Generating Intraday Trading Rules on Index Future Markets Using Genetic Programming

Liu Hongguang and Ji Ping

***Abstract*—Technical analysis has been utilized in the design of many kinds of trading models. Recent results indicate that this market timing approach beats the traditional buy and hold approach in most of the short term trading periods. Genetic programming based on technical analysis was introduced as a mean to automatically generate and optimize short term trading rules on the stock markets in the last decade. Most of the study in history applied genetic programming to the financial time series daily, weekly or even monthly, and reported that no excess return can be obtained with the consideration of transaction cost. This study uses intraday time series to fully exploit the short term forecasting advantage of technical analysis. The utilization of intraday data to train the trading rules can avoid the jumping points in the daily or monthly data, as relevant information is assumed to be fully digested during the market close periods. The emerging market of China Hushen 300 Index Future contract is used to test the efficiency of the proposed trading strategies developed by genetic programming, positive return is obtained.**

***Index Terms*—Intraday trading, genetic programming, technical analysis, HuShen 300 index future.**

I. Introduction

Financial time series related study has drawn the interest of researchers and investors for decades. Many researches have been conducted to uncover the mystery of the movements of financial time series. Both fundamental and technical information of the corresponding financial times series have been utilized in the decision making process. An agreement has been reached that there is no perfect model that could capture all the characteristics of the time series. All that we can do is to find optimal models for a specific time series with certain useful characteristics that could be used in the design of the trading system.

In terms of the design of financial trading system, especially for the target assets of stock index or index future, most of the research frequencies are days, weeks and even months. Some of the models showed indeed outstanding performance in this kind of “long” term trading period. Many researchers have shown the profitability of those trading systems. However, these kinds of theoretical profits are highly unlikely to be obtained from the real markets. This is because most of the trading in future markets is margin trading, some small movements of the stock index could mean large amount of gains or losses. For most of the time, investors do not have enough money to wait for the market to go to the “right” price.

Therefore, the high frequency trading system showed its advantages in such kind of conditions. Many researchers and investors are trying to build automatic trading systems that can generate profit from certain markets every day; high frequency trading is one option to accomplish this target by trading many times with very few profits each time. However, high frequency trading has some drawbacks: firstly, the cost to conduct high frequency trading is very high, extreme advanced trading machines need to be placed as near as possible to the Exchanges; secondly, high frequency trading is strictly supervised by the government in some countries like China, as the data coming out of the exchanges of China is not in real high frequency, it is impossible for the deployment of high frequency trading systems. Thus, intraday trading became the most attractive trading strategies in such countries.

Some of the researchers have conducted the experiments of designing technical trading systems based on genetic programming; however, all of the studies used the daily, weekly or even monthly financial time series. This paper utilizes the intraday high frequency data to build trading strategies. Trading pattern is assumed to stay stable within a single trading day, as relevant information which will lead to jump points, is assumed to be fully digested during the market close periods, making technical analysis more reliable.

II. Literature Review

This review consists of three aspects of studies. The first aspect is technical analysis, the second aspect is the applications of genetic programming to financial analysis, and the last aspect is the study of intraday trading systems.

A. Technical Analysis

Technical analysis is a class of methods and strategies that exploit the short term fluctuation of the security prices. Technical analysis focused on historical data in the discovery of useful information and prediction of future stock price; despite the fact that it has been criticized and scorned by many academic researchers and practitioners, technical analysis has proved itself to be useful and powerful in both academia and practice. Gencay, R., F. Fernańdez-Rodrı́guez *et al.*, and Chavarnakul, T., and D. Enke presented extensive reviews on the relevance of technical analysis in financial markets and the profitability of technical pattern [1]-[3]. Technical trading rules were proved by a considerable amount of work to be capable of producing valuable economic signals in both stock markets and foreign exchange markets. However, technical analysis was proved by Alexander and Fama to be not profitable [4], [5]. The conclusion that technical analysis cannot beat the market was

Manuscript received February 28, 2015; revised April 20, 2015.

The authors are with the Hong Kong Polytechnic University, Hong Kong (e-mail: redblack.liu@gmail.com, p.ji@polyu.edu.hk).

DOI: 10.7763/IJTEF.2015.V6.453

challenged by Pruitt and White, who showed that trading systems designed based on technical indicators are profitable [6]. Taylor and Allen reported that up to 90 percent of traders use some sort of technical analysis in their practical trading [7].

