RISK-ADJUSTED OPTIMAL TECHNICAL TRADING RULES ON SET50 INDEX WITH GENETIC PROGRAMMING

THEERAPHAT VORRAPITTAYAPORN

MASTER OF SCIENCE PROGRAM IN FINANCE
(INTERNATIONAL PROGRAM)
FACULTY OF COMMERCE AND ACCOUNTANCY
THAMMASAT UNIVERSITY, BANGKOK, THAILAND
MAY 2009

RISK-ADJUSTED OPTIMAL TECHNICAL TRADING RULES ON SET50 INDEX WITH GENETIC PROGRAMMING

THEERAPHAT VORRAPITTAYAPORN

MASTER OF SCIENCE PROGRAM IN FINANCE
(INTERNATIONAL PROGRAM)
FACULTY OF COMMERCE AND ACCOUNTANCY
THAMMASAT UNIVERSITY, BANGKOK, THAILAND
MAY 2009

Risk-adjusted optimal technical trading rules on SET50 index with genetic programming

Theeraphat Vorrapittayaporn

An Independent Study
Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science (Finance)

Master of Science Program in Finance
(International Program)
Faculty of Commerce and Accountancy
Thammasat University, Bangkok, Thailand
May 2009

Thammasat University
Faculty of Commerce and Accountancy
An Independent Study

By

Theeraphat Vorrapittayaporn

"Risk-adjusted optimal technical trading rules on SET50 index with genetic programming"

has been approved as a partial fulfillment of the requirements
for the Degree of Master of Science (Finance)

On May, 2009

Advisor: ……………………………………
(Asst. Prof. Dr. Sittisak Leelahanon)

Co-Advisor: ……………………………………
(Dr. Vatcharapol Sukhotu)

CONTENTS

Abstract……………………………………………………………………………..1

1. Introduction and Literature Review………………………………………………...2

2. Theoretical Framework…………………………………………………………..7

3. Data and Methodology……………….…………………………….....................12

4. Empirical Result………………………………………………………………...21

5. Conclusion………………………………………………………………………24

LIST OF TABLES

Table I : Summary of Results………………………………………………………… 27

LIST OF FIGURES

Figure 1 : Example of Population as Decision Tree……………………………….. 28

Figure 2 : Crossover Operation……………………………………………………..29

Figure 3 : Mutation Operation……………………………………………………... 30

Figure 4 : Processes of Genetic Programming……………………………………...30

Figure 5 : Processes of Constructing Decision Tree………………………………..31

Figure 6 : The First Optimal Trading Rule………………………………………… 31

Figure 7 : Position of The First Optimal Trading Rule……………………………..32

Figure 8 : The Second Optimal Trading Rule………………………………………32

Figure 9 : The Third Optimal Trading Rule………………………………………...33

Figure 10 : The Forth Optimal Trading Rule……………………………………….33

Figure 11 : The Fifth Optimal Trading Rule………………………………………..34

Risk-adjusted optimal technical trading rules on SET50 index with genetic programming

ABSTRACT

This paper applies genetic programming to optimize technical trading rules with SET50 index from June 1998 to May 2008 including 0.25% one-way transaction costs. We follow Neely (2003) who extends Allen and Karjalainen (1999) by taking into account the risk-adjusted technique, for example, Sharpe ratio as fitness criteria. In testing period, the result shows that the Sharpe ratio from the best rule is equal to 0.18 which is higher than Sharpe ratio from buy-and-hold strategy, 0.014. Nevertheless, we can't claim that the market is inefficient because the rule may not make superior result in other periods. The average Sharpe ratio, average r_b-r_s, average trades per year, and average %Long in the market are equal to 0.007, -0.002, 0.08, and 31.50% respectively.

I. INTRODUCTION AND LITERATURE REVIEW

Trading rules based on technical analysis is well-known among investors. This method is originated from Dow Theory by Charles Dow, an American journalist and founder of The Wall Street Journal. It is claimed the ability to forecast the future securities price movement by examining past price and volume. It cans also indicate market condition such as uptrend, downtrend, and sideway. Easy to understand is one of its advantages compares to fundamental analysis which requires several fields such as economics, and accounting. Moreover, it can apply with different kinds of product such as stock, foreign exchange, commodity and futures. Various researches study the beneficial of technical analysis. Brock, Lakonishok, and Lebaron, (1992) test two simple trading rules, moving average and trading range break, with Dow Jones Industrial Average Index from 1897 to 1986. The result suggests that the rules can produce positive excess returns over 4 strategies, random walk, AR(1), GARCH-M and Exponential GARCH without transaction cost. Although, ex post selection problem cans happen from such researches. Because of their methodology, they may select the appropriate trading rules after they have seen the historical securities price and yield biased positive excess returns. This problem called data snooping problem (Allen, and Karjalainen, 1999). Moreover, the rules broadly used by technical trader are considered to be one type of data snooping problem as well. They are broadly used since they can make profits with the previous data (Ready, 1998).

Genetic programming (GP), created by John Koza, is recently introduced as a new method to optimize trading rules and mitigate data snooping problem. This method can avoid data snooping problem because the rules are chosen by genetic programming before testing periods. GP is evolutionary algorithm motivated by biological evolution. The sample techniques are crossover, mutation, reproduction, and selection. There are many researches studying trading rules with GP in the last decade. One of famous research papers studies S&P500 index from 1928 to 1995 using daily price data (Allen and Karjalainen, 1999). They investigate whether technical trading rules from genetic programming can make consistent excess returns compared to simple buy-and-hold strategy. The rules are combination of maximum, minimum, and average of prices during a specific time window and other real-valued functions. T-test is used as a hypothesis test. The result indicates that the rules don't earn consistent excess returns after transaction costs and exclusion of dividends. However, the rules have ability to predict daily returns. The rules often hold the index in low-volatility periods with positive daily returns and the rules are out of the market in high-volatility periods with negative daily returns. This is amazing because the rules don't take volatility into consideration. Their explanation is that there is low-order serial correlation in stock index returns. Their excess returns measurement is argued that it is measured in terms of raw excess returns rather than risk-adjusted excess returns which may lead to unclear interpretation for Efficient Market Hypothesis (EMH) since there are several times that the rules are out of the market (Neely, 2003). The returns from the rules have lower volatility or less risk than buy-and-hold strategy. As a result, it's improper to represent only returns as the benefit of the rules. Neely (2003) extends Allen and Karjalainen (1999) by using risk-adjusted techniques. They test whether ex ante, optimal technical trading rules are beneficial on risk-adjusted basis after transaction costs and exclusion of dividends. Risk-adjusted measures are Sharpe Ratio, X^* statistic, and X_{eff} measure. The rules are trained with risk adjustment criteria and benefits are also measured with risk-adjusted returns. The result shows that, although risk-adjusted techniques can enhance the benefit of the rules, the rules still can't outperform buy-and-hold strategy. Nevertheless, the rules trained on X^* measure are the best among risk-adjusted techniques. Its benefit is approximately equal to the benefit from buy-and-hold strategy. They also characterize low-order serial correlation which causes predictability of GP trading rules claimed by Allen and Karjalainen (1999). The result indicates that the rules have forecasting ability and the X^* statistic seems to have the most forecasting ability.

