An Application of Neural Network and Hidden Markov Model: A Case Study for Euro Dollars Exchange Rate

Saeed Rasekhi[1], Majid Rastegar[2]

[1, 2] Department of Economics, University of Mazandaran, Babolsar, Iran

[1] srasekhi@umz.ac.ir; [2] majid_rastegaar@yahoo.com

***Abstract*- In this paper a robust hybrid algorithm for forecasting exchange rate trends is proposed and simulated. The existence of noise in time series of exchange rate causes many forecasting methods encounter with uncertainty prediction degree. This work introduces a coefficient framework based on artificial neural networks (ANNs) tools to overcome this problem. First time series samples of exchange rate are quantized by self organizing map (SOM) which is considered a powerful tool in ANNs. SOMs are different from other artificial neural networks (ANNs) since they use a neighborhood function that has the ability in preserving the topological of the input space. A signal processing at this level can be considered as a signal smoothing operation for denoising input data. In the next level the proposed algorithm uses Elman neural network for estimating the direction of quantized (training/test) input data. The simulated result shows that application of SOM neural network can overcome some limitations of conventional methods. To show the effectiveness and efficiency of proposed algorithm, the performance of the model is compared with a Markov model by using daily data of 7 November 2002 to 9 October 2006 to predict the trend of exchange of Dollar/Euro. The simulated results show that Markov model has superior predicting performance than proposed hybrid algorithms.**

Keywords-Exchange Rate; ANN; Markov Model; SOM Neural Network; Dollar/Euro

I. INTRODUCTION

The numerous methods available for forecasting exchange rates can be categorized into four general groups, technical, fundamental, market-based and mixed. Among them fundamental forecasting plays the most important role in prediction of price movement in many financial assets. It is based on fundamental relationships between economic variables and exchange rate. Although fundamental forecasting accounts for the expected fundamental relationships between factors and currency values, it involves many limitations. The price timing of the impact some factors on currency value is not known or coefficients derived from regression analysis will not necessarily remain constant over time. Therefore, it is very difficult to forecast foreign exchange rate with only fundamental analysis.

To forecast future movement of exchange rate by technical analysis, we need to study historical exchange rate data to discover internal structure of these data to gain a better understanding of the dynamic process by which the time series data are generated. Most of conventional exchange rate forecasting methods such as Auto Regressive (AR), Autoregressive Integrated Moving Average (ARIMA) and Generalized Autoregressive Conditional Heteroscedasticity (GARCH) are based on statistical models [1]. Many numerical or semi-analytical algorithms were presented to obtain optimal models parameters in a way that model shows best performance in forecasting future exchange rate variation [2]. Recently, application of intelligent system science domain in forecasting time series has been paid attention to by economists [3]. Among evolutionary algorithms, artificial neural networks (ANNs) due to strong capability to recognize the pattern of data structure find important significance in exchange rate forecasting. Generally, it involves a network of simple processing elements exhibiting complex global behavior determined by the connections between the processing elements and element parameters. ANNs are used with algorithms designed to alter the strength of the connections in the network to produce a desired signal flow. There are many types of ANNs, such as feed forward neural network, Radial basis function (RBF) network and learning vector quantization, etc. Among them, it is our intention to use recurrent neural networks (RNNs) [4]. Contrary to feed forward networks, RNNs are models with bi-directional data flow. While a feed forward network propagates data linearly from input to output, RNNs also propagate data from later processing stages to earlier stages. It can be used as general sequence processors [5]. Although RNN network was proved to have strong capability to predict when there is a short time correlation in time series data[6], one weak point in using this network is that in some conditions signal to noise ratio(SNR) parameter in time-series signal has low value. In this case the network performance is limited and even the algorithm which uses ANNs in is this case went to fail. To be familiar with other powerful neural network tools maybe brings in the mind an idea like the proposed framework in this paper to solve this problem. The self-organizing feature map (SOFM) is one of the major unsupervised artificial neural network models and often used to learn certain useful features found in their learning process [7]. It basically provides a way for cluster analysis by producing a mapping of high dimensional input vectors onto a two-dimensional output space while preserving topological relations as faithfully as possible. After appropriate training iterations, the similar input items are grouped spatially close to one another. As such, the resulting map is capable of performing the denoising task in a completely unsupervised fashion.

