is that QTS performs much better than other heuristic algorithms in optimization problems, without premature convergence. QTS-algorithm can effectively find the optimal or near optimal combination of rules, and sliding window [23] avoids the problem of over-fitting.

III. PRELIMINARY

In this section, we will describe some technical indicators, such as moving average indicators (MAI), moving average volume indicators (MVI), stochastic indicators (KD), relative strength index (RSI) and rate of change (ROC), used in our approach. We will also present the concepts of sliding window and the QTS algorithm.

A. TECHNICAL INDICATORS

Technique analysis is an action that observes the past fluctuation of price. The principles of technique analysis are that history will repeat itself, and that all factors affecting the stocks will reflect back to price and volume of stock in advance. The technical indicators consist of history data of time, open price, high price, low price, close price and transaction volumes. People will use these indicators to create distinct buying and selling signals, and use these signals to maximize profit.

1) MOVING AVERAGE INDICATOR (MAI)

This is the ratio of two continuous periods of the average closing price. The indicator is defined as the following equation (1).

$$MAI(m, n) = \frac{MA_m}{MA_n} \tag{1}$$

Where MA_m and MA_n represent m-day and n-day moving averages of closing prices. Moving average is a line of smoothing the fluctuation of price. Generally, moving average is regard as support and resistance. When short-term moving average cross long-term moving average from bottom to top means bull market. In contrast, means bear market.

2) RELATIVE STRENGTH INDEX (RSI)

This principle assumes that when closing price rises, it enhances the strength of buying, and when it falls, it enhances the strength of selling. The index is presented as equation (2).

$$RSI(t) = \frac{R}{R+F} \times 100 \tag{2}$$

Where R is the average of the t-day rise in stock prices, and F is the average of the t-day fall in stock prices. Generally speaking, when RSI lower than 30 is buying signal and higher than 70 is selling signal. It's rely on the period of RSI.

3) STOCHASTIC OSCILLATOR (KD)

Its value reflects, in the recent period, the relative location of current closing price. The formula of K line and D line is

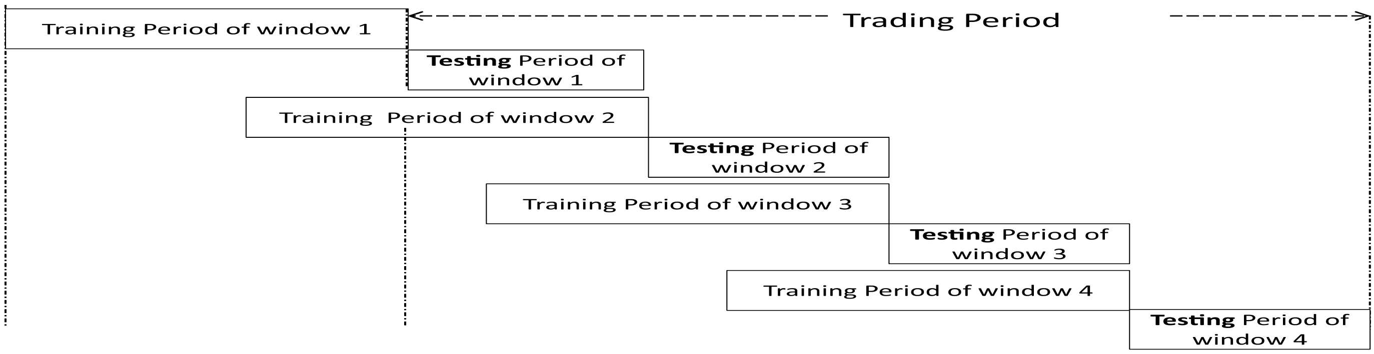

FIGURE 1. Sliding window.

defined as the following equation (3).

$$\begin{aligned} RSV(n) &= \frac{C - L_n}{H_n - L_n} \times 100 \\ \%K &= (K-1) \times \frac{2}{3} + RSV \times \frac{1}{3} \\ \%D &= (D-1) \times \frac{2}{3} + \%K \times \frac{1}{3} \end{aligned} \tag{3}$$

Where C presents the closing price of today, L_n is the lowest price, H_n shows the highest price of recent n days, (K-1) represents the K value of yesterday, and (D-1) denotes the D value of yesterday. When the K line cross the D line form the bottom to top, it means buying signal. In contrast, when the K line cross the D line form top to bottom is means selling signal.

4) RATE OF CHANGE(ROC)

This is an index that observes the change of a stock market by comparing the price of the current day and the price of a few days ago in order to evaluate the degree of changes over that period. The formula is shown as equation (4).

$$ROC(n) = \frac{C - C_n}{C_n} \times 100 \tag{4}$$

C is closing price of the current day, and C_n indicates the closing price of n days ago. When the ROC line from the negative zone breaks into the positive zone, it means that stocks have become strong and have potential. This triggers a buying signal. Conversely, when the ROC line moves from the positive zone into the negative zone, it means stocks transferred to have a weak price. This triggers a selling signal.

B. SLIDING WINDOW

In a stock market, prices undergo great changes every day. So, we propose a dynamic trading system using the sliding window method, which can avoid the problem of over-fitting in this volatile environment. In Fig.1, we use recent historical data to train the best combination of rules based on a QTS-algorithm, and use it for each corresponding testing period. We also shift the training and testing period over time.

C. QUANTUM-INSPIRED TABU SEARCH ALGORITHM

Quantum-inspired tabu search (2011) is a novel quantum-inspired algorithm based on tabu search and quantum computation. The primary characteristic of QTS is that it can efficiently increase the diversification and intensification of searching space because when QTS searches neighbor solutions, it can explore a new region by measuring the quantum for diversity. When QTS updates the quantum states by using the best and worst solutions, it can intensify the search, which allows it to more effectively explore the solution space, avoiding a negative result.

Algorithm 1: Quantum-inspired Tabu Search.

$t \leftarrow 0$
Initialize quantum population $Q(0)$
Initialize the best solution B and worst solution W
while (**not** termination-condition) **do**
  $t \leftarrow t + 1$
  Produce Neighborhood set N by multiple measurements of $Q(t-1)$
  Repair $s \in N$
  Evaluate $f(s)$
  Select the best solution and worst solution among N
  Store best solution and worst solution into B and W respectively
  Update Tabu list T
  Update $Q(T)$ by B and W
end while

QTS can solve the 0/1 knapsack problem in a short time. We also can expand this algorithm to apply to different combinational optimization problems, such as: multiple knapsack problems, travelling salesman, synthesizing optimal reversible circuits [24] and combinatorial optimization problem [25]. QTS demonstrates good results with these problems. Also, we use QTS algorithm to find the combination of trading rules in U.S.A [26] and Japan [27], and there are good performance in those stock markets, so in this research, we still use QTS algorithm to find optimal solutions. Algorithm 1 show the pseudo code of QTS algorithm.

IV. DYNAMIC STOCK TRADING SYSTEM BASED ON QTS-ALGORITHM

This section describes the proposed system in detail. First, the implementation of a QTS-algorithm into a trading system will be discussed. Next, we show how QTS generates and optimizes buying and selling strategies. Some examples are given to help understand the proposed system more easily.

A. ENCODING SCHEME

Encoding is a very important step in every heuristic algorithm. It is a key that decides whether or not the performance of system is good. In this trading system, there is a trading pool which is equipped with trading rules. Each rule is composed of technical indicator. We divide trading rules into two types based on different ways of giving trading signal, one is cross type and the other is value type. The cross type means trading rules give signal of buying and selling when short-day line cross long-day line. The value type indicates trading rules buying and selling when its value achieves the threshold. The trading rules in value type would be separate three parts. The first part are the value of indicators, the second part are the operators and the third part are the constants.