Some researches use GP and technical trading rules with stocks (Potvin, Soriano, and Valee, 2004). They study 14 Canadian stock companies listed on Toronto Stock Exchange from 1992 to 2000. Each stock is selected from 14 different activity sectors. Apart from real-valued functions of Allen and Karjalainen (1999), relative strength index (RSI), rate of change (ROC), and volatility are added and they use both daily price and volume as variables. They also investigate whether excess returns from technical trading rules with genetic programming can make consistent excess returns compared to simple buy-and-hold strategy excluding transaction costs and dividends. They use rank-based method to avoid overselected rules with high fitness which can lead to suboptimal solution. The result suggests that the trading rules can’t make consistent positive excess return compare to buy-and-hold strategy. Even though, the rules are useful when the stock price is stable or fall, on the other hand, the rules are useless when the stock price rise. GP is implemented in foreign exchange markets as well (Neely, Weller, and Dittmar, 1997). They optimize technical trading rules with GP for six exchange rate, USD/DM, USD/JPY, USD/GBP, USD/SF, DM/JPY, and GBP/SF, from 1981 to 1995. They find the evidence to support consistent excess returns of all exchange rates produced by the trading rules including transaction costs. In further investigation, they find no systematic risk are compensated for usefulness of these trading rules as measured by beta calculated for several benchmark portfolios. Moreover, the rules for USD/DM can find the patterns which can’t be detected by standard statistical model, like random walk, ARMA, and ARMA-GARCH model. Not only using GP with technical analysis alone, but also applying GP with fundamental analysis (Neely and Weller, 1999). Neely and Weller (1999) combine technical trading rules and interest rate difference for four European Monetary System (EMS) exchange rates from 1979 to 1996. They think that difference level of interest rates in each country cans significantly affect their exchange rates. The studied exchange rates are DEM/FRF, DEM/ITL, DEM/NLG, and DEM/GBP, while overnight ECU central rates for each country that involved in 4 exchange rates are gathered. The result indicates that three of four exchange rates, DEM/FRF, DEM/ITL, and DEM/GBP, produce statistically significant excess returns including transaction costs. Moreover, the evidence has strong support for the important of interest rate difference. The low volatility of currency, such as NLG, cans reduce the profit of trading rules. As a consequence, DEM/NLG is the currency that can’t produce significant excess return. Similar to Neely, Weller, and Dittmar (1997), the trading rules can’t be detected by moving averages, filter rules, and two rules from target zone rates. Finally, like

Neely, Weller, and Dittmar (1997), no systematic risk are compensated for benefit of the trading rules as measured by beta calculated for two benchmark portfolio, MSCI World Portfolio Index and the Commerzbank Index of German stocks. For futures markets, technical analysis and GP is applied to 24 commodity markets (Roberts, 2003). The paper apply trinary decision tree instead of binary decision tree because, in futures markets, investors can have short position aside from long and no position. They also add some rules, trend lines, channel line, and gaps etc., other than the rules from Allen and Karjalainen (1999). They find 2 of 24 commodity markets earn significant profit after transaction costs. They point out to limitation of genetic programming that can't assure to create the global or even local optimal result. Furthermore, adding more based technical trading rules, such as momentum, and RSI, may make better results.

In Thailand, there are few researches involve with genetic programming. Leemakdej (2003) uses genetic algorithm to form an optimal portfolio that has lowest risk with the level of expected return. The method based on Xia, Liu, Wang, and Lai (2000) who extend Markowitz (1952) by using expected return instead of average return from historical data. Furthermore, Techa-intrawong (2005) applies genetic programming to solve the problem of market timing and index tracking with The Stock Exchange of Thailand (SET) index from 1995 to 2003. Rules from technical analysis are used as inputs. To apply the trading rules in real practice, they form the portfolio to track SET index with genetic programming as well. Excess return is used as fitness criteria. The result suggests that the rules and tracking portfolio have good result in training period. Nevertheless, there is no evidence that they can outperform the buy-and-hold strategy in out of sample period.

In this study, our research question will focus on whether risk-adjusted technical trading rules generated by genetic programming can outperform buy-and-hold strategy on SET50 index including transaction costs and exclusion of dividends. SET50 index was introduced on August 16, 1995. It's calculated from 50 large market capitalization and highly liquid stocks in SET. The index comprises of many sectors such as energy and utilities, banking, property development, construction materials etc. Moreover, SET50 index is used as a benchmark for performance of various products, for example, mutual fund. Recently, ETF on SET50 is released. It has similar properties to SET50 mutual fund but investor can buy or sell anytime during the trading period. The purposes of study are to introduce the methodology and show how to implement the risk-adjusted optimal technical trading rules with genetic programming in Thai stock market. It also provides another tool of technical analysis on SET50 index. Furthermore, it cans examine whether EMH is held on SET50 index. We use daily data of SET50 index from June 1, 1998 to May 30, 2008. There are four limitations of this study. Firstly, Number of data from SET50 index may be insufficient to optimize the trading rules. Secondly, the technical analysis functions don't include other functions such as fundamental analysis and other technical indicators which extend search area. Thirdly, there are several other specifications of genetic programming that can be used to optimize trading rules such as allowing more tree depth. Lastly, short selling is not allowed in this study.

II. THEORETICAL FRAMEWORK

Efficient market hypothesis (EMH)

Efficient market hypothesis asserts that financial markets are efficient. As a result, all known information is already reflected in the prices of traded assets. The efficient market hypothesis states that it is impossible to consistently outperform the risk-adjusted market return by using any known information after all transaction costs. Three common forms of the efficient market hypothesis are introduced by Fama (1970). They're called weak-form efficiency, semi-strong-form efficiency, and strong-form efficiency.

Weak-form efficiency asserts that all past market data, such as share prices, are fully reflected in financial securities prices. As a result, technical analysis techniques can't consistently produce excess returns by examining past price and volume. Furthermore, the securities price is random and there is no form for securities price. All future prices don't involve with historical price movement. However, in this case, trading based on some fundamental analysis techniques and private information can earn excess returns.

Semi-strong-form efficiency asserts that securities prices adjust rapidly to new public issued information and all public information is fully reflected in financial securities prices. As a result, both fundamental analysis technique and technical analysis technique can't consistently produce excess returns. However, trading based on some private information can possibly earn excess returns.

Strong-form efficiency asserts that both public and private information is fully reflected in securities prices. As a result, no one can earn consistently excess returns.

There are many criticisms about efficient market hypothesis led by new financial field, behavioral finance. Behavioral finance explains the inefficient of market that it occurs from human biased perception and irrational decision making process such as cognitive biased, information biased, human error in reasoning, and herd behavior. Speculative economic bubbles such as extremely high oil price, subprime crisis, and global financial crisis are obvious evidences of anomaly that threaten efficient market hypothesis.

Technical analysis

Technical analysis is a financial market technique that involves with the examination of past market data such as price and volume in order to forecast the future movement of securities prices and, thereby, make an investment decision. The idea is that securities price movement often repeats itself because investors' behavior seems to be the same pattern. Technical analysts try to identify price patterns and take the advantage of it. Technical analysis becomes popular since it's easy to implement compare to other analysis methods like fundamental analysis which requires several fields, such as accounting and economics. Moreover, the same knowledge can still be applied with various markets, for example, stock, foreign exchange, commodity and futures. Technical analysts also use indicators that are transformed from past price and volume matching with mathematical theory such as probability theory. Sample popular indicators are Moving Average, Support and Resistance Level, Moving Average Convergence/Divergence (MACD), Relative Strength Index (RSI), and Stochastic Oscillator. The indicators are also used to indicate whether the securities price is trending or sideways such as ADX.