The strategy here is first use SOM to solve noisy time-series data problem. Then Elman neural network is used for forecasting of trends of exchange rate in time-series data. For better show of algorithm efficiency, it is compared with hidden Markov method. As mentioned the Elman neural network has ability to model temporary and short-term correlation while hidden Markov model has a strong mathematical structure and is suitable for modelling natural phenomenon. The paper is organized as follows. Section 2 presents characteristics of financial time series. Apply procedure of neural network and hidden Markov model in economy systems is discussed in Section 3. In Section 4 simulation results of the proposed algorithm are presented and analyzed. Finally, Section 5 makes concluding remarks.

II. CHARACTERISTICS OF FINANCIAL TIME SERIES

Financial time series data are a sequence of prices in some financial assets over a specific period of time. Over the past 50 years, there has been a large amount of intensive research on the stock market [8]. All the financial analysts carry out many kinds of research in the hope of accomplishing one goal: to beat the market. To beat the market means to have a rate of return that is consistently higher than the average return of the market while keeping the same level of risk as the market [9]. Generally, there are some main streams in financial market analysis. Among them fundamental analysis can be considered as one of the important analysing method. Fundamental analysis is the examination of the underlying forces that affect the well-being of the economy, industry sectors, and individual companies. For example, to forecast future prices of an individual stock, fundamental analysis combines economic, industry, and company analyses to derive the stock current fair value and forecast its future value. If fair value is not equal to the current stock price, fundamental analysts believe that the stock is either over or under valued and the market price will ultimately gravitate toward fair value. There are some inherent weaknesses in fundamental analysis. First, it is very time-consuming: by the time fundamental analysts collect information of the company or economy health and give their prediction of the future price, the current price has already changed to reflect the most updated information. Second, there is an inevitable analyst bias and accompanying subjectivity. Maybe it was an important reason to make scientists interested in economy toward artificial intelligent expert systems to cover these weaknesses.

The oldest and the most famous theory about assets value was introduced by Martingale. The theory claims that the value of assets is not systematically predictable. Efficient market hypothesis (EMH) as an important theory in financial theory is accepted by many researchers such as Antony and Briggs (1995), Malkiel (1997) and White (1998). Regarding to Fama, the efficiency means that current value reflects all available information [10]. Robert (1987) puts the hypothesis into three categories:

1) Weak form based on historical information;

2) Semi-strong form based on all data in the market except the confidential information;

3) Strong form based on all data including confidential information.

In this article, we have applied the first form of EMH. Regarding to this form of hypothesis, a financial time series can be modelled as a summation of all noises in each stage. That is:

$$x(k+1) = x(k) + \varepsilon(k) \tag{1}$$

Where *ε(k)* is a Gaussian variable that its mean and variance is zero and σ respectively. ***x(k)*** is the best prediction where ***x(k+1)*** is an estimation in the time of ***(k+1)***. This best prediction will be obtained if the time series is random walk. So, we can suppose a predictable combination such as:

$$x(k+1) = x(k) + f[(x(k), x(k+1), ..., x(k-n+1)] + \varepsilon(k) \tag{2}$$

Above function is nonlinear and the best prediction is shown below:

$$x_t(k+1) = x(k) + f[x(k), x(k-1), ..., x(k-n+1)] \tag{3}$$

Where *x(k+1)* is an estimation at the time of *(k+1).*

$$\delta(k+1) = f[\delta(k), \delta(k-1), ..., \delta(k-n+1)] + V(k) \tag{4}$$

Where x(k) is the main series, $\delta(k+1) = x(k+1) - x(k)$, $V(k)$ is a Gaussian variable and *δ(k)* is obtained by the first difference order. The best prediction is $\delta_t(k+1) = f[\delta(k), \delta(k-1), ..., \delta(k-n+1)]$ where *δ(k+1)* is an estimation of series at the ***(k+1)*** and f(.) is a nonlinear function.

III. APPLY PROCEDURE OF NEURAL NETWORK IN ECONOMIC SYSTEMS

Neural networks are powerful forecasting tools that draw on the most recent developments in artificial intelligence research. They are non-linear models that can be trained to map past and future values of time series data and thereby extract hidden structures and relationships that govern the data. Neural networks are applied in many fields such as computer science, engineering, medical and criminal diagnostics, biological investigation, and economic research. They can be used for analysing relations among economic and financial phenomena, forecasting, data filtration, generating time-series, and optimization.

