Building an Efficient Evolutionary Algorithm for Forex Market Predictions

Rafal Moscinski[1] and Danuta Zakrzewska[1]

Institute of Information Technology Lodz University of Technology, Lodz, Poland
moscinski.rafal@gmail.com,danuta.zakrzewska@p.lodz.pl

Abstract. Foreign Exchange Market is one of the biggest financial markets in the world. In the paper an efficient algorithm for generating profitable strategies on this market is presented. The proposed technique is based on an evolutionary algorithm and uses the combination of technical indicators, which are to enable obtaining the highest profit depending on training and testing data. The algorithm allows to avoid risky strategies and is enhanced in developed mutation and crossover operators for assuring the effectiveness of obtained Forex trade strategies. The performance of the proposed technique was verified by experiments conducted on real data sets.
Keywords: evolutionary algorithm · trend predictions · Forex market

1 Introduction

Financial market predictions have become an object of scientific studies for several years. Foreign Exchange Market is mentioned among the biggest ones in the world. It is the market in which participants are able to buy, sell, exchange and speculate on currencies [1]. In the paper an algorithm which aims at generating profitable strategies for this market is proposed. The presented technique is based on an evolutionary algorithm which uses closing prices of specified time interval and computes them together with technical analysis indicators. The goal of the algorithm is to create such combination of various technical indicators which generates the highest profit on selected training and testing data. Additionally the algorithm detects strategies which generate too high loss during training process and automatically rejects them. Similarly to [2], as a data structure, a pair of decision binary trees is used, what allows to choose effectively combination of technical indicators. However, in the presented approach evolutionary technique has been enhanced in several improvements including input parameters optimization as well as developed mutation and crossover operators, which enable generating new solutions effectively. The performance of the proposed technique has been verified on the real data sets. Experiments showed effectiveness of the presented algorithm in indicating profitable Forex trade strategies.

The remainder of the paper is organized as follows. In the next section related work concerning application of evolutionary and genetic algorithms as well as artificial intelligence methods in Forex trade prediction is described. In the following section the proposed methodology is depicted with details. Section 4 is devoted to experiments and their results. Finally concluding remarks and future research are presented.

2 Related Work

Many researchers considered application of computational intelligence methods for financial market predictions. Research concerning Forex trade market has taken the significant place in the investigations. As the most commonly used methods one should mention artificial neural networks, genetic programming and evolutionary algorithms. The first method has been broadly investigated by many authors. Ni and Yin considered a hybrid model consisting of various regressive neural networks [3]. Lee and Wong combined an adaptive ANN with multi-value fuzzy logic for supporting foreign currency risk management [4]. Neuro-fuzzy approach for Forex portfolio management was proposed by Yao et al. [5]. A survey of applications of ANN in financial markets can be found in [6].

Li and Suohai [7] considered Forex predictions by combining support vector regression machine (SVR) with artificial fish swarm optimization (AFSA). They proved that SVR optimized by AFSA can be successfully used in Forex predictions. They showed the advantages of their method comparing to the techniques based on combination of SVR and cross validation, genetic algorithm or particle swarm optimization. SVR models were investigated by de Brito and Oliveira ([8],[9]). In their papers they considered SVR+GHSOM (growing hierarchical self organizing maps) as well as GA (genetic algorithm) based systems. They concluded that GA model outperforms the one of SVR+GHSOM [9]. Very good performance of GA approach was stated in [10]. But, the authors indicated that when transaction costs are included, GA fails in getting positive results.

Evolutionary approach in Forex trade strategy generation has been investigated by Myszkowski and Bicz in [2]. They built the Evolutionary Algorithm based on BUY/SELL decision trees, that aimed at generating trade strategy, which consisted in technical analysis indicators connected by logical operators.

3 Evolutionary Based Trend Predictions

3.1 Improved Evolutionary Approach

To generate profitable strategies on Forex financial market, we propose application of an evolutionary algorithm, which uses closing prices of specified time interval and computes them with technical analysis indicators. Application of evolutionary algorithm allows to use any data structure. In the current paper, similarly to [2] a pair of decision binary trees has been applied. Evolutionary algorithm creates specified number of initial solutions and then modifies them with two different operators. The solutions are evaluated by fitness function to choose the best ones. The process is repeated until specified iteration number is reached. The mutation operator is responsible for generating new solutions which differ from the ones obtained in the current iteration by random modification of their parameters. Crossover operator is responsible for generating better solutions by combining two of them from the current iteration. Both of the operators have been built to assure that the new combinations of technical indicators are efficiently generated. The impact of algorithm parameters on final solution as well as the entry cost of transaction are also taken into considerations. To exclude too risky strategies maximum loss and profit are specified as input parameters. To generate new solutions effectively, mutation and crossover operators are assumed to have many different behaviors. Wide range of technical indicators such as Exponential, Simple and Weighted Moving Averages, Momentum Indicator, Commodity Channel Index, Relative Strength Index and Stochastic Oscillator [11] is considered. Finally, the impact of all parameters is measured to choose the most effective ones.