Genetic programming (GP)

Koza (1992) introduces genetic programming which is a recent evolutionary algorithm motivated by biological evolution. Genetic programming is a randomized search procedure on a population of solution candidates, in this case, the trading rules. The population which is structured as decision tree evolves through time by several operators such as crossover, mutation, and reproduction. Then, the quality of each individual is evaluated according to a specific fitness function which defines the environment for the evaluation. Next, the individuals are selected at predefined amount. Finally, the selected individuals are set to be new population. The process is repeated itself until the new population converges to the old one that is expressed as the optimal population. The rules are optimal in the sense of fitness measure.

Genetic programming extends the conventional genetic algorithms by allowing the encoded non linear-structures such as decision tree. The parallel search maintains a population of possible solution area and permits the effective discovery of solution area. The genetic algorithms can be used to the problem that consists of discontinuous and non-differentiable function. They are beneficial to the objective function with many local optimal solutions as well. When other optimizations methods confront with difficulties about size of the search area, genetic algorithms can often be flourishing applied. The parallel search process is an effective method to reduce the uncertainty about the search area.

There are some limitations for genetic algorithms. For example, maintaining a population of solutions induces to increasing execution time because the objective function has to be evaluated several times. Due to the fact that genetic algorithms have little knowledge about particular problem, they are less effective than special-purpose algorithms. Furthermore, no theoretical convergence result is a drawback. The genetic algorithms should not be used to replace all other optimization methods. It depends on the character of question; as a result, it may be more appropriate to use them as a complement method.

Many earlier researches find whether the trading rules can outperform the market. However, genetic programming is used to find the optimal trading rules that can predict future securities price and test whether the rules can outperform the market. For trading strategy, it alleviates the problem of data snooping which occurs from ex post selection of rules since the rules are chosen by genetic programming. The common processes are shown as follow:

Initialization: An initial population is randomly created according to predefined specification, for example, amount of population, maximum depth, and possible element of each node. Next, each individual's fitness is evaluated with specific fitness function.

Selection: At this state, predefined amount of population are selected. The selection process based on population's fitness .It means that the rule with higher fitness value has more probability to be chosen than the lower one. The selected individuals are set to be the new population.

Modification: The population is modified by crossover, mutation, and reproduction operation. The new offspring is created and pooled with the selected population.

Evaluation: The selected population and new offspring are evaluated with specific fitness function.

Selection, modification, and evaluation process are repeated in order to develop the population quality. Those processes are iterated until the population quality converges to optimization point. As a result, a set of the most fitness rules is generated. The important related topics such as decision tree and operators are explained as follows:

Decision tree: Individuals, such as trading rules, are represented as tree structures. Tree can be recursively constructed from operator functions, variables, and real numbers. Every node consists of operator function except terminal node which consists of variables and real numbers. The pattern of mathematical expression makes it easy to evolve and evaluate. The sample decision tree is shown in Fig.1 as follows:

[Figure 1 is here]

Crossover: Crossover is an operation that evolves the trees by switching one node from a tree with another node from another tree. If the node isn't terminal node, switching one node means switching the whole branch. Element of the selected node from second tree has to correspond with the element of the selected node from first tree. However, in this research, crossover operation evolves the tree by discarding the selected branch from the first parent and copying the remaining branch to be a new offspring. For second parent, the selected branch is copied to combine with the one from the first parent to be a new offspring. The crossover operation is shown in Fig.2 as follows:

[Figure 2 is here]

Mutation: Mutation is an operation that diversifies the tree by replacing the node or the whole branch by a randomly generated node or branch. Element of the new node has to be correspond with the element of the selected node.

[Figure 3 is here]

Reproduction: Reproduction is an operation that copies the tree to be new population without any adjustment.

III. DATA AND METHODOLOGY

Trading procedures and return calculations

This research use genetic programming to optimize the trading rules for SET50 index. The goal is to test whether the trading rules generated from genetic programming can make excess return compare to buy and hold strategy. This goal can imply that it tests whether SET50 index market is efficient because historical price is used as an input to forecast market direction. Position is categorized into 2 types which are in the market (earning SET50 index return), and out of the market (earning risk free rate). Thus, short selling isn't allowed. For initial period, the position is out of the market. It's means that the investor holds cash. This cash is deposited in the bank and earn the risk free rate of return until the position is changed to be in the market. The trading rule in terms of decision tree will return Boolean value. If the Boolean value is true, the position in next period will be in the market and investor earns SET50 index return. On the other hand, if the return value is false, the position in the next period will be out of the market and investor earns risk free rate. We use 1-day repurchase rate announced by The Bank of Thailand as a proxy for risk free rate to match with daily return. We assumed it as a financial product that pays daily return on each business day and doesn't pay any return on holidays. Its daily return is measured as follows:

$$\text{daily risk free rate} = \frac{\text{annual 1-day repurchase rate}}{\text{number of business day in that year}} \tag{1}$$

Transaction costs are included to make the result closer to reality. It is added on the day that the position changes state, for example, switching from out of the market to in the market and vice versa. We use 0.25% one-way transaction costs which are same as what retail investors pay to brokerage.

Technical analysis

We follow Brock, Lakonishok, and Lebaron, (1992) that simple trading rules could make excess returns compare to buy-and-hold strategy without transaction costs. As a result, we apply variable length moving average and trading range break-out to be trading rules.

Variable length moving average (VMA)

Moving average is the simplest and widely used technical rule. The reason is that it smoothes the volatile series. Buy and sell signal are generated by two moving averages which are a short-period average and a long-period average. The buy (or sell) signal is made when the short-period average rise above (or fall below) the long-period average. When the short-period average crosses the long-period average, trend is considered to be created.

The formula of moving average line is showed as follows:

$$VMA = \frac{1}{N}\sum_{i=1}^{N} P_i \qquad (2)$$

where N is the amount of averaged periods

P_i is the closing price of the previous i^{th} period.

In this research, we use variable length moving average (VMA) whose holding period is varied. From the evidence in Thailand, Toeaditep (2003) suggests that variable length moving average can make overall profit better than using MACD indicator and buy-and-hold approach. To implement VMA, we define the rule as follows:

$$\text{VMA}(s,l)$$

where s is amount of periods of short-period average,

l is amount of periods of long-period average

Trading range break-out (TRB)

Trading range break-out consists of two categories, resistance and support levels. They are defined as the local minimums and maximums. For resistance level, it is believed that many investors would like to sell at peak. This situation causes resistance to a rising price. Nevertheless, if the price rises above the previous peak, it's considered to break the resistance level. This breakout generates a buy signal. For support level, it's believed that many investors would like to buy at bottom. However, if the price falls below the previous bottom, it's considered to break the support level. This breakout generates a sell signal. Furthermore, the band considered as the percentage of resistance (support) level, is also applied with trading range break-out to be the room for error. As a result, the buy (sell) signal is generated when price exceed support (resistance) level by amount of band. The resistance and support levels are defined as follow:

$$\mathrm{RES}(l,b) \text{ and } \mathrm{SUP}(l,b)$$

Where l is amount of periods used to find local minimums or maximums

b is a band measured in percentage difference compares to local minimums or maximums.

Genetic programming (GP)

The processes can be divided into 5 steps as mentioned earlier, for example, initialization, evaluation, modification, selection, and testing. The processes are in Fig.4.