Figure 1 indicates a general diagram block for recognition system on changes trend of Dollar to Euro. *Y(k)* is the main time series that illustrates the exchange rate series.

Fig. 1 General diagram block of recognition system of exchange rate trend

Preprocessing in expert systems is one of the most common versions of neural network to help us to apply data in a better form. In preprocessing of this system, we should calculate the first difference of series, $\delta(k)$, by below form:

$$\delta(k) = y(k) - y(k-1) \qquad k = 1,2,\dots,N-1 \tag{5}$$

Where, $y(k)$ is the main time series of exchange rate and $\delta(k)$ is also obtained from the first order differences. In this framework, we use the logarithm function as follows:

$$x(k) = sign[\delta(k)]\log(|\delta(k)| + 1) \qquad k = 1, 2, \dots, N-1 \tag{6}$$

Where $x(k)$ denotes the logarithm conversion of the first order difference series. It has a wider dynamic state than $\delta(k)$ which is the first order difference of time series. Then $x(k)$ is considered with delay of d_1.

$$x(k, d_1) = [x(k), x(k-1), x(k-2), \dots, x(k-d_1)] \tag{7}$$

In this article, d_1 is equal to 2 since the trend of changes in recent times has more effect while predicting the next time. The other reason is to avoid extra complexity. $X(k, d_1)$ is a state vector and an input of SOM. Then a recurrent neural network will teach SOM output well.

A. Self Organizing Map

Self Organizing Map (SOM) by Teuvo Kohonen [11] provides a data visualization technique which helps to understand high dimensional data by reducing the dimensions of data to a map. SOM also represents clustering concept by grouping similar data together. Therefore, it can be said that SOM reduces data dimensions and displays similarities among data. With SOM, clustering is performed by having several units compete for the current object. Once the data have entered into the system, the network of artificial neurons is trained by providing information about inputs. The weight vector of the unit is closest to the current object and becomes the winning or active unit. During the training stage, the values for the input variables are gradually adjusted in an attempt to preserve neighbourhood relationships that exist within the input data set. As it gets closer to the input object, the weights of the winning unit are adjusted as well as its neighbours. An overall demonstration of self organization mapping system is depicted in Fig. 2.

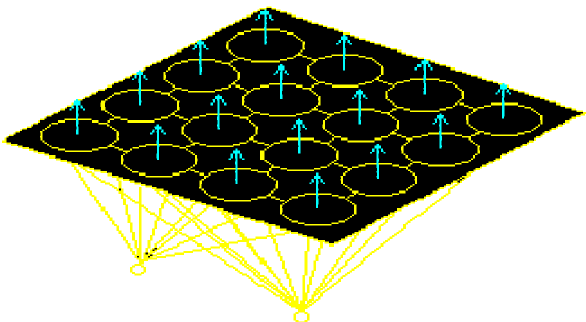

Fig. 2 Self Organizing Map

SOM is mapping pattern from R_n to a regular topology with a set of nodes that usually accompany with dimensional reduction. Each node is marked via a reference vector $m_i = [\mu_i, \mu_{i2}, \dots, \mu_{in}]^T \in R^n$. Assume that there is M groups and during teaching time of each input, $X \in R^n$ is compared with m_i; $i = 1, 2\dots$, M and then the closest match is recognized. Inputs are mapping with this plot in SOM. Nodes are updated with the below equation in SOM:

$$m_i(t+1) = m_i(t) + h_{ci}[x(t) - m_i(t)] \tag{8}$$

Where "t" is time during learning and $h_{ci}(t)$ is the neighbourhood function. Usually its form is $h_{ci}(t) = h(\|r_c - r_i\|, t)$, where "$r_c$" and "$r_i$" are the nodes position in output space of "D" in SOM. R_c is the node with the closest weight vector compared with the input sample and r_i is range on all nodes. Whenever $\|r_c - r_i\|$ increases and also "t" closes to the extreme side, $h_{ci}(t)$ will be close to zero. A following neighbourhood function is frequently used:

$$h_{ci} = \alpha(t)\exp\left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}\right) \tag{9}$$

Where "exp" and $\|\cdot\|$ are exponential form and norm respectively, $\alpha(t)$ is the learning rate and $\delta(t)$ defines the width of the core. Generally both of them decrease uniformly over time. Using the neighbourhood function means that nodes which are similar in the topology to each other are moved to input pattern and it will create a steady effect that results in the general ordering of map. Notice that $\delta(t)$ should not decrease consumedly because when neighbourhood nodes are not updated, map will lose its steady structure.

