ISSN 2281-4299

DIPARTIMENTO DI INGEGNERIA INFORMATICA AUTOMATICA E GESTIONALE ANTONIO RUBERTI

SAPIENZA
UNIVERSITÀ DI ROMA

Optimal Step-wise Parameter Optimization of a FOREX Trading Strategy

A. De Santis
U. Dellepiane
S. Lucidi
S. Renzi

Technical Report n. 6, 2014

April 24, 2014

Optimal Step-wise Parameter Optimization of a FOREX Trading Strategy

A. De Santis, U. Dellepiane, S. Lucidi, S. Renzi

Dipartimento di Informatica e Sistemistica
Sapienza Università di Roma
Via Ariosto, 25 - 00185 Roma - Italy

e-mail: desantis@dis.uniroma1.it
e-mail: dellepiane@dis.uniroma1.it
e-mail: lucidi@dis.uniroma1.it
e-mail: renzi@dis.uniroma1.it

Abstract.
The goal of trading simply consists in gaining profit by buying/selling a security: the difference between the entry and the exit price in a position determines the profit or loss of that trade. A trading strategy is used to identify proper conditions to trade a security. The role of optimization consists in finding the best conditions to start a trading maximizing the profit. In this general scenario, the strategy is trained on a chosen batch of data (training set) and applied on the next batch of data (trading set). Given a strategy, there are different issues to deal with, to obtain the best performances from the optimization. First of all, among all the parameters that define the strategy, it is important to identify and select the most relevant ones that become the optimization problem variables. In this way the problem complexity is reduced and the overfitting on the training set is avoided. Once the variables are chosen, the focus is on the time period used for the training and the trading sets. Accordingly, for any parameter, a proper box constraint is fixed taking into account the frequency of the given trading strategy (time scale, reactivity, etc.). Since the objective function is not defined in closed form but through an algorithm, the problem lies within the framework of black-box optimization.

Keywords. Global Optimization, Black-Box, Forex Trading, Algorithmic Trading, Modelling Procedure

1 Introduction

When trading a security the goal is to achieve a profit by buying/selling the particular security at different price levels. This kind of profit is strictly affected by the behavior of the security price over time. There exist two kinds of trading: the exchange trading and the over-the-counter or off-exchange trading. The first one is characterized by a supervision of an exchange, which is defined as an highly organized market. In contrast, the off-exchange is defined as the trading done directly between two parties, without any supervision. The intraday price movements resume all the information available during the day which can affect all the tradable securities. On the other hand, the trader can take her decisions by the suggestions which receives as outputs of either the technical or the fundamental analysis. When considering the technical analysis, the trader focuses her attention in the price time series features more than on the economical variables related to the price value, which define the fundamental analysis. In this paper we consider mostly the exchange trading and the related technical analysis.

Before introducing the framework in which we decide to operate, we give the reader some basic notions about trading. Generally speaking, we use the terms *buy* or *sell* to define the actions of a trader, but this can create a sort of confusion. Once a trader enters the market, it is said that she has *opened a position*; in these terms, the position defines the state of the trader. The position can be *long* or *short* according to the first action made by the trader. In particular, if she buys a security first then she has opened a long position, otherwise if she sells the security first she is opening a short position. If the trader opened a long position, in order to exit the market, she has to close the position, which means she has to sell the security bought. Obviously, she gains profit if during the time she has hold the position, the security price has increased. On the other hand, if the trader opened a short position, in order to close it, she has to buy the security. For this reason, she's making profit only if the price of the security has decreased. Until now we have talked about opening and closing a position, but how a trader can choose the right moment of doing it? Both, the opening and the closing criteria strictly depend on the trading strategy selected. Regarding the closing side, it is possible to distinguish two different logical exit criteria: the stop and the target. The first one is used to define the price level at which the trader should close her position in order to limit the losses, while the second one is used to define the price level at which the trader should close her position in order to gain profits. The price level at which a position is opened is said *entry price*, being the *exit price* the value at which a position is closed.