TABLE 1. Encoding table of trading rules.

Encode	Indicator
0	MAI(1,20)
1	MAI(10,40)
2	ROC(6)
3	ROC(15)
4	RSI(6)
5	RSI(12)
6	K(9)/D(9)
7	K(15)/D(15)
8	K(15)
9	D(15)

TABLE 2. Example of a trading strategy.

Trading strategy			
Buying	K(15)	D(15)	RSI(6)
Selling	D(15)	ROC(6)	MAI(10,40)
Encoding number of a trading strategy			
Buying	8	9	4
Selling	9	2	1

In this system, there are many technical indicators used in trading rules, such as Stochastic Oscillator (KD), Moving Average Indicator (MAI), Relative Strength Index (RSI) and Rate of Change (ROC). The example of a trading pool (encoding table) is present in Table 1. There might be a lot of rules in an encoding table, but Table 1 shows that if there are only 10 buying and selling rules in a trading pool. In decimal coding, each technical indicator can be represented as $0, 1, 2 \ldots 9$. Example for a trading strategy and its encoding numbers indicate in Table 2.

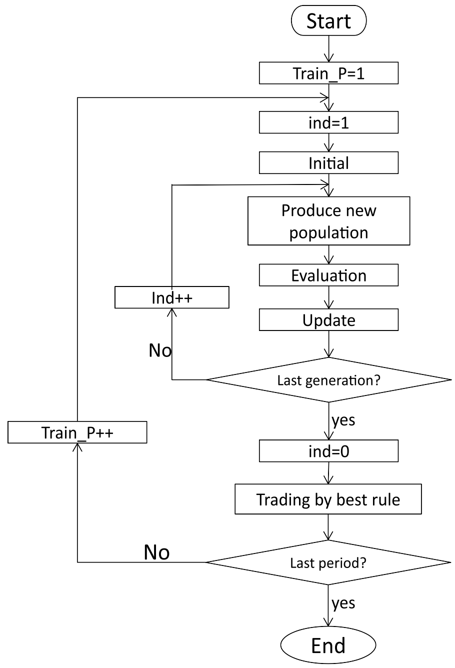

FIGURE 2. **Detailed flowchart of the proposed system based on QTS-algorithm.**

B. RULE-BASED DYNAMIC DECISION-MAKING STOCK TRADING SYSTEM BASED ON QUANTUM-INSPIRED TABU SEARCH ALGORITHM

Fig.2 present the flow chart of the proposed system. Fig. 2 indicates that this system use the concept of sliding window and the core of the proposed trading system is QTS algorithm. Also it shows the detail of QTS algorithm. *Train_P* is a parameter means the number of training periods. In this section, we provide detailed description step by step.

1) INITIALIZE QUANTUM MATRIX

Quantum matrix is used to store the probabilities of measurement, just as each quantum has its probabilities to see different states. First, as encoding scheme described, there are many buying and selling rules put in the trading pool, so we construct two quantum matrices that store the probabilities of measuring buying rules and selling rules. Suppose there are n buying rules and n selling rules put in the trading pool. So, these two quantum matrices $Q_{buy}(0)$ and $Q_{buy}(0)$ store n probabilities respectively and be represented as:

$$Q_{buy} = [q_1, \ q_2, \ \cdots, \ q_n]$$
$$Q_{sell} = [q_1, \ q_2, \ \cdots, \ q_n].$$

The second step is to give an initial value to each matrix. All $q_i, i = 1, 2, .., n$ are initialized with 0.5 and be indicated as :

$$Q_{buy} = [0.5, \ 0.5, \ \cdots, \ 0.5]$$
$$Q_{sell} = [0.5, \ 0.5, \ \cdots, \ 0.5].$$

It means that these qubits will have same probability to collapse into either "0" or "1" state. In other words, when we measure those qubits, there have 50% to see " 0" state and "1" state respectively. "0" state indicates do not choose this rule and on the other side, "1" state signifies choose this rule. Its notice that combining these two quantum matrices would be probabilities of generating a whole trading strategy.

2) PRODUCE NEW POPULATION BY MEASURING QUANTUM MATRIX

This step is inspired by quantum computing. We imitate this measuring action to generate solutions. Each quantum bit can has different states and when we measure it, we only see one of its quantum states. This action can efficiently increase diversification of solution in the initial stage and in the convergence phase it still have probability to see different state to escape local optima.

In this proposed trading system, measuring those two quantum matrices (Q_{buy} and Q_{sell}) p times can produce p trading strategies (solutions). Neighbor solutions N is produced by measuring $Q_{buy}(t-1)$ and $Q_{sell}(t-1)$ multiply with p times and forms a binary matrix which represent as follows:

$$N_{buy} = \begin{bmatrix} x_1^1, & x_2^1, & \cdots, & x_n^1 \\ x_1^2, & x_2^2, & \cdots, & x_n^2 \\ \vdots, & \vdots, & \ddots, & \vdots \\ x_1^p, & x_2^p, & \cdots, & x_n^p \end{bmatrix}$$

$$N_{sell} = \begin{bmatrix} x_1^1, & x_2^1, & \cdots, & x_n^1 \\ x_1^2, & x_2^2, & \cdots, & x_n^2 \\ \vdots, & \vdots, & \ddots, & \vdots \\ x_1^p, & x_2^p, & \cdots, & x_n^p \end{bmatrix}$$

Where x_k^j, $k = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, p$.

Initially, it needs to prepare random numbers r_j with values between 0 and 1. Then, comparing random number r_j with probabilities of each quantum matrix. If r_j greater than q_j, set $x_k^j = 1$, otherwise set $x_k^j = 0$.

TABLE 3. Random number for a buying strategy.

Encode	0	1	2	3	4
r_j	0.35	0.48	0.66	0.21	0.93
Encode	5	6	7	8	9
r_j	0.41	0.55	0.39	0.18	0.26

TABLE 4. Binary string for a buying strategy.

Encode	0	1	2	3	4
x_j	0	0	**1**	0	**1**
Encode	5	6	7	8	9
x_j	0	**1**	0	0	0

For example of a buying strategy. At first, it should prepare 10 random numbers, such as $0.35, 0.48, 0.66, \ldots, 0.18, 0.26$, shown in Table 3.Next, comparing them with each quantum qubit will yield a sequence of binary string as Table 4 showed. Then, according to encoding table will get a buying strategy as Table 5 presented.

TABLE 5. Example for a buying strategy.

Buying Strategy			
Encode	2	4	6
Indicator	ROC(6)	RSI(6)	K(9)/D(9)

3) EVALUATE THE FITNESS FUNCTION

In this paper, we use Taiwan stocks as our sample. We first define the manner of trading in Taiwan. Each buying and selling incurs a fee, which is 0.1425% of the transaction amount. In addition, we also pay the securities transaction tax, which is 0.3%. The equation of how many stocks to buy and how much money return when selling stocks show as equation (5) and (6).

$$\begin{aligned} &\textit{The number of stocks to buy}: \\ &S = \frac{M}{C} \times (1 - (0.1425))\% \\ &\textit{Money return from selling stocks} \\ &\quad = S \times C \times (1 - (0.1425 + 0.3))\% \end{aligned}$$

S is the number of stocks. M is money and C is the close price. The fitness function is the profit gain. It defines as ending money minus initial money.

$$f(k) = M_e - M_i \tag{5}$$

Where M_i is the initial money and M_e is the ending money. Then, the solutions of best and worst are stored into B and W, respectively. In the next step, we will use B and W to update the quantum matrix.

4) UPDATE QUANTUM MATRIX

This step allows a new solution (trading strategy) to become better than the old one. Its principle is to let solution space turn tour the parts of the best solution and remove away from the worst solution. So, the updating action increases the probability of choosing the best solution, and decreases the probability of choosing the worst solution.