3.2 Solutions

The solutions of evolutionary algorithm take the form of pairs of binary trees where one of them is responsible for generating opening transaction signals while the other for closing transaction signals. The tree nodes can be either logical operators *AND,OR* or rules based on technical indicators and greater/smaller operators. All the parameters such as type of technical indicator, input parameter of the indicator, greater/smaller operator and value of the indicator can be modified randomly. The proposed evolutionary approach is presented on Fig. 1.

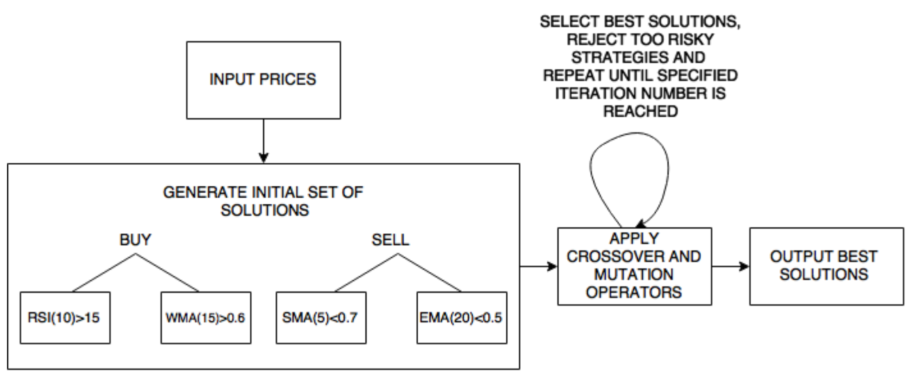

Fig. 1. Proposed evolutionary approach

3.3 Binary trees

Binary trees are built according to the following rules. The root is one of two possible logical operators *AND,OR*. Children nodes can be logical operators or technical indicator based rules. After creating the root both of the children are checked, whether for any closing price from the training data set the respective conditions are satisfied. If they are, an opening or closing signals are generated. The next children are added to the tree as long as they are technical indicator based rules or specified maximum depth of the tree is reached. The tree example is presented on Fig. 2, with technical indicators SMA, WMA, EMA, RSI.

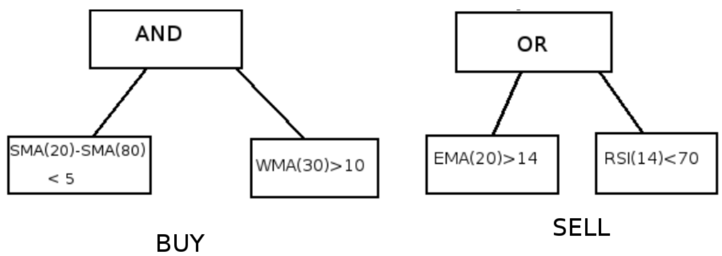

Fig. 2. Exemplary binary tree

In the algorithm such parameters as minimal and maximal tree depth or input ones for technical indicators are determined. The indicators are picked from previously created set. Their parameters are initially randomly determined and then modified by evolutionary operators. The possible value range for every indicator is calculated as a first step of the algorithm according to training and testing data sets. This ensures that mutation operator will select random values within that range. The profit is firstly calculated on the training data set, then the testing one is used. Initialisation function randomly picks tree node types, technical indicators as well as considered rules.

3.4 Evolutionary operators

There are two evolutionary operators. One of them is a crossover operator. It takes random nodes of the same type from one tree and swaps them with nodes from another tree. The tree nodes are taken from these solutions which generated the best profit on the training data. Another way of modifying solutions by crossover operator consists in taking the whole sub-trees and swapping them so that maximum and minimum tree depths are not exceeded. The third possibility is to take sub-trees from current solution and to treat them as a new one. The fact that nodes are taken from the most profitable trees makes it more probable that next solutions will be better than the previous ones. Mutation operator picks random nodes from the tree and replaces its parameters according to initialization function. That way the obtained solution satisfies all required conditions and gives a chance for creating new, completely different solution. Additionally, a solution selection operator is used. It takes a specified number of best solutions from the previous iteration and creates new solutions using crossover and mutation operators.