Nowadays a wide portion of the trading is performed by the so called algorithmic trading systems, which consist in the use of electronic platforms for entering trading orders with a procedure which executes pre-programmed trading instructions. The technical designs of such systems are not standardized. Conceptually, the design can be divided into logical units: 1) the data stream unit, 2) the decision or strategy unit and 3) the execution unit. The data stream unit is the part of the system which receives and elaborates the data input directly from the market. The decision or strategy unit, which is the focus of this work, is a fixed plan that is designed to achieve a profitable return. At last, the execution unit has the purpose of applying the buying/selling orders. A trading strategy can be viewed as a set of rules defined by some actions and some indicators. For example, we can refer to an action as "buy the security" while the indicators defined the conditions at which the action takes place. Typically, an action is performed after comparing the indicator with a threshold value. Generally, these indicators are functions of the price values up to the current time and are characterized by some parameters. Such parameters are typically threshold values, time windows width (for example the number of periods used in the calculation of a Simple Moving Average) or weights.

Generally, a trader, in order to use this kind of trading, initially has to make some decisions: firstly, she selects the security, then chooses a strategy and fixes all its parameters according to her experience or beliefs. After this, she start trading (crossing her fingers) waiting until the end of the period of trading. This kind of procedure can be efficient if we consider a security that has a slow dynamic and the parameters selected have suitable values. If, instead, we consider a security with a higher level of volatility, this kind of procedure may not guarantee the best results. This means that, in this context, this procedure can be significantly improved. This is, indeed, the main goal of this paper. Therefore, once the strategy is chosen, an optimization procedure can be designed in order to select the features which give the best performances.

The standard procedure is applied during the whole trading period with the parameter values fixed at its beginning, while in the proposed procedure the trading period is partitioned in a number of adjacent subintervals and these parameters are changed accordingly. The first subinterval is used in order to define the best parameter values which then are used in the second subinterval to apply the strategy. At the end of the second subinterval, the whole procedure is repeated, using this subinterval as a training period and the follower as the trading one, and so on. The main idea underlying this procedure relies in the market resistance assumption. This means that the features which define the price behavior are supposed to be mostly unchanged during two adjacent time periods. For this reason, it is obvious that the period length becomes an important feature to consider during the modelization of the problem and is strictly related to the security considered.

Considering the market resistance assumption introduced, this procedure differs from the others proposed in the past. The latters simply try to identify the best model from the price time series to forecast its future values. Thus, these forecasts are presumably used to make decisions about trading at the current time. The proposed procedure also uses the information within the data but the focus this time is not on the forecast but on learning. After the procedure is fixed, we try to use it at the best of its performances, by adapting the parameter values over time. Obviously this change takes place only at the end of a subinterval.

The purpose of this work is to address all the issues related to the definition of the optimization problem, which needs to be solved in order to find the optimal parameter values for each training period.

2 I Step - Definition of the security

The first feature to select in order to apply any trading strategy is the security the trader wants to trade. As mentioned before, algorithmic trading is more suitable when used in trading an high volatility security. It is known that the Foreign-exchange market is by far the most volatile one. Foreign-exchange trading (Forex) dates back to ancient times. The foreign-exchange markets underpin all other financial markets and collectively they form the larger financial market by far. Hundred of thousands of foreign-exchange transactions occur every day, with an average turnover totalling 1.9 trillion a day [1]. Among all the currency pairs exchanged in this market, the Euro-U.S. Dollar (EUR/USD) is the most volatile and in our experiments we decided to consider this security. The reasons why we selected this market and in particular the EUR/USD as security can be summarize as follows:

- The Forex Market operates theoretically 24 hours per day, everyday. In particular there are some day periods where the market exchange volumes (number of trades) are very low.

It is anyway, as compared to the other markets, the most active;

- The EUR/USD is the most liquid security exchanged in a trading market. This ensures some features that simplify the trading procedure. In particular there is no slippage, which means that when an order is placed, the trader is sure to buy all the desired amount at the price proposed.