[Figure 4 is here]

We divided time into 3 periods such as training period, selection period, and testing period. In training period, the trading rules are randomly initiated according to encoding principles. Then, they are evaluated with Sharpe ratio as fitness measure. The best rule will be selected to selection period. If there are many rules have the same highest fitness value, the best rule will be selected among these rules with equal probability. Meanwhile, the rules will be selected to modify with various evolutionary operation such as crossover and mutation. Moreover, the good rules are also selected to be new population through reproduction operation. The selection is probabilistic biased based on modified fitness value. Finally, new generation is created and they are evaluated again. The loop is repeated until we get the 50th generation.

In selection period, if best rule comes from the first generation, it is kept as the valid rule. However, for the best rule from other generation, it must be measure Sharpe ratio with data in selection period. If its Sharpe ratio is higher than the previous valid rule, it will be kept as the valid rule as well. This period is used to avoid overfitting problem of best rule from training period. Overfitting problem happens when the evolutionary rule is too fit with the data in training period. Consequently, the rule gets inferior result in other periods. The valid rule must have high Sharpe ratio in both training period and selection period.

Finally, in testing period, all valid rules are evaluated with new data set to test their efficiency compares to buy-and-hold approach. The detail in each step is described as follow:

Encoding

This research represents trading rules as decision trees. The tree consists of various nodes which can be functions or operands. The operand appears only at terminal node. For each function, we follow Allen and Karjalainen (1999) with simple function such as average, minimum, maximum and lag of price. The reason is that plentiful trading rules with specific patterns can be combination of minimum and maximum of historical prices (Neftci, 1991). Moreover, we add Variable length moving average (VMA) and Trading range break-out (in term of support and resistance) as functions. Finally, we added variance as a function and integers from 0 to 250 as operands (Potvin, Soriano, and Valee, 2004).

The functions and terminals in each node can be defined as follows:

Functions:

Arithmetic operators:	+, -, *, /;
Boolean operators:	and, or, not;
Relational operators:	<, >;
Real functions:	
norm(r_1, r_2):	Absolute value of the difference between two rational numbers;
avg(*p*,*n*):	Average of price over the past n days;
max(*p*,*n*):	Maximum value of price over the past n days;
min(*p*,*n*):	Minimum value of price over the past n days;
lag(*p*,*n*):	Price is lagged by n days;
var(*p*,*n*):	Variance in daily returns over the past n days;

Where *p* is the function of SET50 index or current level of SET50 index

If *n* is not an integer, *n* will be modified with floor function to be an integer

Technical trading rules:

VMA(*s*,*l*):	Variable length moving average;
RES(*l*,*b*):	Resistance level
SUP(*l*,*b*):	Support level

Terminals:

Non negative integer:	It's chosen from 0 to 250; where 250 is the approximate number of business day in a year;
Boolean constants:	True, False;
Real variables:	Current level of SET50 index (P in decision tree)

Initialization:

Before initialization, the amount of population is defined to 500 and maximum depth of tree are 6 (Allen, and Karjalainen, 1999). As a result, it can be constructed up to 63 maximum nodes. The population of trading rules is randomly initialized as decision trees. The tree is produced with Grow methods. For this method, the length of tree from root node to terminal node can be varied. However, the length of tree mustn't exceed the predefined depth. Owing to tree structures, the recursive constructed process occurs from root to leaves and the terminals, except Boolean constants, can be chosen at any depth except root. The rules of node selection can be described as follow:

Firstly, root is chosen among Boolean constants, Boolean operators, technical trading rules, and relational operators. Secondly, the descendents may be chosen among any functions and terminals which correspond with the upper node. For example, when a relational operator is chosen, the descendents have to be arithmetic operators, real functions, or terminals. The method for constructing the decision tree is shown in Fig. 5.

[Figure 5 is here]

Evaluation: *Risk-adjusted measures*

The risk-adjusted measures are considered to be fitness criteria because the trading rules sometimes stay out of the markets; as a result, it may bear less risk than buy-and-hold strategy. Using excess return as fitness criteria might not be good for risk-averse investors. According to utility theory, risk-averse investors may prefer lower return with lower risk than higher return with higher risk. This risk-adjusted measure used in this paper is the Sharpe ratio. We use Sharpe ratio since it's widely accepted to measure the performance. Sharpe ratio is the excess return per unit of risk. It's represented by daily asset return minus risk free rate and divided by standard deviation of the excess return. The formula is shown as follows:

$$\text{Sharpe ratio} = \frac{\mathrm{E}[R] - r_f}{\sigma} \tag{2}$$

where $\mathrm{E}[R]$ is expected return,

r_f is risk free rate,

σ is standard deviation of the excess return

The daily excess return over risk free rate excluding transaction cost at time t is provided as follows:

$$\mathrm{E}[R] - r_f = z_t\left[\ln\left(\frac{P_{t+1}}{P_t}\right) - \ln(1 + i_t)\right] \tag{3}$$

where z_t is equal to 1 if the rule is in the market and 0 if the rule is in the market,

P_t and P_{t+1} are the SET50 index at time t and t+1 respectively,

i_t is daily risk free rate from (1)

Nevertheless, if the excess return is negative, for example, the market crashes, the result is hard to interpret and compare with buy-and-hold approach. Furthermore, it cans imply that only deposit money in the bank will be outperform SET50 index. As a result, we will choose only sideways market and rising market that yield positive excess return to make Sharpe ratio positive.

Selection for new generation:

The rules are sorted based on their Sharpe ratio from the worst rule to the best rule. As Allen and Karjalainen (1999) suggested, we use a rank-based method to modify the fitness by assigning 1 as the fitness value to the worst rule and 500 to the best rule. The fitness value increases by 1 to each consecutive better rule. When there is the same Sharpe ratio for 2 or more rules, their fitness are equaled and they are equal to the average fitness of them. This ranking method modifies the raw fitness values to avoid over selection of super-trading rule with high fitness because it bound the highest fitness value to 500. This over selection leads to convergence on a possible suboptimal solution. This method also avoids negative ranking which is difficult to find the probability of selection. Furthermore, when all trading rules have similar fitness value, this ranking method also enlarges a gap between very close fitness values in order to mitigate a random selection.

Then, the new fitness value is modified to the probability of selection. This process is probabilistic biased in favor of higher fitness value. We use fitness-proportionate selection scheme which is shown as follows:

$$prob_i = \frac{f_i}{\sum_{j=1}^{N} f_j} \tag{4}$$

where $prob_i$ is selection probability of the i^{th} trading rule

f_i is fitness of the i^{th} trading rule

N is amount of trading rules in population.

Modification: Crossover, Mutation, and Reproduction

The basic modifications as explained in theoretical framework are applied. However, there is a special case for crossover operation. When the node in the first tree is chosen, compatible nodes in the second tree is identified to be cross points. If there is no compatible node in second tree, we will select the new parents until they can create an offspring. For 500 rules in new generation, 200 rules come from crossover operation, 200 rules come from mutation operation, and 100 rules from reproduction.

Data

This research focuses on the SET50 index. We gather SET50 index from SETSMART and 1-day repurchase rate from Datastream. The data are on daily basis from June 1, 1998 to May 30, 2008. Each investment horizon is divided into 3 periods. The training period comprises of rising, stable, and falling markets which cans train the trading rules to learn from all market conditions. They are shown as follows:

Training period:	June 1, 1998 to May 31, 2002
Selection period:	June 3, 2002 to May 31, 2004
Testing period:	June 1, 2004 to May 30, 2008

IV. Empirical Result

Table I shows the result from applying genetic programming using Sharpe ratio as the fitness measure. When the position changes state, 0.25% one-way transaction is subtracted from return on the same day.