TABLE 6. Example of a best buying strategy.

Best Buying Strategy			
Encode	2	4	6
Indicator	ROC(6)	RSI(6)	K(9)/D(9)

TABLE 7. Example of a worst buying strategy.

Worst Buying Strategy			
Encode	1	3	6
Indicator	ROC(15)	MAI(10,40)	K(9)/D(9)

Taking a buying strategy as an example to explain how to update quantum matrix. Here are best and worst buying strategy stored in B and W respectively and shown in Table 6 and 7. The B stores [2, 4, 6] and W stores [1, 3, 6].

TABLE 8. Example for probabilities of a updated buying strategy.

Encode	0	1	2	3	4
q_i	0.5	**0.6**	**0.4**	**0.6**	**0.4**
Encode	5	6	7	8	9
q_i	0.5	**0.5**	0.5	0.5	0.5

Compare the elements $x_i{}^b$ in B and $x_i{}^w$ in W. $x_i{}^b$ is the i-th encoding number in B which is best solution in this iteration. $x_i{}^w$ is the i-th encoding number in W which is worst solution in this iteration. In this example, after updating we will get more probabilities to choose to rule 2 and 4, lower chances to select rule 1 and 3 and the probabilities of choosing rule 6 is unchanged. The example result indicates in Table 8.

TABLE 9. The parameter table of QTS algorithm.

The parameters of QTS algorithm	
Initial Funds	1 million
Trading unit	1 stock
The price of buying/selling	Closing price
Financing	No
Securities lending	No
banking charges /per buying or selling	1.425‰
Transaction tax /per transaction	3‰

5) TERMINATION-CONDITION

The terminal condition in an algorithm can be achieved in many ways. Examples of terminal conditions are when the optimal solution is achieved, or when run-time iteration achieves the setting value. This QTS-algorithm uses the second condition as our terminal condition, and the sliding window stops when runs through all of the periods.

V. EXPERIMENT RESULT

This section shows the results of the experiment using the sample of Taiwan stocks and indices. The proposed trading system was implemented in C++. To confirm the truth and performance of QTS trading system, the data of simulation was from real stock market in yahoo finance. The profit return of the QTS-trading system is compared with other researches and the Buy & Hold approach, which is the benchmark of this financial field. Next, we will discuss the performance of this trading system on two different parameters. First, we examine the size of sliding window which is the length of training and testing period. Second, we will discuss the optimal number of rules. Finally, the experiment results are given.

A. CONDITION OF QTS TRADING SYSTEM

These trading parameters are shown in Table 9. It is supposed that when the proposed trading system simulated trading stocks, the initial fund was $ 1 million and the stock trading unit was unlimited so that one could buy as much stock as possible with the available funds. When either the buying/selling signal is generated, the trading system will buy/sell stocks with the closing price of the day. This trading system does not consider financing and securities lending. The banking charges and transaction tax is according to real situation in Taiwan stock market. Banking charges and transaction tax is set as fitness function used, each buying and selling the bank will charge 1.425% of transaction amount and each transaction (contains one buying and selling) will paid 3% of transaction amount as transaction tax.

In the QTS system, the parameters are shown in Table 10. In each training period the population is set 250 and terminal condition is achieved when 150 iterations are performed then begins training the next period. Every result uses an average of 10 independent simulations.

TABLE 10. The trading parameters of this method.

The trading parameters	
Number of population	250
Number of iterations	150
Range of updated probability	0.008π

TABLE 11. Period of technical indicator set.

Indicator	Period 1	Period 2	Period 3
KD	6	9	15
K	6	9	15
D	6	9	15
* MACD	DIF(5,10)	DIF(5,20)	DIF(10,20)
MVI	(1,20)	(5,20)	(10,20)
MAI	(1,20)	(5,20)	(10,20)
Williams %R	10	15	x
Bias	6	9	15
RSI	6	9	15
PSY	6	10	15
ROC	5	10	15

All trading rules we used in simulation present in Table 11.

B. COMPARING WITH OTHER METHODS

This section we will compare our proposed system with other method in computational intelligence such as fuzzy method, genetic algorithm, neural network and hybrid method.

1) BUILDING TRADING SYSTEM FOR TAIWAN STOCK MARKET USING GENETIC NEURAL NETWORKS [28]

In this study, Genetic Neural Networks, it use 18 kinds of price and volume technical indicators such as relative strength indicators (RSI), price value Indicators (PVI) and moving average volume indicators (MVI), transferred from Taiwan stock price index as the input parameter. Genetic algorithm is optimization tool to construct the trading system based on neural network. It compares with other methods, Genetic Logic Rule (GLR), Single Genetic Logic Rule(SGLR), Buy & Hold, and the RSI technical indicator in normal and abnormal operating. The GLR uses genetic algorithm to optimize trading system, and the trading strategy has tree buying and selling rules. The SGLR only has one buying and selling rule. The RSI in normal operating is buying stock when the value is less than 30 and selling stocks as the value grater than 70. On the contrary, The RSI in abnormal operating is buying stock when the value is grater than 70 and selling stocks as the value less than 30. The method of Buy and Hold buy the stock in the beginning day and sell stock in the end of testing period. In the proposed system, we use two trading rules and the sliding window size is set that training and testing period both are 500 days (about two years) and the number of rules is 2. The experiment is performed with the stock index of TAIEX, and its testing period is from 2000/6/28 to 2004/7/2.

TABLE 12. Comparison result with genetic neural network.

Method	Year profit return
Buy and Hold	-7.2 %
RSI in normal operating	-12.0 %
RSI in abnormal operating	5.4 %
Genetic Logic Rule	2.02 %
Single Genetic Logic Rule	-0.05 %
Genetic Neural Networks	10.27 %
QTS trading system	**15.96 %**

Table 12 shows the returns of GNN, GLR, SGLR, B&H and QTS. The proposed system can earn more profit than the other six approaches. The differences between the QTS system and other trading system are that they use different technical indicators, and the proposed method use more kinds of trading rules. QTS algorithm show its capability to find better combination of trading rules. Also, this trading system is a dynamic system, it uses the sliding window method to prevent the problem of over-fitting which happened in the methods of GNN and GLR.

This shows that choosing the appropriate trading rules as the composition of trading strategy, deciding on the number of rules and prevent the over-fitting problem are very important aspects of a trading system. QTS performs much better than genetic neural network and genetic algorithm in the optimization problem.

2) AN ETF TRADING DECISION SUPPORT SYSTEM BY USING NEURAL NETWORK AND TECHNICAL INDICATORS [21]

This research proposes a decision support system based on a Neural Network and the Taiwan 50 Index Exchange Traded Funds (ETF) is used as samples. This method is composed of three main models. The first is a training model, which decides when to buy or sell for Taiwan 50 Index ETF. It uses six kinds of technical indicator such as moving average, BIAS, RSI, William index, KD and Directional Movement Index (DMI) to find transaction time. The second is a training model for predicting price and fluctuation. The third model constructs buying and selling strategies for the Taiwan 50 Index ETF. In the proposed system, we use three trading rules and the sliding window size is set that training period 375 days (about one and half year) and testing period is 102 days (about five months) and the number of rule is 2. The testing period is from 2005/1/3 to 2005/10/31.

TABLE 13. Comparison result in simulation 2.

Approach	Year profit return
TAIEX	-7.42%
Taiwan 50 Index	-7.41%
Buy and Hold	-7.76%
Neural Network	9.95 %
QTS trading system	**13.26%**

Table 13 shows the experiment result. The Neural Network and QTS methods both outperform other indices and B&H method. Though those indices have negative returns, the neural network and QTS systems still have positive returns. In this experiment, the QTS system makes more profit than the neural network system does. This indicates that QTS-algorithm has the ability to maximize profit over a very short period. Table 14 and 15 show the investment from neural network and the proposed system, from these two tables we can see QTS trading system can buy in the lower price and sell in higher price. Also QTS trading system doesn't trading frequently so that we can avoid unnecessary charges and tax.