B. Elman Network

Elman Networks are a form of recurrent neural networks which have connections from their hidden layer back to a special copy layer. This means that the function learnt by the network can be based on the current inputs plus a record of the previous state(s) and outputs of the network. In other words, the Elman network is a finite state machine that learns what state to remember (i.e., what is relevant). The special copy layer is treated as just another set of inputs and so standard back-propagation learning techniques can be used (something that is not generally possible with recurrent networks) [12, 13].

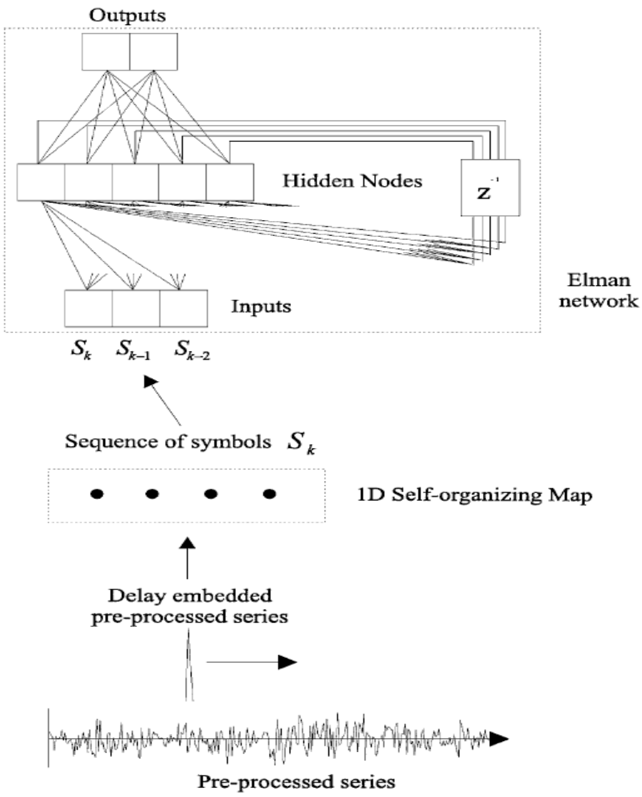

Fig. 3 Recognition system of changes trend of Dollar to Euro

For Elman network, we have:

$$O(k+1) = C^T Z_k + C_0 \tag{10}$$

$$Z_k = F_{nk}(AZ_{k-1} + BU_k + b) \tag{11}$$

Where, "*C*" is a "$n_k * n_0$" vector that explains the weight of hidden layer to output node and n_0 is the number of output nodes. Also $Z_k \ R^{nh}$ is the output of the hidden layer. *U(k)* comes below and d_2 is the dimension of the delay:

$$U(k) = \begin{pmatrix} S(k) \\ S(k-1) \\ \vdots \\ S(k-d_2) \end{pmatrix} \tag{12}$$

Also, A and B matrices with appropriate dimensions indicate feedback weights from hidden node to another hidden node and from weights of the input layer to the hidden node. Furthermore, F_{nk} is a complex function. In this article, n_0 is equal to 2 that one of the outputs is educated for positive changes and another one is done for negative changes. Therefore, for whole system:

$$O(k+1) = F_1(d(k), d(k-1), d(k-2), d(k-3), d(k-4), d(k-5)) \tag{13}$$

$$O(k+1) = F_1[y(k), y(k-1), \ldots, y(k-6)] \tag{14}$$

Attention must be paid that F_1 is not a constant function but it depends on the internal state of the hidden knowledge network in the weights.

C. Coping with Non-Stationary

Stationary is used as a tool in time series analysis, where the raw data are often transformed to become stationary; for example, economic data are often seasonal and/or dependent on a nonstationary price level. An important type of non-stationary process that does not include a trend-like behaviour is the stationary process. In fact, the non-stationary characteristic implies that the distribution of financial time series is changing over time. Figure 4 illustrates a financial time series which is non-stationary. As a system educates based on Part A and consequently we test it at the Part B, system will have a little chance to succed. In fact, financial time series have no high stationary.