Obviously, choosing this kind of security may stress too much the market resistance assumption but on the other hand gives us the opportunity to show the potentialities of the proposed procedure.

3 II Step - Definition of the strategy

Since a strategy is strictly dependant on the particular security traded, the selection of the trading strategy is then subject to the selection of the security. For our work, we selected a strategy proposed by J. Welles Wilder Jr in [citazione libro wilder].

3.1 Basic Notions

The basic price information are summarize in time bars, which consist in pre-defined time periods characterized by the following values:

- the Open price value($Op(i)$) , which is the first price exchanged in the time bar period;
- the Close price value ($Cl(i)$), which is the last price exchanged in the time bar period;
- the High price value ($Hi(i)$), which is the highest value assumed by the price security during the time bar period;
- the Low price value ($Low(i)$), which is the lowest value assumed by the price security during the time bar period.

Since we are considering time bar period fixed to one minute, we can assume that the $Op(i)$ is always equal to $Cl(i-1)$ and for this reason we are only interested in the Close, High and Low price values.

We indicate with $\bar{x}(i)$ the average price, at the *i-th* time_bar given by:

$$\bar{x}(i) = [Cl(i) + Hi(i) + Low(i)]/3 \tag{1}$$

and with TR(i) the True Range of the price, given by:

$$TR(i) = max\{Hi(i) - Low(i), |Hi(i) - Cl(i-1)|, |Low(i) - Cl(i-1)|\} \tag{2}$$

Now we can define the stop and target threshold for a long position:

$$STOP_L(i) = \bar{x}(i-1) - TR(i-1) \tag{3}$$

$$TARGET_L(i) = 2 * \bar{x}(i-1) - Low(i-1) \tag{4}$$

While, for a short position, the same values are given by:

$$STOP_S(i) = \bar{x}(i-1) + TR(i-1) \tag{5}$$

$$TARGET_S(i) = 2 * \bar{x}(i-1) - Hi(i-1) \tag{6}$$

This strategy is improved by the possibility of *reversing the position*, that is switching the current opened position at the opposite. This means that both the conditions to hold the position opened fail and the ones to open the opposite position are satisfied. The strategy can be described as follows:

The Momentum Factor Strategy

We define Momentum Factor as the difference between the close price today and the close price two days ago: MF(i) = Cl(i) - Cl(i-2)

We define STOP as a protective value in case of an extreme move opposite to our guess

We define TARGET as the value we accept to close a position taking profit

If no Positions Opened then

if Cl(i) > Cl(i-2) + min {MF(i-1),MF(i-2)} then

Open a Long Position

if Cl(i) < Cl(i-2) + max {MF(i-1),MF(i-2)} then

Open a Short Position

If a Long Position Opened then

if Cl(i) ≤ Cl(i-2) + min {MF(i-1),MF(i-2)} then

Reverse to a Short Position

if Cl(i) ≤ $STOP_L$ or Cl(i) ≥ $TARGET_L$ then

Close the Position

If a Short Position Opened then

if Cl(i) ≥ Cl(i-2) + max {MF(i-1),MF(i-2)}then

Reverse to a Long Position

if Cl(i) ≥ $STOP_S$ or Cl(i) ≤ $TARGET_S$ then

Close the Position

4 III Step - Definition of the problem

In order to apply the proposed procedure, we need to formulate an optimization problem in the form of:

$$\begin{aligned} &Max_x \quad f(x) \\ &\text{s.t.} \\ &x \in X \end{aligned} \tag{7}$$

4.1 Definition of the objective function

First we need to define the object of the maximization procedure, namely the objective function of our problem. In a single transaction, the profit/loss value obtained, if a long position is closed, is given by:

$$\Pi_L = exit\_price - entry\_price$$

i.e. a trader sells a security (at the *exit_price*) previously bought (at the *entry_price*) having guessed the price would increase. On the other hand, if a short position is closed, the same profit/loss value is given by:

$$\Pi_S = entry\_price - exit\_price$$

i.e. a trader buys a security previously sold having guessed the price would decrease.

