Modeling Electromagnetic Susceptibility of Processor Using MultiLayer Perceptron

R. M. Alaoui*[1]; L. Bouhouch[2]

Department of Electrical Engineering, University Ibn Zohr, Agadir, Morroco

*[1]r.alaoui@iuz.ac.ma; [2]l.bouhouch@ iuz.ac.ma

Abstract

In this paper, we present an electromagnetic susceptibility study of processors. On a standardized test bench, our investigations have precisely defined the susceptibility and immunity of the microcontroller under test. To predict the resulting power failure at the processor, a model was developed based on multilayer neural networks MLP. The comparative study between experimental observations and the values estimated by our model has a 1.6 of MSE dBm and a 0.99 of regression. In addition to these results, and the stability of our model, in this work the effectiveness of models has been validated based on artificial intelligence, and ease of use by the final user.

Keywords

Susceptibility Electromagnetic; DPI; Multi-Layer Perceptron; MSE, Regression

Introduction

With advances in microelectronics technology and the trend toward miniaturization of integrated circuits, the electronics industry, computing and telecommunications converge on embedded processors connected in networks, especially in the automotive and aerospace *(C.L. Liu & all)*.

Obviously, evolution has enabled lithography manufacturing processors with a frequency more and more important, thus accelerating the processing capacity; unfortunately this is accompanied by a sharp deterioration in the appearance of these electronic systems electromagnetic *(Dominique Paret)*. Figure 1 illustrates the phenomenon of aggression by an electromagnetic disturbance mediated by coupling to an electronic component on a plane.

In this regard, Electromagnetic Compatibility became a normative obligation for which the consumer electronics came into force on 1 January 1996. As an indication, the trend toward smaller dimensions, higher clock frequencies and the increasing complexity of electronic systems, processors behave as switches internal currents up to considerable values, creating an electromagnetic emission (Figure 2) that can affect radio frequency communications, which refers to the security systems by mode coupling leads or radiated. The expertise and knowledge in electromagnetic compatibility level components have become an industrial issue, which has encouraged reconciliation in research and development between industry and academic research laboratories.

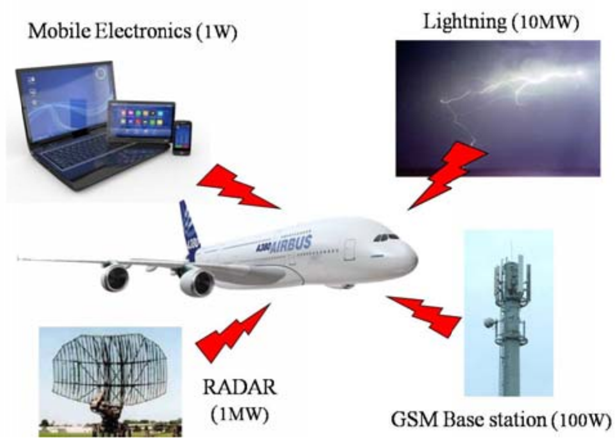

FIG. 1 EXAMPLES OF ELECTROMAGNETIC DISTURBANCES FOR A PLANE

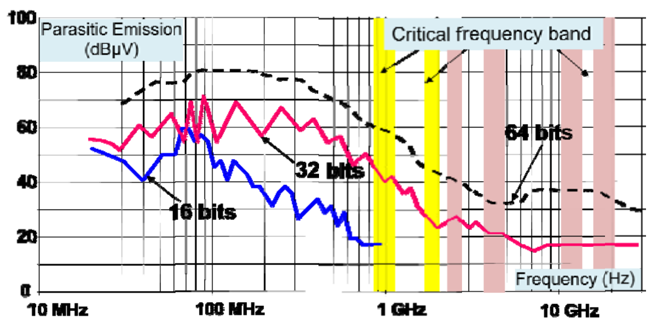

FIG. 2 MICROELECTRONICS EVOLUTION TO SMALL DIMENSIONS, WITH THE INCREASE OF ELECTROMAGNETIC EFFECTS

To help solve this problem for processors, our component under test is developed by the S12X microcontroller Fresscale *(Freescale)*. This is used to drive the intelligent case of constraint (BSI) and currently the heart of integrated vehicle CAN networks providing the roles of coordinator and CAN gateway, system safety, diagnostics, driver assistance, etc. Therefore, it is recommended to subject the processor installed on a bench complying with EMC requirements, an intense burst of electromagnetic aggression with a standardized method to better define its two spaces: susceptibility and immunity. We explain in the next section the general concept of the evaluation, the worst-case approach used, the method using DPI, the tester and the interpretation of results obtained.