[Table I is here]

Training and selection period

As expected, the result gets high Sharpe ratio in training period and selection period. The average Sharpe ratio over these two period are 0.078 and 0.094 respectively, while Sharpe ratio from buy-and-hold, 0.003 and 0.06, and Techa-intrawong (2005), 0.04 and 0.054, are much lower. Moreover, the average r_b-r_s from our rules are 0.151 and 0.107 respectively. Average trades are 5.8 and 2.6 respectively. The percentage of time that the rule is in the market increases from 9.56% in training period to 30.45% in selection period. The result is correspondence to market condition in each period. The market is sideways in training period and rising in selection period.

Testing period

The average Sharpe ratio of our rules is equal to 0.007 which is lower than buy-and-hold approach 0.014, while the rule from Techa-intrawong (2005) yields negative value, -0.01. This means that such rule may not give superior result in all market condition. Moreover, the best rule (The 1st rule in Table I) yields 0.018 Sharpe ratio higher than the one from buy-and-hold strategy. The average r_b-r_s is equal to -0.002 which is lower than 0.08 from buy-and-hold. Our best rule yield 0.049 of r_b-r_s lower than buy-and-hold strategy as well. The average trades are 3.2 and average %Long in the market are 31.50%.

Optimal technical trading rules

[Figure 6 is here]

One of the trading rules provided by our program is shown in Fig.4. This trading rule is the first rules provided in Table I. The reduced form can be represented as Var(Max(P,P),P) < 177. It means that if variance of Max(P,P) function in previous level of SET50 periods is less than 177, investors should hold SET50 index and hold cash otherwise. Max(P,P) function means that the maximum level of SET50 index in previous level of SET50 index periods. This rule indicates that SET50 index tend to rise when the variance is low and fall when variance is high. Position with SET50 index is shown in Fig.5.

[Figure 7 is here]

Furthermore, other rules (The 2nd to 5th rule in Table I) are shown in Fig.6 to Fig.9. They are involved with volatility as well. Similar to the first rule, they indicates that SET50 index tend to rise when the variance is low and fall when variance is high. The second rule can be represented as Var(Max(P,P),Add(Div(P,41),Var(P,P))) < 164. It means that if variance of Max(P,P) function in Add(Div(P,41),Var(P,P)) previous periods is less than 164, investors should hold SET50 index and hold cash otherwise. Add(Div(P,41), Var(P,P)) function is the result from adding Div(P,41) function and Var(P,P) function. Div(P,41) function provides the result from dividing current level of SET50 index with 41. Var(P,P) function return the variance of SET50 in previous level of SET50 index periods. The third rule can be represent as Var(Max(P, P), P) < 75. It means that if variance of Max(P,P) function in previous current level of SET 50 index periods is less than 75, investors should hold SET50 index and hold cash otherwise. The reduced form of forth rule can be represent as Var(Max(Sub(P,1),P),P) < P. It means that if variance of Max(Sub(P,1),P) function in previous current level of SET50 index periods is less than current level of SET50 index, investors should hold SET50 index and hold cash otherwise. Max(Sub(*P*,1),*P*) function returns the maximum value of Sub(*P*,1) function in previous current level of SET50 index periods. Sub(*P*,1) function is the result from subtracting current level of SET50 index with 1. Finally, The fifth rule cans be represent as Var(Max(Sub(*P*,Div(*P*, 222)),*P*),*P*) < *P*. It means that if variance of Max(Sub(*P*,Div(*P*, 222)),*P*) function is less than current level of SET50 index, investors should hold SET50 index and hold cash otherwise. Max(Sub(*P*,Div(*P*, 222)),*P*) function returns the maximum value of Sub(P,Div(*P*, 222)) function in previous current level of SET50 index periods. Sub(*P*,Div(*P*, 222) function is the result from subtracting current level of SET50 index with Div(P, 222) function. Div(*P*, 222) function is the result from dividing current level of SET50 index with 222.

[Figure 8, 9, 10, 11 are here]

V. Conclusion

This study is focused on applying genetic programming with SET50 index to find the optimal trading rules which based on technical analysis. Our method follows Neely (2003) who extends Allen and Karjalainen (1999) by taking into account the risk-adjusted technique, for example, Sharpe ratio. 0.25% one-way transaction cost is also included in the return.

Our result suggests that the best rule (1^{st} rule in Table I) has higher Sharpe ratio, 0.018, than buy-and-hold approach, 0.014, in testing period. However, we can not claim that the market is inefficient because the trading rule may not make superior result in other periods. The average Sharpe ratio of our rules is equal to 0.007 which is lower than buy-and-hold approach while the rule from Techa-intrawong (2005) yields negative value, -0.01. The average r_b-r_s is equal to -0.002 which is lower than 0.08 from buy-and-hold. Our best rule yield 0.05 lower than buy-and-hold strategy as well. The average trades per year are 0.08 and average %Long in the market are 31.50%.

REFERENCES

Allen, F., Karjalainen, R., 1999, Using genetic algorithms to find technical trading rules, *Journal of Financial Economics* 51, 245–71.

Brock, W., Lakonishok, J., LeBaron, B., 1992, Simple technical trading rules and the stochastic properties of stock returns, *Journal of Finance* 47, 1731-1764.

Fama, E.F., 1970, Efficient capital markets: a review of theory and empirical work, *Journal of Finance* 25, 383-417.

Koza, J.R., 1992, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, (MIT Press, Cambridge, MA).

Leemakdej, A., 2003, Stock portfolio formation by genetic algorithm, Working paper, Thammasat University.

Markowitz, H., 1952, Portfolio Selection, *Journal of Finance* 7, 77-91.

Merton, R.C., 1980. On estimating the expected return on the market: an exploratory investigation. *Journal of Financial Economics* 8, 323-361.

Neely, C.J., Weller, P., Dittmar R., 1997, Is Technical Analysis in the Foreign Exchange Market Profitable? A Genetic Programming, *Journal of Financial and Quantitative Analysis* 32, 405-426.

Neely, C.J., Weller, P., 1999, Technical trading rules in the European Monetary System, *Journal of International Money and Finance* 18, 429-458

Neely, C.J., 2003, Risk-adjusted, ex ante, optimal technical trading rules in equity markets, *International Review of Economics and Finance* 12, 69-87.

Neftci, S.N., 1991, Naive trading rules in financial markets and Wiener-Kolmogorov prediction theory: a study of “technical analysis”, *Journal of Business* 64, 540-571.

Ready, M. J., 1998, Profits from technical trading rules, Working paper, University of Wisconsin, Madison.

Roberts, M.C., 2003, Technical Analysis and Genetic Programming Constructing and Testing a Commodity Portfolio, *Journal of Futures Markets* 25, 643-660.

Potvin, J.-Y., Soriano, P., Valee, M., 2004, Generating trading rules on the stock markets with genetic programming, *Computers & Operation Research* 31, 1033-1047.

Techa-intrawong, S., 2003, MARKET TIMING AND INDEX TRACKING BY GENETIC PROGRAMMING, Thesis, Chulalongkorn University.

Toeaditep, S., 2003, Profit Making by Utilizing Technical Analysis of Investments, Special project, Chulalongkorn University.