3.5 Fitness function

The choice of a fitness function should ensure indicating the most profitable solutions on specified data set. The profit is calculated by subtracting closing price of the opening moment from price of the closing moment. There are three types of transactions defined in the algorithm: SELL, BUY and ANY. In the first case the profit is obtained when the closing moment price is smaller than the opening moment price. In the second case the profit is got when opening moment price is smaller than closing moment price, while in the third case any of these two options can take place. Taking into account all these cases makes the algorithm more useful as various strategies can be generated depending on whether the currency is in an ascending or descending trend. Another factor included into fitness function is a number of units which will be subtracted from the profit as an entry fee. The last factors, taken into account in the fitness function, concern maximal and minimal risks, which can be determined on the selected data set. Thus all the solutions which do not satisfy the presented conditions can be omitted. Depending on the strategy fitness function may take the form:

$$F(A) = \begin{cases} \sum_{i=1}^{m}(p_k(i) - p_o(i) - P)) & \text{for BUY strategy} \\ \sum_{i=1}^{m}(p_o(i) - p_k(i) - P)) & \text{for SELL strategy} \\ \sum_{i=1}^{m}(|p_k(i) - p_o(i)| - P)) & \text{for ANY strategy} \end{cases}, \tag{1}$$

where $A = (p_o(i), p_k(i))|i = 1, 2, ..., m,\ p_o(i)$ is a price of opening transaction moment indexed with i; $p_k(i)$ is a price of closing transaction moment; m means the number of all performed transactions and P is an entry fee.

In the first step of evolutionary algorithm the first set of solutions is initialized. The number of them is specified at the beginning and is fixed. Next, the most and the least profitable solutions are indicated by using fitness function. The copies of the best solutions are used as an input for crossover and mutation operators. The new solutions substitute the worst ones indicated by fitness function. The process is repeated until specified number of iterations is reached.

4 Experiment results and discussion

The experiments aimed at evaluating the performance of the proposed algorithm. There were considered two real historical data sets: the first one of four hour interval set of GBPJPY currency pair and the second one consisting of ten minute intervals on EURUSD currency pair [12]. The first data set was used to compare the algorithm results with simple buy-and-hold strategy being a control point for profitability evaluation. The second one was also used to compare obtained results with the evolutionary algorithm (EA) described in [2]. The considered data sets contained ascending as well as descending price trends. Each of them consisted of 1417 instances. The detailed information concerning the data sets is presented in Table 1.

To obtain the best possible results, the procedure of tunning parameters has been carried out. There were examined such parameters as number of iterations,

Table 1. Datasets information

Data type	Currency Name	Time interval	Start date	End date
TRAINING	EURUSD	10 minutes	2009.09.24, 08:00	2009.10.07, 23:56
TESTING	EURUSD	10 minutes	2009.08.30, 22:01	2009.09.11, 20:00
TRAINING	GBPJPY	4 hours	2014.01.01, 23:01	2014.07.10, 21:40
TESTING	GBPJPY	4 hours	2014.07.11, 14:20	2014.12.30, 22:35

population size, numbers of mutations and crossovers, minimum and maximum tree depths, as well as minimum and maximum intervals for technical indicators. Table 2 shows the parameter values, for which the final profits for the both of the tested data sets were the highest and that have been used during experiments.

Table 2. Best result parameters

Tested parameter	Value
Number of iterations	100
Population size	80
Number of mutations	30
Number of crossovers	30
Minimum tree depth	5
Maximum tree depth	50
Minimum interval for technical indicator	5
Maximum interval for technical indicator	50

The data sets were equally divided into training and testing parts. Obtained profits for different data types, currency names and time intervals are shown in Table 3.

Table 3. Profits for the best parameters

Data type	Currency Name	Time interval	Profit
TRAINING	EURUSD	10 minutes	2490
TESTING	EURUSD	10 minutes	2326
TRAINING	GBPJPY	4 hours	2495
TESTING	GBPJPY	4 hours	2038

The performance of the proposed technique is examined by comparing the resulted profits with the ones obtained by buy-and-hold strategy for the both of the data sets. Additionally, in the case of the EURUSD data set the profits have been compared with the ones obtained by evolutionary strategy (EA) [2]. Exemplary results for time intervals of 10 minutes and 4 hours, together with the profits of buy-and-hold strategy as well as EA strategy are presented in Table 4. The first three columns contain data sets features. The three last ones show profits obtained respectively by the proposed approach, EA ([2]) and buy-and-hold strategies.