B. Genetic Programming

Developed by Holland, genetic algorithms were first combined together with technical trading rules which were able to generate excess returns in the US exchange market by Bauer [8], [9]. Based on the study of genetic algorithms, Potvin et al. introduced the genetic programming, which has now been used in a number of research areas, especially in finance and technical analysis [10]. Decision trees are generated in genetic programming in the searching of better rules. Koza provided a flexible framework for adjusting the trading rules to the current environment [11]. Since then, Many researchers have been trying to find trading rules that can generate excess returns, among them, Chen and Yeh and Allen and Karjalainen failed in their attempt of finding excess returns over buy and hold approach on stock indexes, however, Neely *et al.*, Neely and Weller and Marney *et al.* found otherwise, at least when the notion of risk is not considered [12]-[16].

Another work worth to be mentioned here was conducted by J. Y. Potvin *et al.*, instead of testing the genetic programming based trading systems on market indexes and exchanges rates, they adjusted the rules to 14 individual stocks listed on Toronto stock exchange market [17]. The experiments results of each stock vary from each other, some of which are profitable, which some of the others may suffer heavy losses. One of the major drawbacks of this study is the neglecting of transaction cost, which makes it less practical. With the consideration of transaction cost, Mallick *et al.* and Esfahanipour *et al.* proved the capability of generating excess returns over buy and hold strategies in the markets [18], [19]. Taking one step ahead, Akbar Esfahanipour and Somayeh Mousavi considered risk adjusted excess returns in their study, a three signals technical trading rule on individual stock was built in their study; the performance of their system beats the buy and hold strategy in all market conditions. Janko Straflburg *et al.* proposed parallelization to improve the generation of technical trading rules by speeding up the computing and enabling more results [20].

C. Intraday Trading Systems

High-frequency trading is quantitative trading that is characterized by short portfolio holding periods. All trading decisions are made by computerized quantitative models. Modern High frequency trading is believed to start from 1999, after the authorization of electronic exchanges in 1998 by U.S. Securities and Exchange Commission (SEC). At the very beginning of 21st century, the minimum time taken by high frequency trading was several seconds, whereas by 2010 this had declined to millisecond or even microsecond. According to the data from NYSSE, the high frequency trading volume soared by 164% from 2005 to 2009. Liquidity provision is another key factor that leading to the rapid growth of high frequency trading. Many market makers introduced high frequency trading to the market with lower volatility and wider bid-ask spreads.

Neely and Weller tested the performance of two technical intraday trading strategies — a genetic program and an optimized linear forecasting model on the exchange market. They stated that no excess returns could be gained by either strategy when realistic transaction costs and trading hours are taken into consideration. Thus they claimed the results to be consistent with market efficiency theory; however, as more and more high frequency trading team reported positive excess return over market return, there is a high possibility that their strategies' poor performance might be affected by the selected models and the specified assets. Exchange rate always tends to stay at a certain level, thus linear forecasting model's bad performance maybe simply caused by the special properties of the specified asset [21].

With out-of sample data, Kablan Abdalla introduced Sugeno model as implementation of the ANFIS (Adaptive Neuro Fuzzy Inferences Systems) in the forecasting of high frequency financial series and the estimation of parameters. In this research, Kablan proposed a system that performs a prediction on the financial times series in high frequency using intraday data (5-minutes intraday trading prices), buy and sell orders will be placed based on the predication of the movement of the prices; although this system was observed to have a very high accuracy rate and opening the right position at the right place, however, theses trading did not always translate into higher returns due to transaction costs. He then modified the trading strategy by holding the assets for a relatively "longer period" –do not change position as long as the current position matches the latest prediction, the performance was improved as the reduced of number of trading, hence the reduction of transaction costs. Kablan introduced a trigger point in the final modification of the system; the results of the systems' performance was very promising compared to the traditional buy and hold strategy, with positive Sharpe and Sortino ratios, which means that the system did not take higher risk for the amount of return gained [22].

By far, most of the profitable trading models are still holding as essential assets, most of which are protected commercial laws. This paper showed the basic components that are needed to create a profitable trading strategy, based on which further study could be conducted to improve the profitability of all kinds of intra-day trading strategies.

III. Generating Trading Rules with Genetic Programming

The process to conduct the adaptation of genetic programming is shown below. The genetic programming will generate trading rules based on technical indicators encoded as programs. The generated trading rules will return either a buy or sell signal.