TABLE 14. Investment by neural network trading strategy.

Time	Closing price	Transaction type
2005/01/14	44.24	Buying
2005/01/31	45.24	Selling
2005/03/18	45.01	Buying
2005/06/13	47.00	Selling
2005/07/05	47.03	Buying
2005/08/19	47.57	Selling

TABLE 15. Investment by QTS trading system.

Time	Closing price	Transaction type
2005/01/24	45.39	Buying
2005/08/03	49.75	Selling
2005/10/24	44.57	Buying
2005/10/31	45.44	Selling

3) FUZZY RULED-BASED STOCK TRADING SYSTEM [18]

This fuzzy rule-based systems (FRBS) method are decision-making system combining the non-linear model of fuzzy rules. It uses two rules: a buy-rule and sell-rule which are composed of Moving Average Indicator and the Moving Average Volume Indicator. When these technical indicators trigger the fuzzy buy-rule, a buying decision is made. When these technical indicators trigger the fuzzy sell-rule, a selling decision is proposed. We test in the period, 2001/01/01 to 2005/12/31. The QTS use the parameters, training and testing both 750 dyas, and the number of rules is 3. The comparison criterion is the annual profit return.

TABLE 16. Comparison result with fuzzy system.

Method	Annual profit return
Symmetry Rules Approach(1,1)	8.97 %
Asymmetry Rules Approach(0.8,1)	6.47 %
Buy and Hold	5.67 %
QTS trading system	**12.27 %**

Table 16 indicates the profit gain of the Fuzzy system, Buy & Hold method, and the QTS system. The proposed trading system shows a much better result than the other methods. There are two differences between the Fuzzy system and the QTS system. First, the Fuzzy system uses two technical indicators, while the QTS system uses ten indicators for technical analysis. Second, the QTS system uses the sliding window method to avoid the problem of over-fitting, while the Fuzzy system does not. This confirms that using the appropriate number of indicators may yield a better combination of rules, and that sliding window can effectively avoid the problem of over-fitting.

C. DISCUSSION OF COMPARISON RESULT

The above three simulation results are in different stocks and testing periods. Those methods use different computer intelligences apply to stock trading to determine the timing for buying or selling stock in the Taiwan stock market. Methods comparison with the proposed system include the fuzzy system, genetic algorithm, neural networks and hybrid method of genetic algorithm and neural networks. The comparison results in recent years show that the proposed method has better performance in the Taiwan stock market than the other methods do, and always outperforms the Buy & Hold approach. Since the proposed system use more and useful technical indicators, and QTS algorithm show its great ability to fine optimal combination of trading rules. Also we use sliding window to dynamically construct trading strategies to prevent the problem of over-fitting. This confirms that our system is suited to many situations in the Taiwan stock market and can make profit over many time periods. In conclusion, the comparison results represent that our system is flexible, profitable and minimizes the problem of over-fitting.

D. THE PROPOSED SYSTEM WITH DIFFERENT PARAMETERS

The purpose of the proposed trading system is to make more profit with lower risk. In the course of the study, we found that there were two key factors affecting the performance of this trading system. One is determining the size of sliding window, how long training and testing periods are, and the other is deciding how many trading rules to use.

First, too long or too short periods of training and testing are not good, because too long a training period may cause the problem of over-fitting, and too short a training period may not provide enough time to train a successful trading strategy. Second, too many or too few rules will adversely affect the performance of this system. Since using too many rules to analyze the market may confuse the buying and selling signals, and will make it difficult to achieve the threshold of buying and selling. While using too few rules will not analyzing the stock market effectively and trading too frequently than costs many fee and tax. Based on the above description, this section discusses these two aspects in detail.

In the following simulations, the Taiwan Stock Market Index (TAIEX) and Taiwan 50 Index ETF are chosen as samples. TAIEX is a stock price index made from the Taiwan Stock Exchange. The feature of TAIEX is that greater equity shares have more impact to the index than small capital stocks. Taiwan 50 ETF is an exchange traded fund and established form Yuanta Securities Investment Trust Co., Ltd. It consists of the top 50 market value companies and also is a high level of transparency index. Most people in Taiwan are familiar with these stock indices, and it is deemed to accurately reflect the direction of economic trends of Taiwan. So, using these two indices can provide a reasonably accurate representation of the movement of the economy and stock market of Taiwan. The experiments run from 2010/11/1 to 2013/10/31, total three years. Since this testing period is recent and long enough, it has the value to test. Fig. 3 and 4 present the closing price trend of TAIEX and Taiwan 50 index ETF. The B&H method has 0.11% profit return in TAIEX and 1.34% profit return in Taiwan 50 index ETF. Each result uses an average of 10 independent runs.

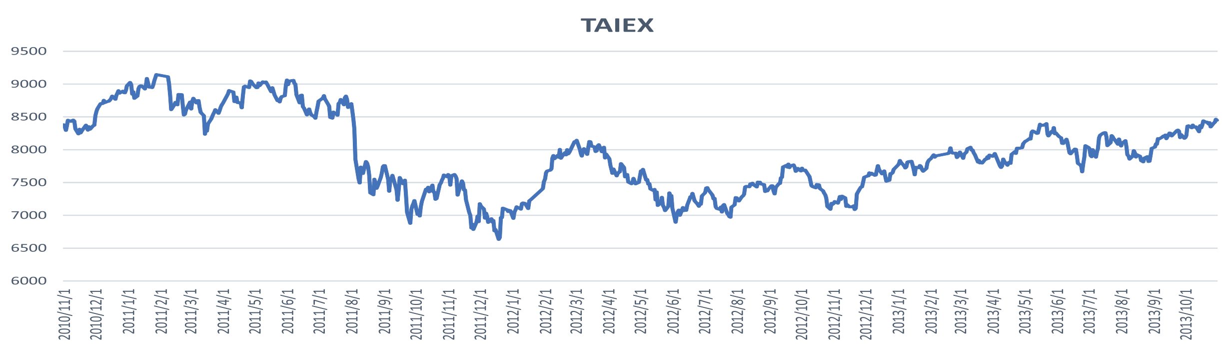

FIGURE 3. The trend of TAIEX from 2010/11/1 to 2013/10/31.

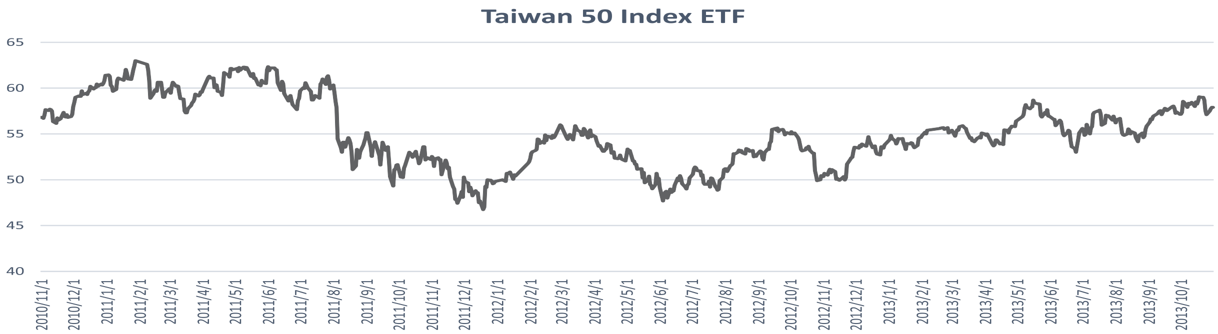

FIGURE 4. The trend of Taiwan 50 index ETF from 2010/11/1 to 2013/10/31.