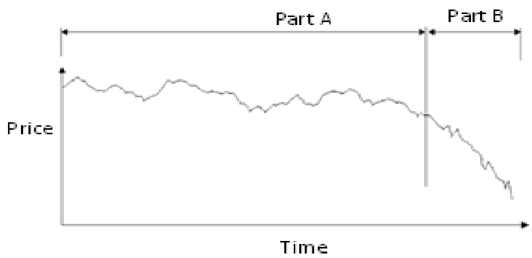

Fig. 4 A financial time series with no stationary

To deal with this problem, as was shown in Figure 5, first some samples are chosen as the best set from the whole available samples, then system is tested with other samples. Finally whole set is tested again and the best samples are shifted to forward with the size of the test set [14].

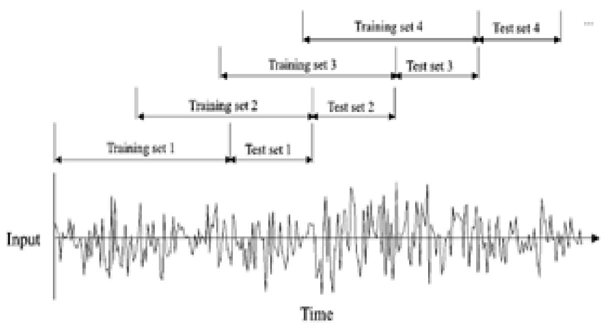

Fig. 5 Choice manner of exam and best sets

Source: Giles et al (2001)

D. Predicting Systems Based on Markov Model

Markov models are used to train and recognize sequential data, such as speech utterances, temperature variations, biological sequences, and other sequence data. In a Markov model, each observation in the data sequence depends on previous elements in the sequence. Consider a system where there are a set of distinct states, S = 1, 2,…,N. At each discrete time slot *t*, the system takes a move to one of the states according to a set of state transition probabilities *P*. We denote the state at time *t* as s_τ . In many cases, the prediction of the next state and its associated observation only depends on the current state, meaning that the state transition probabilities do not depend on the whole history of the past process. This is called a first order Markov process. For example, knowing the number of students admitted this year might be adequate to predict next year admission. There is no need to look into the admission rate of the previous years. Here we do not want to discuss much about this model. It just used to have better presentation of proposed algorithm. For more information, there are the Markov properties described in [15]. The same procedure like using Elman network happens here for Markov model. The classified value of output neurons in SOM network is applied to the input Markov model. After learning the Markov model, parameters of Markov model, that are transferred by matrices of the model are utilized. At the test stage, the next series of the model is obtained through calculated parameters. For obtaining the logical solution, algorithm here is repeated for each predicting sample and frequent events are included as the solution.

IV. EXPERIMENT AND RESULTS

Table 1 indicates simulation results of neural network system in predicting the exchange rate (Dollar against Euro). In this table, we have used 10 hidden units (neuron), 100 training sets, SOM size of 5 nodes, and with different test sets from 10 to 100. As this table shows, after simulation of systems, at the best state, system could predict the trend with the rate of error about 46.88 percent. In comparison to this one, the least possible error percent is less in HMM. We have used Markov model in two ways. At first Markov model has performed with the dimension of SOM equal to 8 in Table 2 and then we changed dimension of SOM with number of Markov state equal to 5 in Table 3. The findings of these two ways are the same and equal to 46.5.

TABLE 1 SIMULATION RESULTS OF NEURAL NETWORK SYSTEM IN PREDICTING OF THE EXCHANGE RATE

Training Set	Test Set	Hidden Neuron (ELMAN)	SOM Size	Error %
100	10	10	5	47.44
100	20	10	5	47.33
100	30	10	5	46.88
100	100	10	5	49.55

TABLE 2 SIMULATION RESULTS OF MARKOV MODEL SYSTEM IN PREDICTING OF DOLLAR TO EURO (THE DIMENSION OF SOM NEURAL NETWORK IS EQUAL TO 8)

Number of Markov States	3	4	5	6	7	8
Error (%)	54	48.5	46.5	49	51.5	52.5

TABLE 3 SIMULATION RESULTS OF MARKOV MODEL SYSTEM IN PREDICTING OF DOLLAR TO EURO (THE NUMBER OF MARKOV MODEL STATE IS EQUAL TO 5)