A. Encoding of the Technical Indicators

The initial computer programs represented as tree structures are recursively constructed from a predefined set of functions *F* and Terminals *T*. Both functions *F* and Terminal *T* change dynamically during the evolution process. Functions *F* and Terminal *T* are defined in the following (J. Y. Potvin *et al.* 2004).

Functions:

Arithmetic operators: +,-,*,/;

Boolean operators: and, or, not;

Relations operators: <,>;

Boolean functions: if-then-else.

Real functions (user defined functions, here are the technical indicators)

Technical indicators, variable *s* represents for constant Price:

Norm(r1,r2): absolute *value of the difference between real number;*

Avg(s,n): Average *of price over the past n periods;*

EMA(s,n): Exponential *moving average of the past n periods;*

Max(s,n): Maximum *value of price over the past n periods;*

Min(s,n): Minimum *value of price over the past n periods;*

Lag(s,n): Price value lagged by n periods;

Valotility(n): Variance *in returns over the past n periods;*

RSI(n): Relative *strength index;*

ROC(n): *Rate* of *change.*

$$ROC(n)=\left(\frac{closing\ price\ of\ the\ current\ minute}{closing\ price\ of\ n\ minutes\ ago}-1\right)\times 100 \quad (1)$$

$RSI(n) = 100\text{-}(\frac{100}{1+RS(n)})$, where $RS(n)=\frac{\sum_{i\in D^{+}(n)} r_i}{-\sum_{i\in D^{-}(n)} r_i}$, D^{+} is the set of minutes with rising prices, D^{-} is the set of minutes with falling prices and r_i is the return of minute i, which is positive when the price is rising and negative otherwise.

Terminals:

Constants: chosen in the interval [0,270], where 270 is the approximate number of minutes in a single trading day;

Boolean: True, False;

Others: Price

Real variable: P price of the current minute;

Order Types: Entry order and Exit order

Entry order: Market Entry order: enter into the market at market price;

Stop Entry order: these orders are placed above the market for a long entry and below the market for a short entry;

Limit Entry order: these orders are placed below the market for a long entry and above the market for a short entry;

Exit order: Exit at target profit, Exit at target percentage profit, Exit at Target price, Protective Stop, Trailing Stop, Exit after N Bars, Exit after N Bars profit, Exit after N Bars loss, Exit after certain time, Exit at Market, Exit End-of-Day.

It is obvious that the functions and terminals sets chosen in this study violate the closure assumption of genetic programming. Some of the function set cannot be argument to another function in F, given that both Boolean and real functions are found in that set. The restrictions means that the real functions are always in the lower part of the tree structures; this makes sense as all the trading decisions are made based on technical indicators; while Boolean functions and relational operators are found on the upper part of the tree.

B. Fitness Evaluation Criteria

In most of the relevant studies, fitness values are calculated using the excess return over the markets. However, Net Profit is a much more simple and efficient way to evaluate the profitability of trading strategies. The total net profit is the total profit (gross profit minus gross loss) for all closed trades, both wins and losses, in the trading period.

C. Initialization of the Base Population

There are two methods to generate the initial population of the trading blocks. The Full method creates tree structures such that the length of every path between a terminal and the root is equal to a predefined depth, and the grow method create the tree structures with variable shapes. A combination of the Full and Grow methods is used in this study. Due to the particular structure of function set, the recursive construction must follow the rules bellow. Firstly, the root of the tree must be selected from Boolean functions and operators, (buy and sell signal must be generated at the top of the tree structure). Secondly, the tree root's descendants can be selected among Boolean constants, Boolean functions, Boolean or Boolean operators. Thirdly, a relational operator's descendants must be selected among real functions or terminals.

After the generation of the initial population, more complex trading decisions are built using the initial trading block with some mutations and crossover.

D. Crossover, Mutation, Reproduction and Selection

As the initialized population usually has low fitness, genetic operators must be utilized in the searching of better program.

The crossover operator is used to combine the genetic material of two parents by swapping a part of one parent with a part of the other. In our study, the offspring resulted by crossover replaces the parent with less fitness.

The mutation operator is use to operate on only one individual. When one of the individuals in a particular population has been selected for mutation, our program selects a point the tree randomly and replaces the existing sub-tree with a randomly created sub-tree. The mutated individual is then put back to the original population to calculate the fitness.

The reproduction operator is simply a clone of the current tree structure, this usually happens to a tree with rather high fitness value. This operator is controlled by the parameter reproduction probability.