TABLE 17. Profit return form 750 and 500 training days in TAIEX and taiwan 50 ETF.

Training period(day)	750					500			
Testing period(day)	750	500	250	125	62	500	250	125	62
Profit return in TAIEX(%)	10.51%	3.20%	7.72%	21.67%	6.97%	12.40%	13.55%	22.01%	6.97%
Profit return in Taiwan 50 index ETF (%)	0%	6.64%	10.75%	29.37%	28.42%	19.28%	19.85%	22.52%	18.15%
Variance in TAIEX (%)	0.75 %	0.24%	0.53%	0.65%	0.06%	0.62%	0.60%	0.37%	1.78%
Variance in Taiwan 50 index ETF (%)	0 %	0.06%	0.72%	0.92%	1.36%	0.37%	0.38%	0.45%	0.21%

TABLE 18. Profit return form 250,125 and 62 training days in TAIEX and taiwan 50 ETF.

Training period(day)	250			125		62
Testing period(day)	250	125	62	125	62	62
Profit return in TAIEX (%)	8.89%	12.67%	18.49%	5.62%	5.94%	5.93
Profit return in Taiwan 50 index ETF (%)	11.49%	10.68%	12.07%	6.82%	2.68%	0%
Variance in TAIEX (%)	0.30%	0.52%	0.70 %	0.41 %	0.50%	0.28 %
Variance in Taiwan 50 index ETF (%)	0.53%	0.53%	0.06%	0.35 %	0.11 %	0.00% %

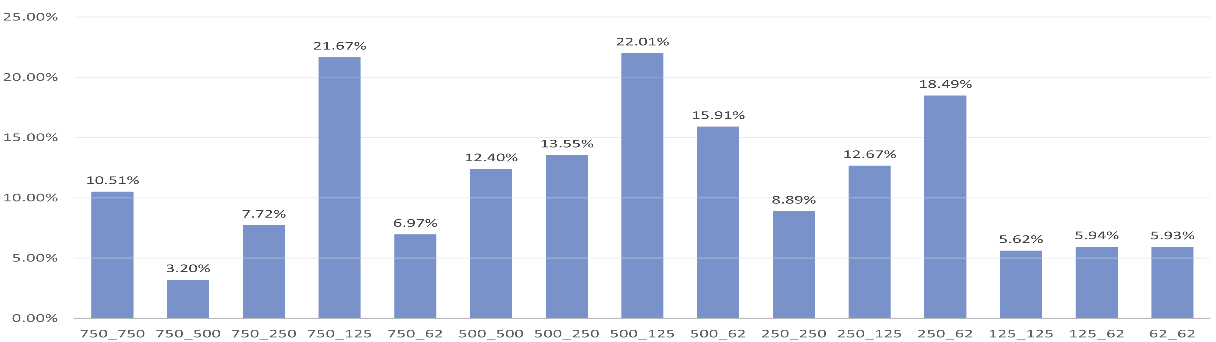

FIGURE 5. The histogram of average profit return of every training and testing period in TAIEX.

1) TRAINING AND TESTING PHASE

The reason for this experiment is that we want to test how long a time would be good as a training and testing period. This system is a long-term trading system, since a long-term trading system can reduce unnecessary fees. This system uses periods ranging from 750 days to 62 days (about three years to a season). We use the profit return and variance as the criterion for evaluating how long training or testing periods should be.

The experiment results in TAIEX and Taiwan 50 index ETF are shown in Table 17 and 18. The first row indicates the training periods, the second row shows the testing period relative to its training period, and the third row shows the profit return, which is an index of trading system performance. The x-axis of Fig. 5 and 6 means permutations of training period and test period (training days _ testing days) and the y-axis denotes the average profit return form 20101/11 to 2013/12. The results signify the proposed system outperforms buy and hold method. In detailed analysis, it can find in fig. 5 and 6 show training and testing periods that are too long or too short have worse performance than other combinations

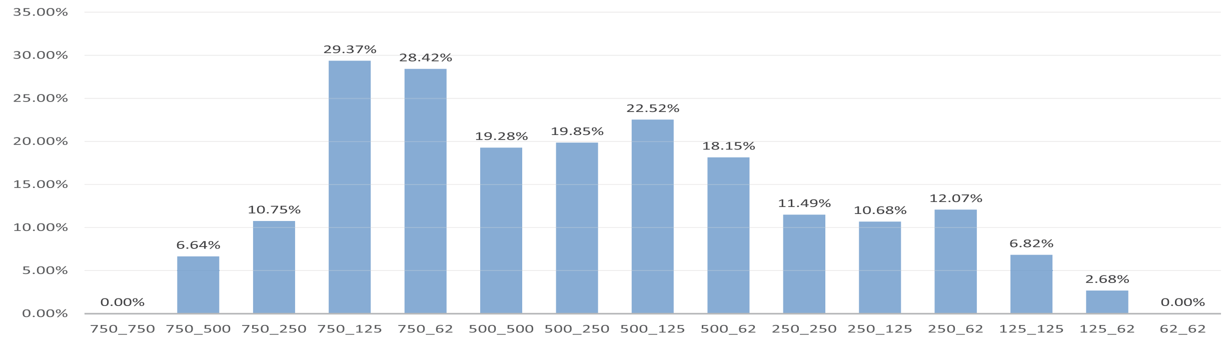

FIGURE 6. The histogram of average profit return of every training and testing period in Taiwan 50 index ETF.

TABLE 19. Average of profit return from 1 to 10 rules in TAIEX and taiwan 50 index ETF.

The number of rule	1	2	3	4	5	6	7	8	9	10
TAIEX	2.50%	-2.24%	17.65%	22.01%	21.09%	13.40%	12.42%	12.35%	7.73%	4.03%
Taiwan 50 index ETF	-3.75%	10.06%	16.01%	13.97%	22.52%	19.21%	11.20%	10.05%	1.53%	0.00%
Variance in TAIEX	0%	0.08%	0.90%	0.97%	1.08%	0.41%	0.45%	0.63%	0.47%	0.24%
Variance in Taiwan 50 index ETF	0%	0.82%	0.52%	0.97%	0.45%	0.27%	0.36%	0.33%	0.21%	0%

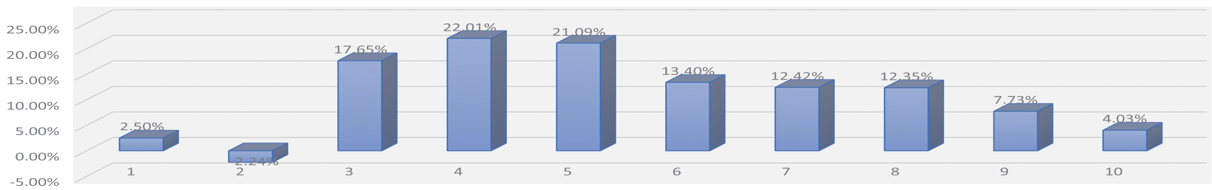

FIGURE 7. The histogram of average profit return from 1 to 10 rules in TAIEX.