Xia, Y., Liu, B., Wang, S., and Lai, K.K., 2000, A Model for Portfolio Selection with Order of Expected Returns, *Computers & Operations Research* 27, 409-422.

Table I: Summary of Result

The table provides the statistics of trading rules found by genetic programming compared to buy-and-hold approach and the rule found by Techa-intrawong (2005). Each statistic is divided into 3 periods which are training (1), selection (2), and testing (3) period. The first column provides rules from each method which our results are the first to the fifth rule. r_b - r_s is the difference of SET50 index return when the rule is in the market and out of the market. r_b - r_s is shown on yearly basis and Sharpe ratio is shown on daily basis. %Long is the percentage of time when the rule is in the market compare to overall time.

Rules	Sharpe ratio			r_b - r_s			Trades			% Long		
	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
1st	0.079	0.105	0.018	0.184	0.171	0.049	6.0	6.0	4.0	11.88%	29.80%	38.61%
2nd	0.082	0.101	0.026	0.075	0.048	0.045	1.0	1.0	1.0	4.57%	20.20%	21.04%
3rd	0.088	0.100	-0.023	0.091	0.067	-0.106	5.0	2.0	5.0	2.64%	20.20%	10.93%
4th	0.071	0.085	0.008	0.204	0.137	0.006	8.0	2.0	3.0	14.31%	40.82%	43.41%
5th	0.070	0.079	0.006	0.202	0.114	-0.004	9.0	2.0	3.0	14.42%	41.22%	43.51%
Mean	0.078	0.094	0.007	0.151	0.107	-0.002	5.8	2.6	3.2	9.56%	30.45%	31.50%
Buy-and-hold	0.003	0.060	0.014	0.055	0.245	0.082	1.0	1.0	1.0	99.90%	99.80%	99.90%
Techa-intrawong (2005)	0.040	0.054	-0.010	0.404	0.146	0.008	35.0	15.0	39.0	53.10%	58.16%	59.86%

Figure 1

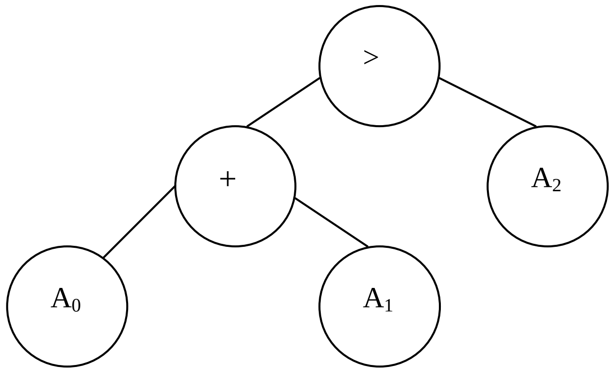

Fig.1 Above tree structures represent testing whether $A_0 + A_1 > A_2$, where A_0, A_1, A_2 are constant numbers. If the expression returns true value, it will release some signal such as buy signal. Meanwhile, if the expression returns false value, it release some signal such as sell signal.

Figure 2

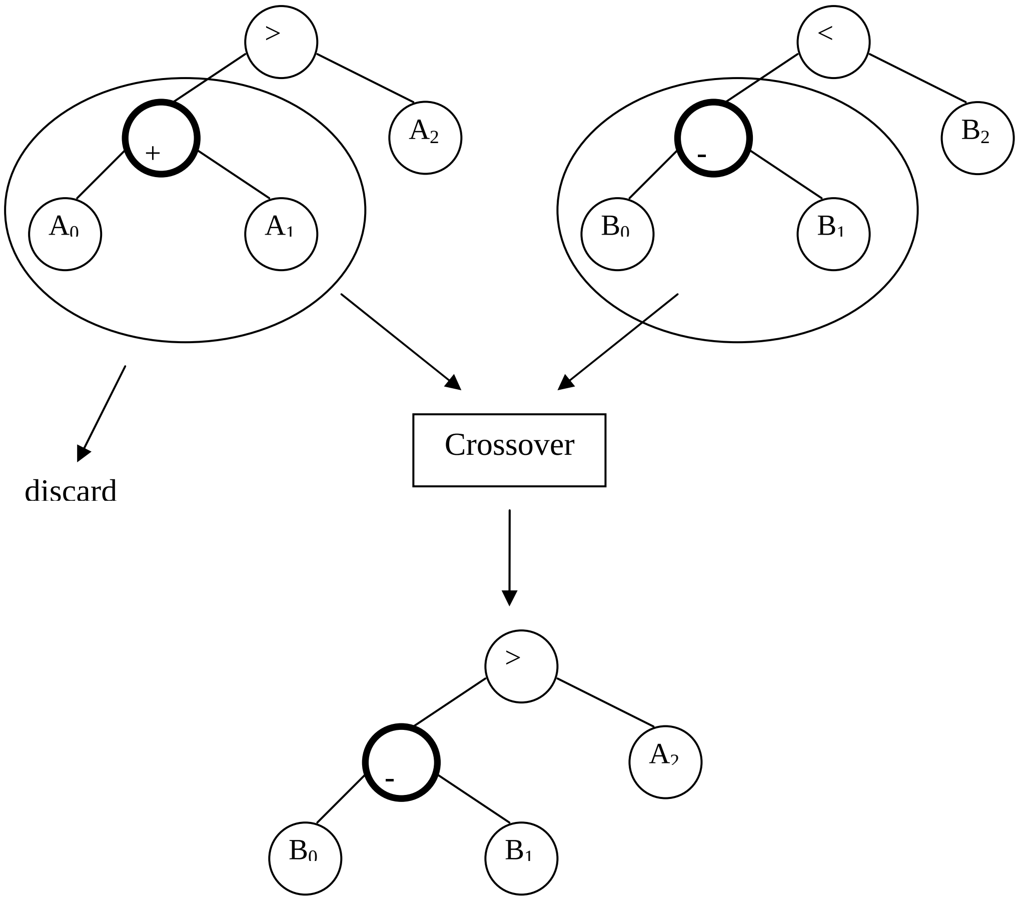

Fig.2 Crossover operation is applied with two parents, $A_0 + A_1 > A_2$ and $B_0 - B_1 < B_2$. The branches whose roots are by + and – from the first and second tree are selected. As a consequence, the offspring, $B_0 - B_1 > A_2$, is generated.

Figure 3

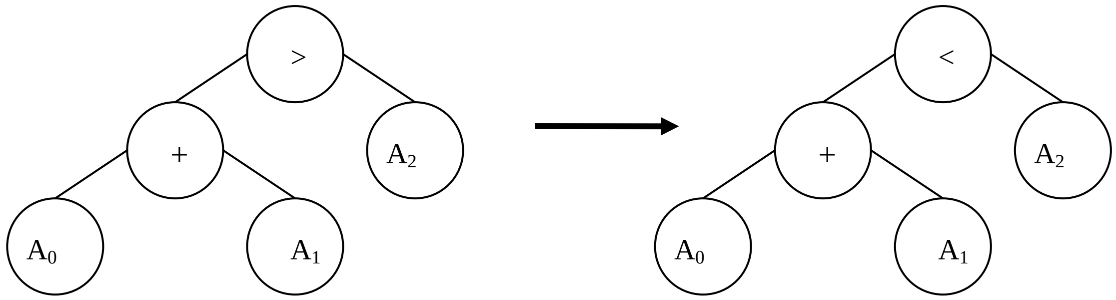

Fig.3 Mutation operation is applied with single decision tree, $A_0 + A_1 > A_2$. Root node with 'greater than' (>) operator is selected and the operator is changed to 'less than' (<) operator. New decision tree represent $A_0 + A_1 < A_2$.