Table 4. Exemplary results

Data type	Currency Name	Time interval	Profit	EA strategy	Buy-and-hold strat.
TRAINING	EURUSD	10 minutes	2490	x	1310
TESTING	EURUSD	10 minutes	2326	2062	1770
TRAINING	GBPJPY	4 hours	2495	x	1600
TESTING	GBPJPY	4 hours	2038	x	1400

The effect analysis showed that the results got by the presented algorithm were better than the ones received by buy-and-hold and EA ([2]) strategies. The profit for the EURUSD training data set was even 90.1 per cent better than the one obtained by buy-and-hold strategy, 31.4 per cent better on the test data set and 12.8 per cent better than the ones received by EA [2]. Such amelioration is driven by main advantages of the presented algorithm over the EA approach, that consist in improving evolutionary operators as well as input parameters optimization. The main advantage of the algorithm over buy-and-hold strategy is the ability of finding local minimum and maximum of fitness function while in buy-and-hold strategy they are omitted. Analysis of entry moments showed that the proposed algorithm takes into account more minimum and maximum of price functions on time intervals, what additionally allows to obtain profits higher than the ones obtained by buy-and-hold strategy. The best entry moments for the EURUSD data sets are presented on Fig. 3.

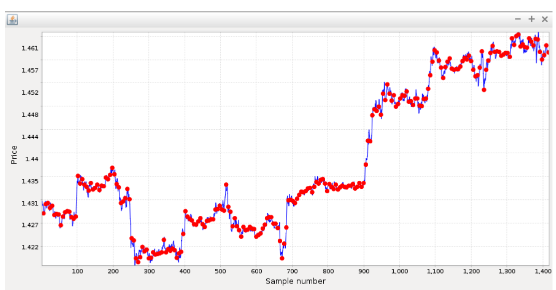

Fig. 3. Best results entry moments

5 Concluding Remarks

In the paper, the new evolutionary algorithm for generating profitable strategies on Forex financial trade market is presented. In spite of the EA approaches investigated so far, several enhancements which are to assure good performance of the evolutionary technique are introduced. Developing of the crossover as well as mutation operators improved effectiveness of the proposed algorithm. Using of the proposed form of mutation operator enables finding local extremes for different price levels. Optimal parameters tunning additionally improves effectiveness of the obtained strategies. Thanks to using technical indicators, strategies which generate too high loss during training process are detected and automatically rejected. The experiments carried out on historical data sets showed good performance of the proposed technique comparing to EA ([2]) and buy-and-hold strategies.

Future research will consist in further experiments carried out on data of different characteristics as well as in development of the presented algorithm, by including fundamental data. It will allow to take into account factors, which are unpredictable by technical analysis. Worth considering is also the problem concerning connection of correlated exchange pairs, as including their growths and falls may significantly ameliorate the performance of the algorithm.

References

1. http://www.investopedia.com/terms/forex/f/foreign-exchange-markets.asp [May 30, 2015]
2. Myszkowski, P., Bicz, A.: Evolutionary Algorithm in Forex Trade Strategy Generation. In: Computer Science and Information Technology (IMCSIT), Proceedings of the 2010 International Multiconference, pp. 81-88 (2010)
3. Ni, H., Yin, H.: Exchange Rate Prediction Using Hybrid Neural Networks and Trading Indicators. Expert Syst. Appl. 72, 2815-2823 (2009)
4. Lee, W., Wong H.T.: A Multivariate Neuro-Fuzzy System for Foreign Currency Risk Management Decision Making. Neurocomputing 70, 942-951 (2007)
5. Yao, S., Pasquier, M., Quek, C.: A Foreign Exchange Portfolio Management Mechanism Based on Fuzzy Neural Networks. In: 2007 IEEE Congress on Evolutionary Computation, pp. 2576-2583 (2007)
6. Li, Y., Ma, W.: Applications of Artificial Neural Networks in Financial Economics: a Survey. In: 2010 International Symposium of Computational Intelligence and Design, pp. 211-214 (2010)
7. Li, M., Suohai, F.: Forex Prediction Based on SVR Optimized by Artificial Fish Swarm Algorithm. In: 2013 Fourth Global Congress on Intelligent Systems, pp. 47-52 (2013)
8. de Brito, R.F.B., Oliveira, A.L.I.: Comparative Study of FOREX Trading Systems Built with SVR+GHSOM and Genetic Algorithms Optimization of Technical Indicators. In: 2012 IEEE 24th International Conference on Tools with Artificial Intelligence, pp. 351-358 (2012)
9. de Brito, R.F.B., Oliveira, A.L.I.: Sliding Window-Based Analysis of Multiple Foreign Exchange Trading Systems by Using Soft Computing Techniques. In: 2014 International Conference on Neural Networks (IJCNN) pp. 4251-4258 (2014)
10. Mendes, L., Godinho, P., Dias, J.:A Forex Trading System Based on Genetic Algorithm. J. Heuristics 18, pp. 627-656 (2012)
11. Thiel, S.: Rynek kapitalowy i terminowy (2007)(in polish).
12. http://www.forextester.com/data/datasources [May 30, 2015].