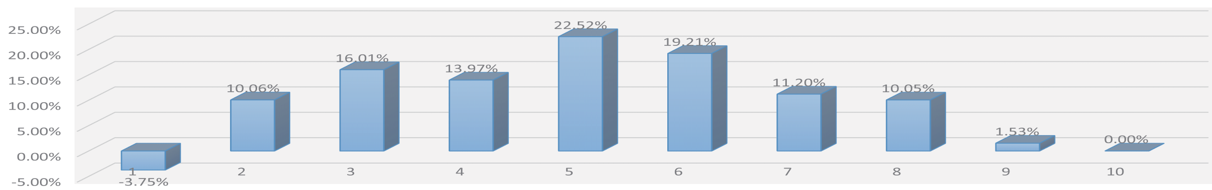

FIGURE 8. The histogram of average profit return from 1 to 10 rules in Taiwan 50 index ETF.

of training and testing such as 750_750 and 62_62. In the other side, the results also indicate that a nice long training period with short testing period have better performance. For example, 750_125, 500_125 and 250_62 in TAIEX and Taiwan 50 index ETF both have much better performance than other combinations. It may need such long period to train better combination of trading rules and slide with short period to prevent over-fitting.

2) NUMBER OF TRADING RULES

This section focuses on how many rules are required for excellent performance in this trading system. Using too few trading rules will generate many buying and selling signals will cause problems such as trading too frequently to cause much more charges, and the system may not accurately analyze the status of the stock market. Although, using more technical indicators may bring lower risk, using too many technical indicators still causes confusion in technical analysis. Some stock market analysts say that "Too much analysis equals no analysis." So, the purpose of this experiment is to determine the optimal number of rules. This experiment uses a training period of 500 days, and 125 day testing period, determined to be the good performance periods in the previous experiment. It chooses TAIEX and the Taiwan 50 index ETF as samples. Table 19 shows the result and the fig 7 and 8 help us observe the trend.

The histogram 7 and 8 prove that when the number of rules is too few or too many has poor performance, also in one or two rules appear negative profit return and in ten rules there no any transaction happened. Since that too few rules will result in this system incur large costs in transaction fees

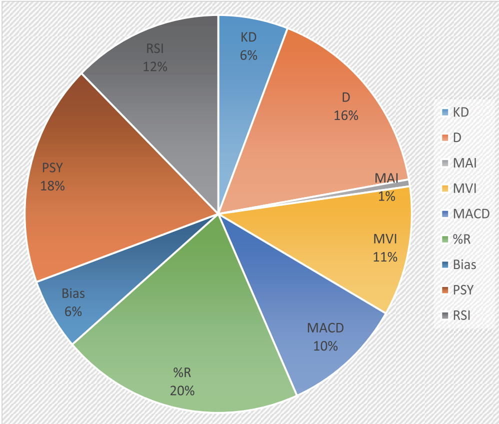

FIGURE 9. **The percentage pie chart of used indicators in 3 trading rules for TAIEX from 2010 to 2013.**

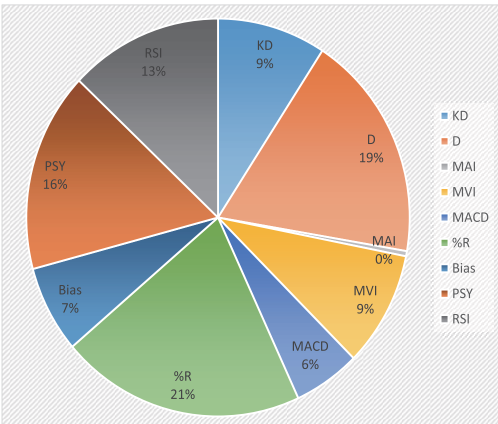

FIGURE 10. **The percentage pie chart of used indicators in 4 trading rules for TAIEX from 2010 to 2013.**

and too many rules has too many restrictions to achieve the threshold. So the number of rules between one and two, nine and ten will not suit for a trading strategy. As the experiment results show, we can find the number of rules between 3 and 6 have better performance than other numbers and outperform the B&H method. The performance of six to eight rules are also not bad and still earn better profit than B&H method. Regardless of the number of rules, QTS trading system still can earn profit in a certain level.

Instead of provide the profit return, we do more analysis to this simulation. We choose the simulation in top three profit to do the analysis of the percentages of used technical indicators. The fig 9, 10 and 11 show the ratio of used indicators from three, four, and five rules, for TAIEX. Fig 12, 13 and 14 show the ratio of used indicators from three, five, and six rules, for 0050. The ratio of used technical indicators in fig. 9 to 14 have no significant difference. It means QTS can find optimal solution from different size of sliding window and number of rules.

From the two above experiments, we determine that the best training and testing periods are 500 and 125 days, 4-rule,

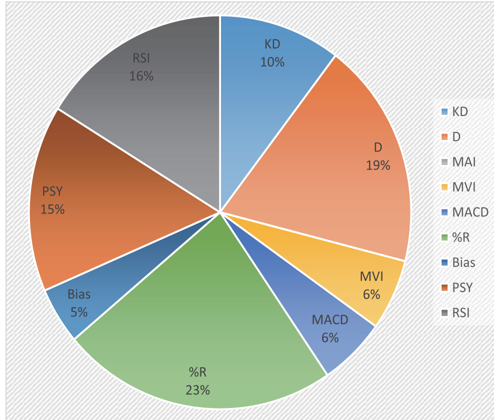

FIGURE 11. **The percentage pie chart of used indicators in 5 trading rules for TAIEX from 2010 to 2013.**

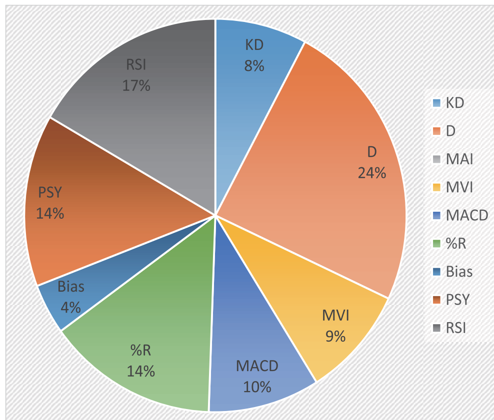

FIGURE 12. **The percentage pie chart of used indicators in 3 trading rules for Taiwan50 index EFT from 2010 to 2013.**

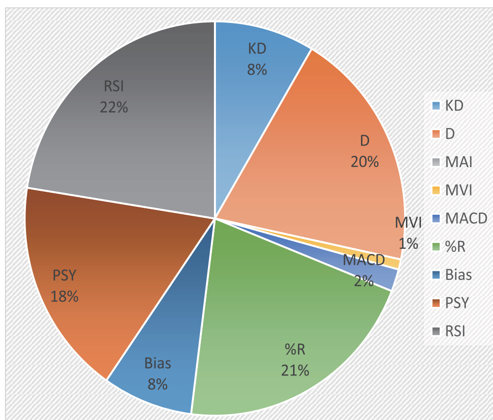

FIGURE 13. **The percentage pie chart of used indicators in 5 trading rules for Taiwan50 index EFT from 2010 to 2013.**

in TAIEX and 750_125, 5-rule in Taiwan 50 index ETF. The profit return of the strategy made by QTS in TAIEX is 28.72% and in Taiwan 50 Index ETF is 36.78%. Fig. 15 and 16 show the buying and selling points in TAIEX and Taiwan 50

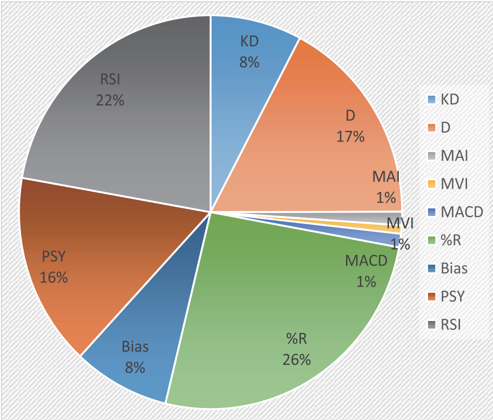

FIGURE 14. The percentage pie chart of used indicators in 6 trading rules for Taiwan50 index EFT from 2010 to 2013.

index ETF, respectively. Blue line means the trend of closing price and the orange line denotes the period from buying to selling.