Since the trader is interested in obtaining the greatest profit available over a fixed period of time, the objective function is simply given by the sum of all the single profits or losses achieved in every closed transaction. This function can be written as follow:

$$\Pi = \sum_{i \in [1,T]} \Pi_i$$

This objective function is not available in an analytical form since its values are defined only after a transaction is closed. Furthermore, we do not know both the entry point and the exit point of a transaction because this depends on the data and on the trading strategy. We consider this function as a black-box and for this reason, in order to solve the optimization problem we are defining, we recour to derivative free algorithm.

Once the objective function is defined, we need to identify the variables of the problem, which is the topic of the following section.

4.2 Definition of the variables

Considering the procedure described in 3.1, we identified the following seven variables:

- x_1: considering the definition of the Momentum Factor, MF(i) = Cl(i) - Cl(i-2), we can introduce a variable instead of the fixed lag of 2 previous periods.
  Therefore, MF(i) = Cl(i) - Cl(i-x_1)

- x_2, x_3, x_4 represent the weights associated respectively to Cl(i), Hi(i) and Low(i) in (1). We can rewrite $\bar{x}$ as a weighted average in the following way:

$$\bar{x}_w(i) = [x_2 * Cl(i) + x_3 * Hi(i) + x_4 * Low(i)]/(x_2 + x_3 + x_4) \quad (8)$$

- x_5 represents the number of previous periods considered in the right side part of the conditions that establish the opening or the reverse for a position. Thus, we can write:

$$\begin{aligned} \{Cl(i) &> Cl(i-2) + min\{MF(i-1), MF(i-2)\} \rightarrow \\ \{Cl(i) &> Cl(i-x_5) + min\{MF(i-x_5-1), MF(i-x_5-2)\} \end{aligned} \quad (9)$$

$$\begin{aligned} \{Cl(i) &< Cl(i-2) + max\{MF(i-1), MF(i-2)\} \rightarrow \\ \{Cl(i) &< Cl(i-x_5) + max\{MF(i-x_5-1), MF(i-x_5-2)\} \end{aligned} \quad (10)$$

$$\begin{aligned}\{Cl(i) &\leq Cl(i-2) + min\{MF(i-1), MF(i-2)\} \rightarrow \\ \{Cl(i) &\leq Cl(i-x_5) + min\{MF(i-x_5+1), MF(i-x_5)\}\end{aligned} \tag{11}$$

$$\begin{aligned}\{Cl(i) &\geq Cl(i-2) + max\{MF(i-1), MF(i-2)\} \rightarrow \\ \{Cl(i) &\geq Cl(i-x_5) + max\{MF(i-x_5+1), MF(i-x_5)\}\end{aligned} \tag{12}$$

- x_6 represent the number of previous periods used in the calculation of (3) and (5). We obtain the following equations:

$$STOP_L(i) = \bar{x}_w(i-1-x_6) - TR(i-1-x_6) \tag{13}$$

$$STOP_S(i) = \bar{x}_w(i-1-x_6) + TR(i-1-x_6) \tag{14}$$

- x_7 represent the number of previous periods used in the calculation of (4) and (6). We obtain the following equations:

$$TARGET_L(i) = 2 * \bar{x}_w(i-1-x_7) - Low(i-1-x_7) \tag{15}$$

$$TARGET_S(i) = 2 * \bar{x}_w(i-1-x_7) - Hi(i-1-x_7) \tag{16}$$

4.3 Definition of the feasible set

Since the variables of the problem cannot vary on all $\Re^7$, we have constrained each of them to belong to an interval, namely:

$$x_i \in [lower\_bound_i, upper\_bound_i] \quad for i = 1, \cdots, 7 \tag{17}$$

While the lower bound of each variable is fixed to the literature values, the upper bounds are defined formally in Section 7.