In the selection of programs in the population, a rank based method proposed by Baker (1985) and Whitley (1981) is used. In order to use the ranking system, the fitness of all programs must be calculated and sorted from the best (rank 1) to the worst (rank *n*). A new value f_i is then associated with the program or rank *i* as follows:

$$f_i = Max - [(Max - min)\frac{i-1}{N-1}] \quad (2)$$

In this way, the best program gets fitness Max, and the worst program gets fitness Min, and most importantly, the remaining programs are equally spaces between Min and Max. The "super program problem" that crowded out all the other programs can be avoided.

IV. Empirical Experiment

In this empirical analysis, 21 days of Hushen 300 index future data is used, each containing 270 data points. The original data points cover the time period from 29/09/2014 up to 03/11/2014. In each trading day, the first 130 data points are used to train the model, and the last 140 data points will be used to test the robustness of the model. As the morning session of a trading day consists of 130 minutes, the training processes were basically conducted in the morning session and the out of sample testing process were conducted in the afternoon session. To overcome the over fitting problem, more weight is putting the out of sample performance the developed model. Only the out of sample performance will be discussed as in sample trading profit is highly unrealizable.

A. Parameter Settings

Basic settings for the trading Symbol are shown in Table I.

TABLE I: SYMBOL SETTINGS

Bar Type	Intraday
Bar Size	1
Session Start time	09:16:00
Session End time	15:15:00
Point Value	300
Bid/Ask Spread	0.1
Trading cost per trade	50

Excessive preliminary experiments were conducted to optimize the parameter settings for the genetic training process. The parameter values in Table II below were finally selected.

TABLE II: PARAMETER SETTINGS

Population size	300
Number of generations	30
Crossover percentage	60%
Mutation percentage	50%
Tree depth	5
Tournament size	5
Limit of entries per day	8
Wait for exit before entering new trade	Yes
Max bars looking back for Indicators	30

The 21 days of trading results without holding assets overnight is shown is Table III.

The genetic programming trainings were conducted using the software Adaptrade Builder and the out of sample testing processes were running on Multicharts. As shown in Table III, the out of sample performance of the trained models varies during different trading days. The positive value represents gains and the negative value represents losses. The sum of the total trading profits and losses for the training method is 2460 for the training using the intraday data points.

As the models were trained every day in the morning session, different trading strategies were obtained using the genetic programming methods. Based on the trading results showed in Table III, although the win rate is only 0.33, some of the models were able to capture the main trend of the trading in a certain day to accumulate profits.

The average daily return is 117.14, and the variance is quite big as 6476571, which make the Sharpe ratio quite small, however, despite of the large max drawdown on some of the trading days, profits can be accumulated in a long run, although more experiments are needed to make this point more valid.

However, when we use the first 14 days data to train the model and the last 7 days' data to test the out of sample performance of the trained model, a severe loss of 37320 was reported. This result proved our assumption that trading pattern tends stay stable during a single trading day and varies during multiple days.

TABLE III: TRADING RESULTS FOR INTRADAY TRAINING METHOD

Date	Out of sample Profit/Loss of model trained with original data
2014-9-29	-960
2014-9-30	-1140
2014-10-08	2100
2014-10-09	480
2014-10-10	6900
2014-10-13	-840
2014-10-14	780
2014-10-15	2100
2014-10-16	-2640
2014-10-17	2640
2014-10-20	960
2014-10-21	0
2014-10-22	-5640
2014-10-23	540
2014-10-24	0
2014-10-27	-540
2014-10-28	-120
2014-10-29	-1740
2014-10-30	-480
2014-10-31	3120
2014-11-03	-3060

V. CONCLUSION

In this paper, an application of the genetic programming and wavelet de-noise were utilized in the generating of intraday programmed trading rules. The results showed that the trading rules generated by genetic programming are able to accumulate trading profits within a certain period. The trading strategies trained with morning session data are most likely to be profitable in the afternoon session of the same trading day, however, the trading strategies trained with multi-days of data can lead to severe losses in the following several trading days, which indicates that trading trend tend to vary in different trading days and stay the same in a single trading day.

One of the drawbacks of this study is that the high frequency data points utilized in this study can be noisy and non-stationary, technical analysis may suffer from such kind of characteristics, some filters like wavelet can be used to the process the original data points to make it more suitable for technical analysis.

ACKNOWLEDGMENT

The work described in this paper was partially supported by two grants from The Hong Kong Polytechnic University, China (Project No. G-RTK6 and 4-BCBJ).