Figure 4

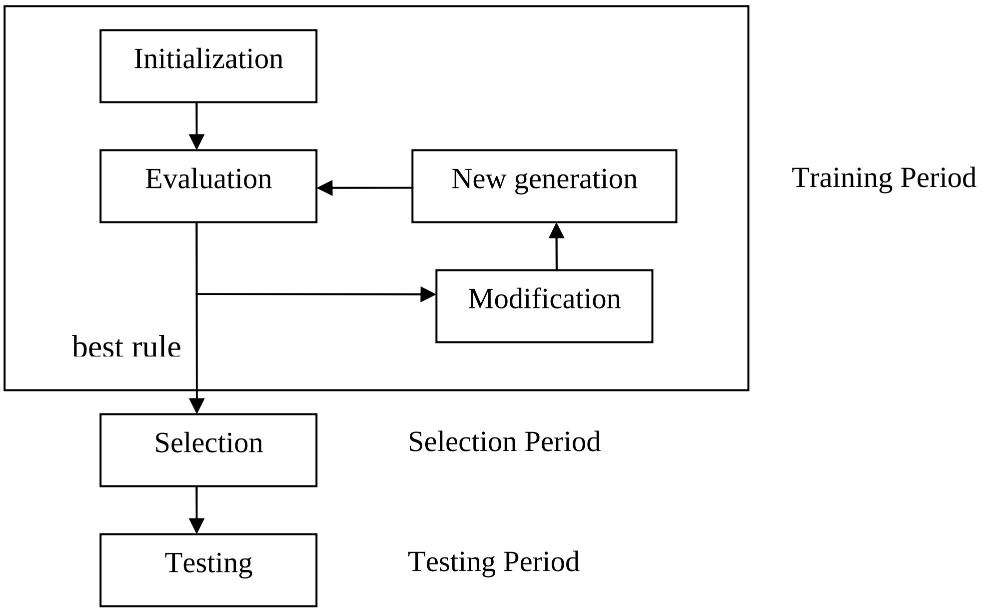

Fig.4 The processes of genetic programming consist of training, selection, and testing period.

Figure 5

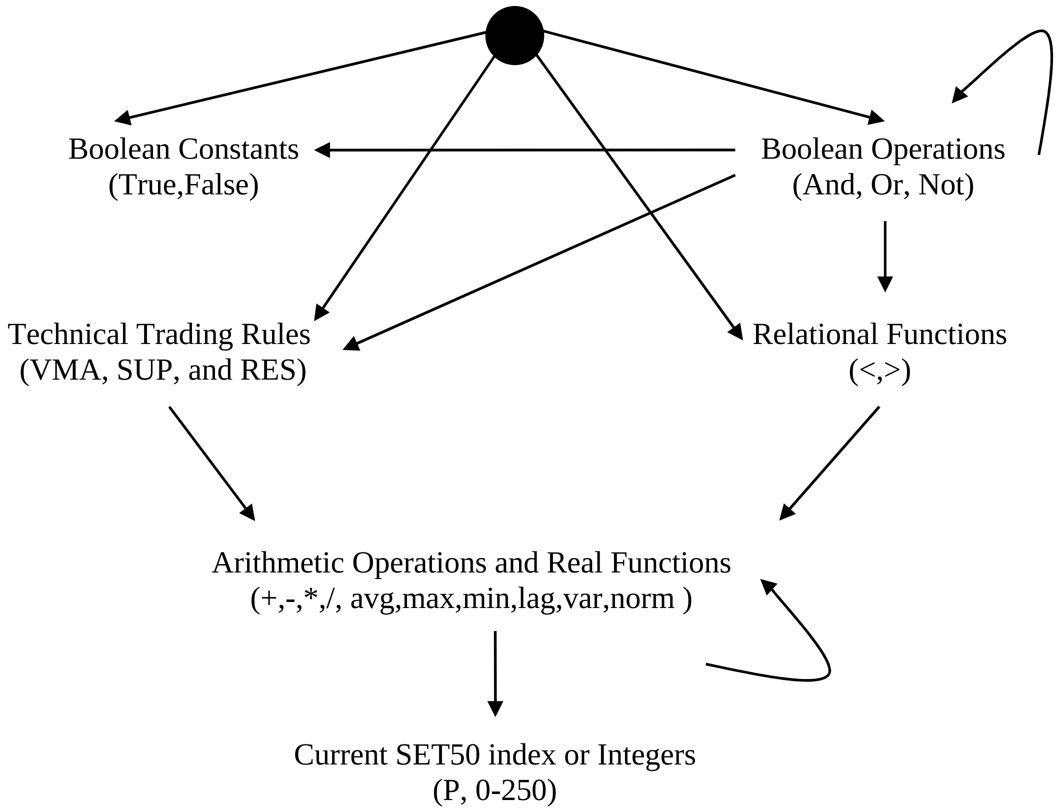

Fig.5 Method for constructing decision tree. Tree is constructed from root which can be Boolean constants, Boolean operations, technical trading rules, and relational operations. The descendent node must be compatible to upper node as explained by above picture.

Figure 6

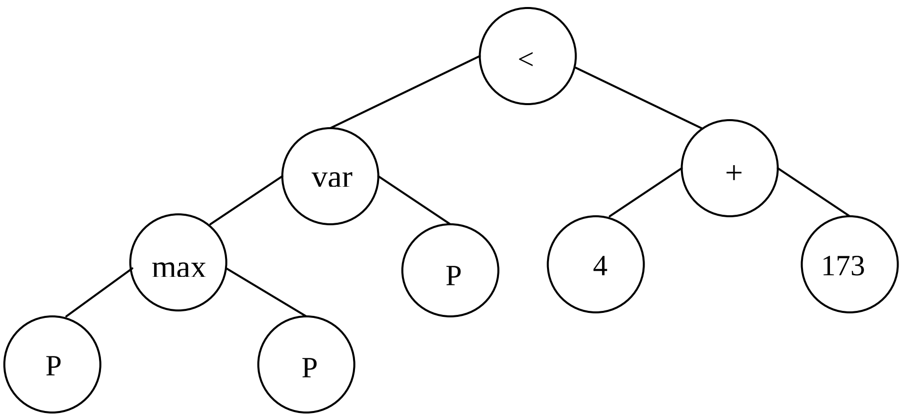

Fig.6 The 1st trading rule in Table I. The reduced form cans be represented as Var(Max(*P*, *P*), *P*) < 177. Where *P* is current level of SET50 index.

Figure 7

Fig.7 Position and SET50 index are plot together. Value of SET50 index is on the first y-axis, the left side. Position value is on secondary y-axis, the right side. When the rule is in the market, the position is equal to 1, while the position will be 0 when the rule is out of the market.