E. DISCUSSION & FUTURE WORK

Above two experiments, the trading days contain form 2010 to 2013 that are about 3 years. In this 3 years, the stock market of Taiwan have been some turbulent periods. Taiwan stock market is affected by the Asia and American economic. From 2011, the 311 earthquake happened in Japan, the flood hit Thailand, European debt crisis spread and Double-dip Recession came to United States. Taiwans industry have been great affected and leads a period of fallen in Taiwan stock market also in global stock market. In the 2012, Taiwan and U.S. president election create a period rising but after it there have a sudden falling. Since 2013, the housing market have been good improved to support the stock market of United States to go upward, the European debt crisis has been under control and so on. The various signs let the stock market have been improved.

However, there were still many uncertain factors affected the stock market of Taiwan. From 2010 to 2013, the stock market of Taiwan had been gone through ups and downs. Even though the stock market of Taiwan is so volatile the proposed trading system shows stable and profit. The experiment indicates potential results from 2010 to 2013. The steady of the proposed trading system is based on the Quantum-Inspired Tabu Search (QTS) Algorithm, technical analysis and sliding window. We use many kinds of technical indicators to cope with various situation in stock market, also QTS-algorithm present its great abilities in effectiveness and efficiency, it can optimal or near optimal combination of trading rule in a short time and sliding window can minimize the problem of over-fitting. No matter there is upward or downward trend, most of time QTS can find great trading rules for Taiwan stock market.

For future studies, first, our system can use other or more technical indicators to find better timing for buying and selling. Second, we can use the method of financing and security lending to deal with more falling and rising period. In addition, the above experiment results show that while this trading system is profitable in the Taiwan stock market, other markets need to be tested to show that it can maximize profit in other countries and other stocks.

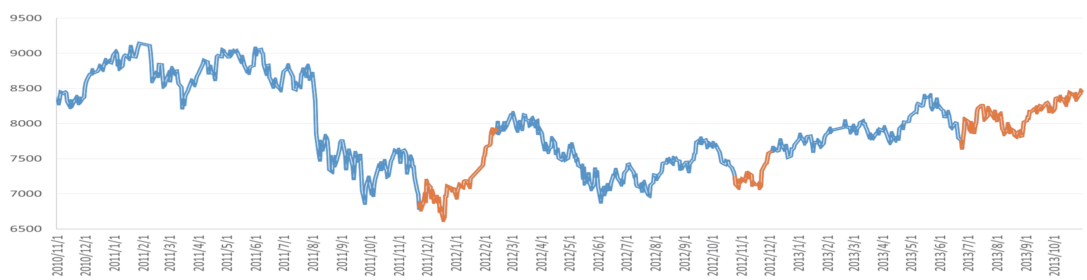

FIGURE 15. The trend from buying to selling based the strategy which QTS algorithm find in TAIEX from 2010/11/1 to 2013/10/31 and the profit return is 28.72%.

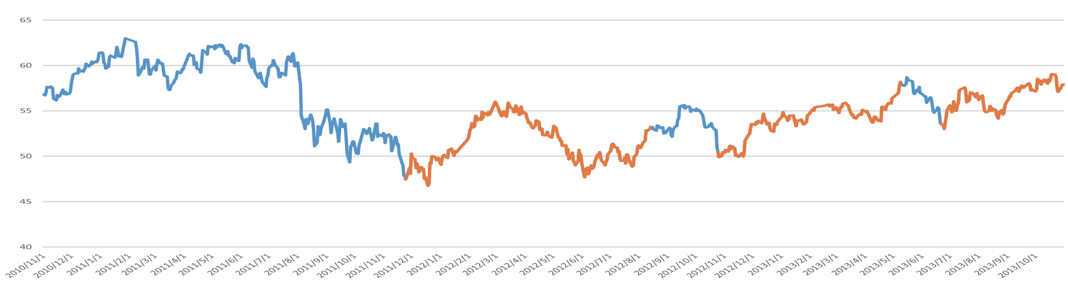

FIGURE 16. The trend from buying to selling based the strategy which QTS algorithm find in Taiwan 50 Index ETF from 2010/11/1 to 2013/10/31 and the profit return is 36.78%.

VI. CONCLUSION

This paper proposes a dynamic stock trading system based on a QTS-algorithm. There are three key points in this paper. First, the trading strategy made by this system is based on technical analysis. This system uses technical indicators such as moving average indicators (MAI), stochastic indicator (KD), relative strength index (RSI) and rate of change (ROC) to find the signals for buying and selling. Second, it uses a Quantum-inspired Tabu Search (QTS) algorithm to find the optimal combinations of rules, which are composed of technical indicators. QTS performs better than other heuristic algorithms, both in terms of time and the ability to find optimal solutions. Finally, it uses sliding-window to achieve the goal of a dynamic system, and to avoid the problem of overfitting. Through experiment results, the return of our system shows the potential result. Compared with other approaches and the Buy & Hold method, the proposed system performs much better in the Taiwan stock market.

REFERENCES

[1] Y.-H. Chou, Y.-J. Yang, and C.-H. Chiu, "Classical and quantum-inspired tabu search for solving 0/1 knapsack problem," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2011, pp. 1364–1369.

[2] P.-C. Chang, C.-Y. Fan, and C.-H. Liu, "Integrating a piecewise linear representation method and a neural network model for stock trading points prediction," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 39, no. 1, pp. 80–92, Jan. 2009.

[3] K. K. Ang and C. Quek, "Stock trading using RSPOP: A novel rough set-based neuro-fuzzy approach," *IEEE Trans. Neural Netw.*, vol. 17, no. 5, pp. 1301–1315, Sep. 2006.

[4] H. Huang, M. Pasquier, and C. Quek, "Financial market trading system with a hierarchical coevolutionary fuzzy predictive model," *IEEE Trans. Evol. Comput.*, vol. 13, no. 1, pp. 56–70, Feb. 2009.

[5] J.-H. Wang and J.-Y. Leu, "Dynamic trading decision support system using rule selector based on genetic algorithms," in *Proc. IEEE Signal Process. Soc. Workshop, Neural Netw. Signal Process.*, Sep. 1996, pp. 119–128.

[6] J.-H. Wang, S.-M. Chen, and J.-Y. Leu, "Stock trading decision support system using a rule selector based on sliding window," in *Proc. IEEE Int. Conf. Syst., Man, Cybern., Comput. Cybern. Simul.*, vol. 1. Oct. 1997, pp. 559–564.

[7] P. Parracho, R. Neves, and N. Horta, "Trading with optimized uptrend and downtrend pattern templates using a genetic algorithm kernel," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2011, pp. 1895–1901.

[8] F. Allen and R. Karjalainen, "Using genetic algorithms to find technical trading rules," *J. Financial Econ.*, vol. 51, no. 2, pp. 245–271, Feb. 1999.

[9] L. A. Becker and M. Seshadri, "GP-evolved technical trading rules can outperform buy and hold," in *Proc. 6th Int. Conf. Comput. Intell. Natural Comput.*, Sep. 2003, pp. 26–30.

[10] J.-Y. Potvin, P. Soriano, and M. Vallee, "Generating trading rules on the stock markets with genetic programming," *Comput. Oper. Res.*, vol. 31, no. 7, pp. 1033–1047, Jun. 2004.

[11] D. Lohpetch and D. Corne, "Discovering effective technical trading rules with genetic programming: Towards robustly outperforming buy-and-hold," in *Proc. World Congr. Nature Biol. Inspired Comput.*, Dec. 2009, pp. 439–444.