5 IV Step - Selection of the variables

It is possible that, among the variables identified, some of them are not important or that if we let all of them free to vary on the domain the strategy performance will decrease. If this is not the case, it will result that all the variables identified have to be considered as significant. In order to identify if some variables are useless or create a sort of noise in the optimization procedure together with the others, we arranged a series of 14 Tests. For each Test, a subgroup of variables has been bounded on a certain domain while the others have been fixed to the literature values, namely to their lower bound. The experiments are implemented in Fortan 90 and considered the same data series of the EUR/USD. For the optimization procedure, we consider a Simulated Annealing strategy in order to explore the feasible set and the Derivative Free Linesearch algorithm, presented in (citazione articolo DFL) as a local optimization algorithm. First, the analysis of the performances achieved by each Test has been evaluated according to the profit obtained on the trading period, since this value is the most important from a practical point of view. The results are reported in Table 1. It is clear that the optimized procedure perform

Test	Free Variables	Profit
Literature Value	–	0.14023
Test 1	x_1, x_2, x_3, x_4	0.20076
Test 2	x_2, x_3, x_4, x_5	0.20037
Test 3	x_2, x_3, x_4, x_6	0.33803
Test 4	x_2, x_3, x_4, x_7	0.19083
Test 5	x_1, x_2, x_3, x_4, x_5	0.18781
Test 6	x_1, x_2, x_3, x_4, x_6	0.33435
Test 7	x_1, x_2, x_3, x_4, x_7	0.19372
Test 8	x_2, x_3, x_4, x_5, x_6	0.3416
Test 9	x_2, x_3, x_4, x_5, x_7	0.18596
Test 10	x_2, x_3, x_4, x_6, x_7	0.33824
Test 11	$x_1, x_2, x_3, x_4, x_5, x_6$	0.33813
Test 12	$x_1, x_2, x_3, x_4, x_5, x_7$	0.17846
Test 13	$x_2, x_3, x_4, x_5, x_6, x_7$	0.34314
Test 14	$x_1, x_2, x_3, x_4, x_5, x_6, x_7$	0.33613

Table 1: Comparison between the profit obtained on the trading periods by the Literature and the different Tests

better than the literature one, which represents a standard algorithmic trading approach. Then, it is possible to notice that a subgroup of Tests perform significantly better than the others. These Tests are considered as the best performing while the others are discarded. We can finally consider that each of these Tests let free to vary on the domain the variables x_2, x_3, x_4 and x_6. Since we are interested in identifying just one optimization problem, the analysis on the results is widened. First we focused our attention on the performances archived by the selected Tests on the training period and on the structure of the Test itself. The results concerning this analysis are reported in Table 2. In particular Test 8 is the best performing one, followed by Test 6 and Test 10. Regarding the feasible set structure of each Test, it is possible to notice that Test 8, 6 and 10 share a subgroup of variables while the other are different. Since we are interested in a formulation of the problem that could considered suitable in a general case, we cannot select which of these three tests is the best one. Furthermore, we notice that among the selected Tests, Test 11, 13 and 14 have a feasible set that contain the feasible set of Test 8. Further observations regarding the coordinates of the optimal points within the domains of the latter Tests, led us to conclude that Test 8 is the one which model better the problem. In fact, the optimal point found for Test 11 and 13 lies on the feasible set of Test 8. A different argument is used to discard Test 14: among the 7 Tests selected, it is the worst one and this can be explained by the fact that letting free all the variables may cause a sort of noise which damages the optimization performance. Indeed, selecting the smallest subgroup of variables makes the calculations faster and creates some advantages for the optimization procedure in exploring the feasible region.

In order to complete the modelization of the optimization problem we need to fix the upper bound on each variable and the time period length for each training/trading subinterval. Since two of these variables represent time units, first we need to select the time period length and only later we can allow to define properly the feasible set. For this reason, the following section is concerned in the definition of the most suitable time period length and the next one studies the definition of the domain, which will complete the optimization problem.