Figure 8

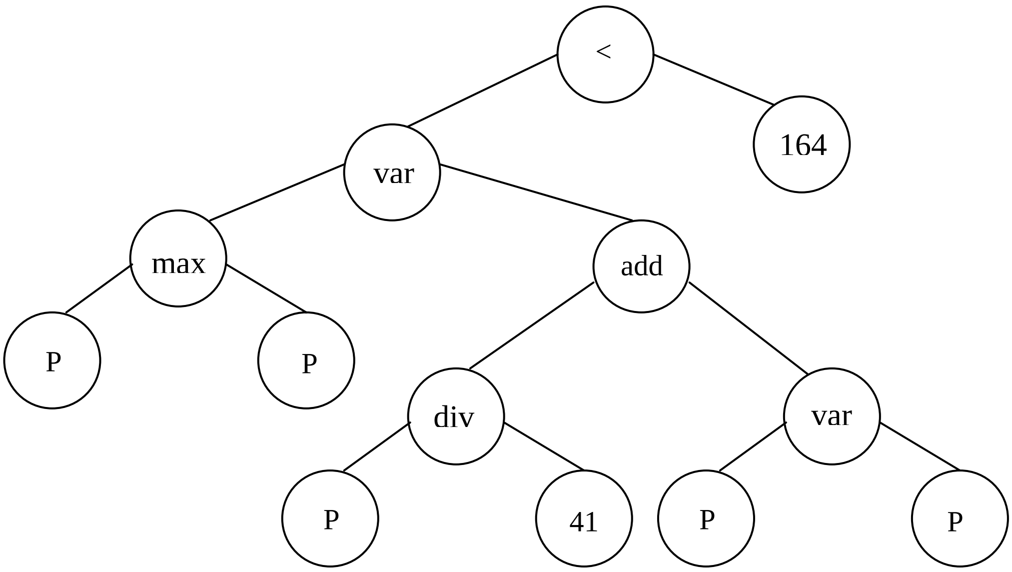

Fig.8 The 2nd trading rule in Table I. It cans be represented as Var(Max(P, P), Add(Div(P, 41), Var(P, P))) < 164. Where P is current level of SET50 index.

Figure 9

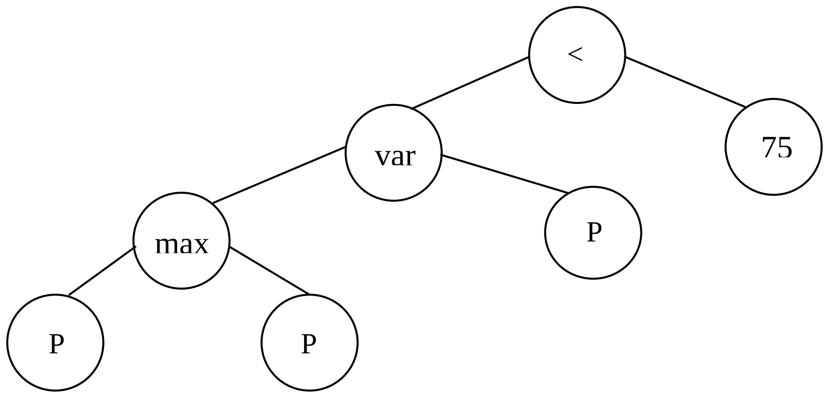

Fig.9 The 3rd trading rule in Table I. The reduced form cans be represented as Var(Max(*P*, *P*), *P*) < 75. Where *P* is current level of SET50 index.

Figure 10

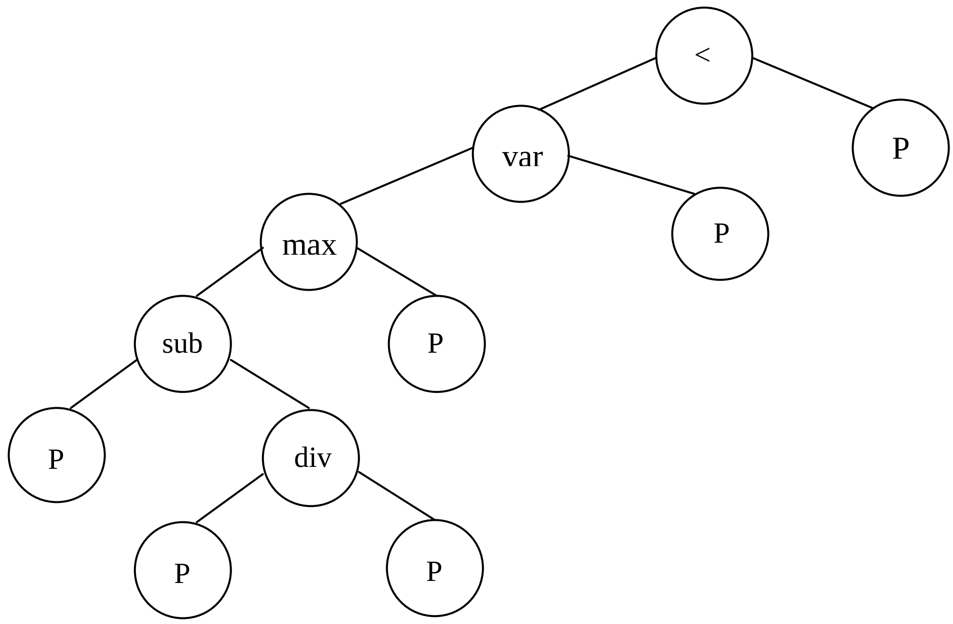

Fig.10 The 4th trading rule in Table I. The reduced form cans be represented as Var(Max(Sub(*P*, 1), *P*), *P*) < *P*. Where *P* is current level of SET50 index.

Figure 11

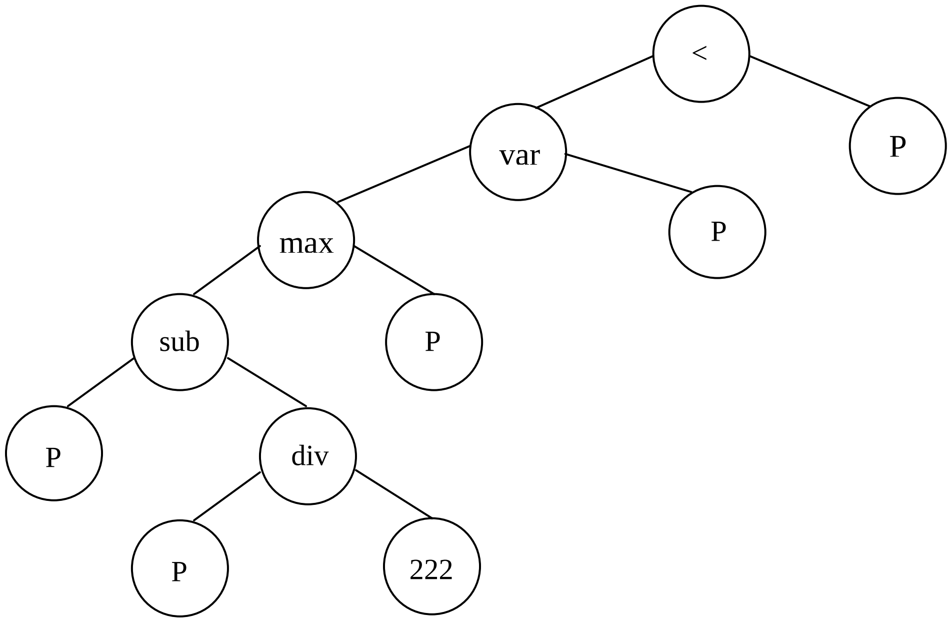

Fig.11 The 5th trading rule in Table I. It cans be represented as Var(Max(Sub(*P*, Div(*P*, 222)), *P*), *P*) < *P*. Where *P* is current level of SET50 index.