[12] D.-Y. Chiu, C.-Y. Shiu, and Y.-S. Lin, "USA S&P 500 stock market dynamism exploration with moving window and artificial intelligence approach," in *Proc. 7th Int. Conf. Netw. Comput. Adv. Inf. Manage. (NCM)*, Jun. 2011, pp. 341–345.

[13] Y. Chen, S. Mabu, K. Hirasawa, and J. Hu, "Genetic network programming with sarsa learning and its application to creating stock trading rules," in *Proc. IEEE Congr. Evol. Comput.*, Sep. 2007, pp. 220–227.

[14] N. M. Kwok, G. Fang, and Q. P. Ha, "Moving average-based stock trading rules from particle swarm optimization," in *Proc. Int. Conf. Artif. Intell. Comput. Intell.*, vol. 1. Nov. 2009, pp. 149–153.

[15] A. C. Briza and P. C. Naval, Jr., "Design of stock trading system for historical market data using multiobjective particle swarm optimization of technical indicators," in *Proc. 10th Annu. Conf. Companion Genet. Evol. Comput. (GECCO)*, 2008, pp. 1871–1878.

[16] N. M. Kwok, G. Fang, and Q. P. Ha, "Moving average-based stock trading rules from particle swarm optimization," in *Proc. Int. Conf. Artif. Intell. Comput. Intell.*, vol. 1. Nov. 2009, pp. 149–153.

[17] F. Wang, P. L. H. Yu, and D. W. Cheung, "Complex stock trading strategy based on particle swarm optimization," in *Proc. IEEE Conf. Comput. Intell. Financial Eng. Econ. (CIFEr)*, Mar. 2012, pp. 1–6.

[18] I.-C. Yeh and C.-H. Lien, "Fuzzy rule-based stock trading system," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ)*, Jun. 2011, pp. 2066–2072.

[19] Y.-L. Huang, S.-J. Horng, T.-W. Kao, I.-H. Kuo, and T. Takao, "A hybrid forecasting model based on adaptive fuzzy time series and particle swarm optimization," in *Proc. Int. Symp. Biometrics Secur. Technol. (ISBAST)*, Mar. 2012, pp. 66–70.

[20] I.-H. Kuo, S.-J. Horng, T.-W. Kao, T.-L. Lin, C.-L. Lee, and Y. Pan, "An improved method for forecasting enrollments based on fuzzy time series and particle swarm optimization," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 6108–6117, Apr. 2009.

[21] M.-H. Kuo and C.-L. Chen, "An ETF trading decision support system by using neural network and technical indicators," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2006, pp. 2394–2401.

[22] M.-C. Chen, E. Y.-H. Lin, and R.-J. Kuo, "The application of ant colony optimization system on the investment strategies at Taiwan stock market," *J. Inf. Manage.*, vol. 15, no. 1, pp. 153–178, 2008.

[23] Y. Chen, S. Mabu, E. Ohkawa, and K. Hirasawa, "Constructing portfolio investment strategy based on time adapting genetic network programming," in *Proc. IEEE Congr. Evol. Comput.*, May 2009, pp. 2379–2386.

[24] W.-H. Wang, C.-H. Chiu, S.-Y. Kuo, S.-F. Huang, and Y.-H. Chou, "Quantum-inspired tabu search algorithm for reversible logic circuit synthesis," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2012, pp. 709–714.

[25] S.-Y. Kuo, Y.-J. Yang, F.-J. Lin, I.-I. Liu, and Y.-H. Chou, "Improved quantum-inspired tabu search algorithm for solving function optimization problem," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2013, pp. 823–828.

[26] S.-Y. Kuo, C. Kuo, and Y.-H. Chou, "Dynamic stock trading system based on quantum-inspired tabu search algorithm," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2013, pp. 1029–1036.

[27] S.-Y. Kuo, C. Kuo, Y.-C. Tsai, C.-Y. Lin, C.-C. Chen, and Y.-H. Chou, "Intelligent stock trading system based on QTS algorithm in Japan's stock market," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2013, pp. 977–982.

[28] L.-C. Lien, I.-C. Yeh, C.-T. Yuan, and F.-M. Yang, "Building trading system for Taiwan stock market using genetic neural networks," *J. Inf. Manage.*, vol. 15, no. 1, pp. 29–51, 2008.

[29] Y.-H. Chou, C.-Y. Chen, C.-H. Chiu, and H.-C. Chao, "Classical and quantum-inspired electromagnetism-like mechanism and its applications," *IET Control Theory Appl.*, vol. 6, no. 10, pp. 1424–1433, Jul. 2012.

[30] C.-Y. Chen, Y.-H. Chou, and H.-C. Chao, "Distributed quantum entanglement sharing model for high-performance real-time system," *Soft Comput.*, vol. 16, no. 3, pp. 427–435, Mar. 2012.

[31] M.-Y. Su, S.-C. Yeh, and K.-C. Chang, "Using incremental mining approach to analyze network traffic online based on fuzzy rules," *J. Internet Technol.*, vol. 9, no. 1, pp. 77–86, 2008.

[32] W.-Z. Guo, N.-X. Xiong, C. Lee, L. T. Yang, G.-L. Chen, and Q. Weng, "Task allocation algorithm based on particle swarm optimization in heterogeneous computing environments," *J. Internet Technol.*, vol. 11, no. 3, pp. 343–351, May 2010.

YAO-HSIN CHOU received the Ph.D. degree from the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, in 2009. He is currently an Associate Professor with the Department of Computer Science and Information Engineering, National Chi-Nan University, Nantou, Taiwan. His current research interests include quantum information science and electronic design automation.

SHU-YU KUO received the B.S. and M.S. degrees in computer science and information engineering from National Chi-Nan University, Nantou, Taiwan, in 2012 and 2013, respectively, where she is currently pursuing the Ph.D. degree with the Department of Computer Science and Information Engineering. Her current research interests are quantum-inspired evolutionary computation and automated trading systems.

CHI-YUAN CHEN (M'10) received the Ph.D. degree in electrical engineering from National Dong Hwa University, Hualien, Taiwan, in 2014. He has been an Assistant Professor with the Department of Computer Science and Information Engineering, National Ilan University, Yilan, Taiwan, since 2014, and the Associate Editor-in-Chief of the *Journal of Internet Technology* since 2014. His research interests include mobile communication, network security, and quantum computing. He is a member of the Association for Computing Machinery.

HAN-CHIEH CHAO (SM'04) is currently a joint appointed Full Professor with the Department of Computer Science and Information Engineering, National Ilan University (NIU), Yilan, Taiwan. He has served as the President of NIU since 2010. He was the Director of the Computer Center with the Ministry of Education, Taipei, Taiwan, from 2008 to 2010. His research interests include high-speed networks, wireless networks, IPv6-based networks, digital creative arts, e-Government, and the digital divide. He received the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1989 and 1993, respectively. He has authored or co-authored four books, and has published about 400 refereed professional research papers. He has mentored over 100 M.S.E.E. thesis students and four Ph.D. students. He has been frequently invited to give talks at national and international conferences and research organizations. He is the Editor-in-Chief of *IET Networks*, *Journal of Internet Technology*, *International Journal of Internet Protocol Technology*, and *International Journal of Ad Hoc and Ubiquitous Computing*. He has served as a Guest Editor of *Mobile Networking and Applications* (ACM), the IEEE Journal on Selected Areas in Communications, the *IEEE Communications Magazine*, the IEEE Systems Journal, *Computer Communications*, *IEE Proceedings Communications*, *The Computer Journal*, *Telecommunication Systems*, *Wireless Personal Communications*, and *Wireless Communications and Mobile Computing*. He is a fellow of the Institution of Engineering and Technology.