Test	Free Variables	Profit
Test 3	x_2, x_3, x_4, x_6	0.41794
Test 6	x_1, x_2, x_3, x_4, x_6	0.41731
Test 8	x_2, x_3, x_4, x_5, x_6	0.41826
Test 10	x_2, x_3, x_4, x_6, x_7	0.41692
Test 11	$x_1, x_2, x_3, x_4, x_5, x_6$	0.41523
Test 13	$x_2, x_3, x_4, x_5, x_6, x_7$	0.41465
Test 14	$x_1, x_2, x_3, x_4, x_5, x_6, x_7$	0.41139

Table 2: Comparison between the profit obtained on the training periods by the best performing Tests

Time Period	Profit
6 Hours	0.3416
12 Hours	0.74802
18 Hours	1.14398
24 Hours	1.50195

Table 3: Profit obtained by the procedure on trading periods with different length

6 V Step - Definition of the subinterval time width

In this section we concentrate on the selection of the subinterval time width. To this aim, we are interested in a time window which satisfies simultaneously two opposite criteria:

1. the time window length should not be too long in order to not to stress too much the market resistance assumption;
2. the time window length should be long enough to reduce the number of subinterval and then the number of optimization procedures used during the trading.

Obviously, this kind of selection can be made only empirically, since it is not possible a priori to define the best time window length. For this reason, we selected four different lengths (6, 12, 18, 24 hours) and then considered the results obtained by the procedure, which are reported in Table 3.

First of all, it is important to notice that greater is the time window length considered, higher is the profit achieved, which is not obvious. In particular, this results strictly depends on the likelihood of the market resistance assumption made. For this reason, the best performance are achieved when the time windows are equal to 24 hours. Although this result, it is also important to notice that a trading period of 24 hours is larger than a day, due to the fact that we are not considering in the subinterval the periods of time where there are no transactions made, i.e. the volume exchanged is zero. We noticed, then, 18 hours are the nearest time length to a trading day period, and for this reason we chose it as the time period length for our problem.

7 VI Step - Definition of the feasible set

As mentioned in Section 4.3, the feasible set of the problem is defined by box constraints. First of all, the attention is reported on the definition of the lower bound of each variable. In particular, it is fixed to the literature value, which is not a common procedure in optimization theory. However, this was a decision we have to take during the modelling of the problem for several reasons. It is trivial to notice that, since the parameters are time units, they cannot assume negative values. Furthermore, the lower value they can assume is equal 0. This means that the particular indicator is not defined and therefore the strategy would change. We underline that, if we consider the three weights of the weighted average, if the values are all equal 0, we have an undefined formula, which we want to avoid. Then the definition of the feasible set is just related to the correct definition of the upper bounds on the variables.

For the tests made before we fixed each upper bound to a value which we considered reasonable and suitable for the strategy proposed. At this point, we decide to consider if an improvement on the performance could be achieved changing these values. In particular, we defined several feasible sets by increasing or decreasing the initial upper bounds. The results achieved show that:

- when the upper bounds are decreased, the performances of the proposed procedure worsened and this meant that a better solution, that relied in the dropped feasible set portion, was missed;
- on the other hand, when the upper bounds are increased, the larger the feasible set, the better the solutions we found.

The procedure used in the definition of the feasible set was considering the set used in the first experiments as middle point and then increase and decrease by the same amounts each upper bounds. Studying the optimal point coordinates obtained in the larger feasible set, we found out that they almost always lied in the relative interior and for this reason we assumed that it was not necessary to expand it further.