REFERENCES

[1] R. Gencay, "The predictability of security returns with simple technical trading rules," *Journal of Empirical Finance*, vol. 5, pp. 247-359, 1998.

[2] F. Fernańdez-Rodr´ıguez *et al*., "On the profitability of technical trading rules based on artificial neural networks: Evidence from the Madrid stock market," *Economics Letters*, vol. 69, pp. 89-94, 2000.

[3] T. Chavarnakul and D. Enke, "A hybrid stock trading system for intelligent technical analysis-based equivolume charting," *Neurocomputing*, vol. 72, no. 16-18, pp. 3517-3528, 2009.

[4] Alexander, "Price movements in speculative markets: Trends or random walks," in *The Random Character of Stock Market Prices*, P. Cootner, Ed. 2MIT Press, Cambridge, 1964, pp. 338–372.

[5] E. F. Fama, "Efficient capital markets: A review of theory and empirical work," *The Journal of Finance*, vol. 25, no. 2, pp. 383-417, 1970.

[6] S. W. Pruitt and R. E. White, "The CRISMA trading system: Who says technical analysis can't beat the market?" pp. 55-58, 1988.

[7] T. P. Mark and H. Allen, "The use of technical analysis in the foreign exchange market," *Journal of International Money and Finance*, vol. 11, pp. 304-314, 1992.

[8] J. H. Holland, "Outline for a logical theory of adaptive systems," *Journal of the Association for Computing Machinery*, vol. 3, pp. 297-314, 1962.

[9] R. J. Bauer, *Genetic Algorithms and Investment Strategies*, John Wily & Sons. Inc, New York, 1994.

[10] J. Y. Potvin and S. Bengio, "The vehicle routing problem with time windows — Part II: Genetic search," *INFORMS Journal on Computing*, vol. 8, pp. 165-172, 1996.

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

[12] F. Allen and R. Karjalainen, "Using genetic algorithms to 3nd technical trading rules," *Journal of Financial Economics*, vol. 51, pp. 245–271, 1999.

[13] S.-H. Chen and C.-H. Yeh, "Toward a computable approach to the efficient market hypothesis: an application of genetic programming," *Journal of Economic Dynamics & Control*, vol. 21, pp. 1043–1063, 1996.

[14] C. Neely, P. Weller, and R. Dittmar, "Is technical analysis in the foreign exchange market profitable? a genetic programming approach," *Journal of Financial and Quantitative Analysis*, vol. 32, pp. 405–426, 1997.

[15] C. Neely and P. Weller, "Technical trading rules in the European monetary system," *Journal of International Money and Finance*, vol. 18, pp. 429–458, 1999.

[16] J. P. Marney, H. Tarbert, and C. Fyfe, "Technical trading versus market efficiency—a genetic programming approach," *Computing in Economics and Finance, Society for Computational Economics*, Barcelona, Spain, July 2000.

[17] J. Y. Pottvin, P. Soriano, and M. Vallee, "Generating trading rules on the stock markets with genetic programming," *Computers & Operations Research*, vol. 21, pp. 1033-1047, 2004.

[18] Mallick *et al*., "An empirical study of Genetic Programming generated trading rules in computerized stock trading service system," in *Proc. International Conference on Service Systems and Service Management*, 2008.

[19] Esfahanipour *et al*., "The profitability of technical trading rules in Tehran stock exchange: An application of genetic programming," in *Proc. International Symposium on Innovations in Intelligent Systems and Applications*, 2009.

[20] J. Strabburg, G.-M. Christian, and V. Alexandrov, "Parallel genetic algorithms for stock market trading rules," in *Proc. International Conference on Computational Science*, 2012.

[21] C. Neely and P. Weller, "Technical trading rules in the European monetary system," *Journal of International Money and Finance*, vol. 18, pp. 429–458, 1999.

[22] A. Kablan and J. Falzon, "The use of dynamically optimized high frequency moving average strategies for intraday trading," *Word Academy of Science, Engineering and Technology*, 2012.

Hongguang Liu is now a PhD candidate of the Hong Kong Polytechnic University. His research interest is financial time series analysis and forecasting. He has been working on the application of wavelet transformation into the building of automatic trading systems during his PhD study.

Ji Ping received his PhD degree from West Virginia University, USA. His research interests are industrial engineering including supply chain management, operations management. He has published over 100 papers in various journals including IIE Transactions, IEEE Transactions on Robotics and Automation, European Journal of Operational Research, and Bioinformatics.