Dimension of SOM	3	4	5	6	7	8
Error (%)	51	49	46.5	49.5	51.5	54.5

V. CONCLUSION

The purpose of this paper is to compare two methods including neural networks and Markov model in predicting exchange rate. For this, we have employed daily Dollar/Euro exchange rate data of 7 November 2002 to 9 October 2006. Based on the obtained results, it seems that Markov model is better than the former method. Although with respect to low number of data, decisive assessment is not possible according to the weak dependence between data, Markov modelling has resulted in better results. This is mentionable that if data have more specific regularity, the neural networks will have more logical result. However, we can test the programs on more range of data and compare results in a better manner. The research work can also be improved by developing an analytical model using artificial neural network with other model or any suitable analytical method.

REFERENCES

[1] F. Robert Engle, "Auto Regressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation", *Econometric Society Table*, vol. 50, iss. 4, pp. 987-1007, 1982.

[2] F. Canavo, "Modeling and Forecasting Exchange Rate with Beysian Time-Varying Coefficient Model", *Journal of Economic, Dynamics and Control, North-Holland*, vol. 17, pp. 233-261, 1993.

[3] J. F. Chang, "Forecasting Exchange Rates by Genetic Algorithms Based Back-Propagation Network Model", *Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Date of Conference*: 12-14, 2009.

[4] A. K. Nag and A. Mitra, "Forecasting Daily Foreign Exchange Rates Using Genetically Optimized Neural Networks", *Journal of Forecasting*, vol. 21, pp. 501-511, 2002.

[5] P. Tenti, "Forecasting Exchange Rates Using Recurrent Neural Network, *Applied Artificial Intelligence*", vol. 10, pp. 567-581, 1996.

[6] R. F. Ghazali, A. J. Hussain, M. N. M. Salleh, "Application of Polynomial Neural Networks to Exchange Rate Forecasting, "*International Conference on Intelligent Systems Design and applications,* pp. 90-95, 2008.

[7] Shu. Ch .Li, Kuo; Yi. Ch. Cheng, and M-H. Ho and Sh. Tun, "Knowledge Discovery with SOM Networks in Financial Investment Strategy", *Proceedings of the Fourth International Conference on Hybrid Intelligent Systems*, 2004.

[8] B. Wuthrich, Ch. V, Leung, S. Permunetilleke, D. Sankaran and K. Zhang, "Daily Stock Market Forecast from Textual Web Data , Systems, Man, and Cybernetics", *IEEE International Conference on, Date of Conference*: vol. 3, pp. 2720-2725,1998.

[9] E. F. Fama, "The Behavior of Stock Market Prices" *Journal of Business*, pp. 34-105**,** 1965.

[10] E. F . Fama, "Efficient Capital Markets: A Review of Theory and Empirical Work", *Journal of Finance*, pp. 383-417, 1970**.**

[11] T. Kohonen, "Self Organizing Map", *IEEE,* vol. 78, iss. 9, pp. 1464-1480, 1990.

[12] J. L. Elman, "Distributed Representations, Simple Recurrent Neural Network and Grammatical Structure", *Machine Learning*, vol. 7, iss. 2, pp. 195-226, 1991.

[13] J. L. Elman, "Finding Structure in Time", *Cognitive Science*, vol. 14, iss. 2, pp. 179-211, 1990.

[14] C. L . Giles , S. Lawrence and Ah. Ch. Tsoi, "Noisy Time Series Prediction Using Recurrent Neural Network and Grammatical Inferences", *Machine Learning*, vol. 44, iss. 1, pp. 161-183, 2001.

[15] Y. Zhang and B. Eng, "Prediction of Financial Time Series with Hidden Markov Models", *Computing Science, Shandong University*, pp. 13-14, 2004.

Dr. Saeed Rasekhi is an Associate Professor in Economics at the University of Mazandaran of, Babolsar, Iran. He has received his PhD in International Economics and Mathematical Economics at the University of Allameh Tabatabaee of, Tehran, Iran. Also, he has achieved his MA in Energy Economics at the University of Tehran, Iran. He has taught different courses of economics at the University of Mazandaran since 2002. He has written several papers and books about international economics and other topics of economics.

Majid Rastegar has received his MA in Economics from University of Mazandaran, Iran. He is a PhD student in Economics at the university. The field of his interest is Financial Economics.