8 Conclusion

The focus of this work was to study the optimization problem that arises from the application of algorithmic trading. We studied every issues that need to be solved in order to formulate correctly the optimization problem that has to be solved to apply the trading procedure. First of all we selected the security we want to trade, which represent the object of the procedure. The choice to test the procedure was the EUR/USD, which represent the most volatile security exchanged nowadays. Then the momentum strategy has been outlined, and in this phase we identify the parameters on which it depends. We considered first all of them (in the number of 7) and then we studied the structure of the problem in order to identify the most significant ones. This, obviously was subject to the definition of an objective function of the problem, which we identify in the maximization of the profit over a fixed period of time. Then we defined a feasible set of the problem in order to start our tests. The formal definition of the latter was then made after the subinterval time width was studied. In the end, the output of the whole procedure is the following optimization problem:

$$
\begin{aligned}
&Max_x \quad \sum_{t \in T} Profit(\pi, x) \\
&\text{s.t.} \\
&x_1 : 1 \leq x_1 \leq 30 \\
&x_2 : 1 \leq x_2 \leq 30 \\
&x_3 : 1 \leq x_3 \leq 30 \\
&x_4 : 1 \leq x_4 \leq 30 \\
&x_5 : 1 \leq x_5 \leq 40 \\
&T = [0, 1080min]
\end{aligned}
$$

By applying this procedure, the profits indicated in Table 1 are achieved.

At this point we have a formulation of the problem and the next logical step, which is not the object of this work, is to select the best optimization method to solve it, considering both the quality of the solutions found and the computational effort required in doing so. Further possible developments of this study may include to test this procedure on other time series, other security and on other trading strategy. In this way, we can identify which of the hypothesis made strictly depend on the choices made through this work and which can be used in order to generalize the modelling procedure.

References

[1] Marc Levinson Guide to Financial Markets, Fourth Edition, The Economis

[2] R.K. Dase and D.D. Pawar *Application of Artificial Neural Network for stock market predictions: A review of literature.* SInternational Journal of Machine Intelligence, ISSN: 09752927, Volume 2, Issue 2, 2010, pp-14-17.

[3] R.F. Engle. *Autoregressive conditional heteroscedasticity with estimates ofthe variance of United Kingdom in?ation* Econometrica 50 (1982) 9871008

[4] T. Bollerslev. *Generalized autoregressive conditional heteroscedasticity.* Journal of Econometrics 31 (1986) 307 327

[5] M.H.F. Zarandi, B. Rezaee, I.B. Turksen and E. Neshat. *A type-2 fuzzy rule based expert systemmodelfor stockpriceanalysis.* ExpertSystemswithApplications 36 (2009) 139154.

[6] C.F. Liu, C.Y. Yeh and S.J. Lee. *Application oftype-2 neuro-fuzzy modeling in stock price prediction.* Applied Soft Computing 12 (2012) 13481358.

[7] P.F. Pai and C.S. Lin. *A hybrid ARIMA and support vector machines model in stock price forecasting.* Omega:The InternationalJournalofManagement Science33 (2005) 497505.

[8] G. Valeriy and B. Supriya. *TSupport vector machine as an ef?cient framework for stock market volatility forecasting.* Computational Management Science 3 (2006) 147160.

[9] J. Nenortaite and R. Simutis. *Stocks' Trading System Based on the Particle Swarm Optimization Algorithm.* Computational Science - ICCS 2004, Lecture Notes in Computer Science Volume 3039, 2004, pp 843-850

[10] P.B. Myszkowski and A. Bicz. *Evolutionary Algorithm in Forex Trade Strategy Generation.* OProceedings of the International Multiconferenceon Computer Science and Inromation Technology pp. 81-88.

[11] C. Neely, P. Weeler and R. Dittmar. *Is Techincal Analysis in the Foreing Exchange Market Profitable? A Genetic Programming Approach .* Journal of Financial and Quantitative Analysis 32(4), pp. 405-426, 1997.

[12] L. Lin, L. Cao, J. Wang and C. Zhang. *The applications of Genetic Algorithms in Stock Market Data Mining Optimization* Zanasi, A., Ebecken, N.F.F. (eds.) Data Mining V. WIT Press (2004)

[13] Y. Chen, S. Mabu, K. Shimada and K. Hirasawa. *A Genetic Network Programming With Learning Approach for Enhanced Stock Trading Model*, Expert System with App., 36(10), pp. 12537-125466, 2009