Investigation Tools

In practice, an EMC assessment must follow a defined methodology. Generally, three methods are available for EMC assessment:

- Application of EMC harmonized standards;
- An EMC assessment where no harmonized standards has been applied and the manufacturer applies its own methodology;
- A mixed assessment, combining the above two methods. For example, one can apply the harmonized EMC standards to cover emission phenomena and detailed technical evaluation EMC aspects of immunity.

Only those aspects of immunity will be our investigations.

Worst Case Approach

The EMC assessment will be available when it has been conducted on a configuration producing the maximum disturbance and giving the maximum sensitivity of the device to electromagnetic interference. This method is often called the scenario "worst case" and aims to minimize the costs of the assessment. It is this approach that should be adopted to attack our strong component under test during the experiment *(Guide d'application)*.

Test Bench Methods and Measurement Results

The evaluation of the sensitivity of components required the development and standardization of specific methods, grouped under the *(IEC-62132-2)*. These methods differ in the technique used to inject the disturbance in the circuit. As well in the type of coupling leads or radiated fields and the frequency of use. The method we use for this study is the method-Power Direct-Injection (DPI), characterized by simplicity, inexpensiveness, expandability to 10 GHz. Its implementation is described in Figure 3. A signal controlled by computer is amplified before injected by capacitive coupling on the component. A measure of a real-time processor output alerts the measuring device when the signal is out of range (time and voltage) of a predefined template.

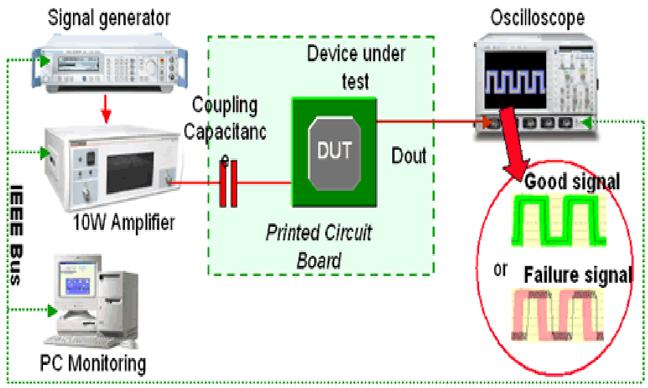

FIG. 3 PRINCIPLE OF DPI BENCH INJECTION

Figure 4 illustrates a sample of those things or each point corresponding to a vector susceptibility characterized by a forward power of aggression expressed in dBm for a given frequency.

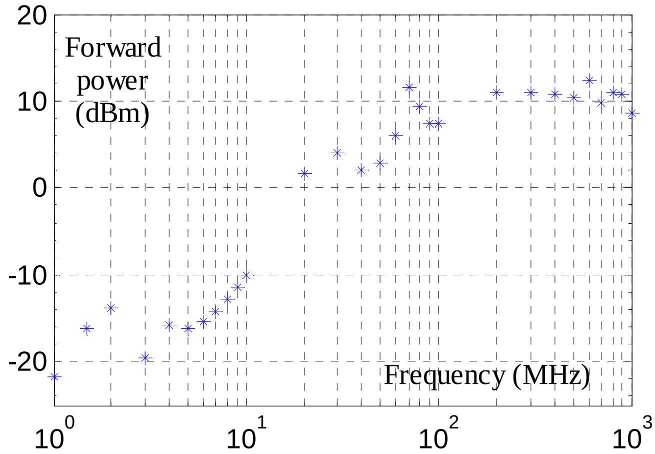

FIG. 4 SUSCEPTIBILITY OF S12X MICROCONTROLLER

All vectors of susceptibility, with powers equal and greater as indicated on the curve with their respective frequencies, are the space of the processor which induces susceptibility to deterioration during its service life. However, the processor immunity is represented by vectors with powers below the previous space vectors with the same frequency. A brief reading of the measured values shows that the variation in susceptibility and/or immunity of the device under test is not linear.

Note that the dBm (sometimes dBmW) is an abbreviation of the power ratio in decibels (dB) between the measured power and one milliwatt (mW). It is used in radio networks, microwave and optical fiber as a convenient measure of absolute power because of its ability to express both very large values and very small in abbreviated form. There are also dBmW, which are reported to one watt (1000 mW).

Modeling

Electromagnetic Compatibility EMC is based on Maxwell's equations, which give a precise mathematical framework and the basic concept of field in physics introduced by Faraday in the 1830s. In the most general case, we must speak of the electromagnetic fields, some of which identify with those components of the electric field and others to those of the magnetic field. But this synthesis of the electromagnetic phenomenon is more in transmission and susceptibility that has been little researched. Our investigations, using the method DPI, have identified two areas of susceptibility and immunity of the device under test. It is seen in section models that the hyper plan would be determined that separates the two areas experienced.

Mathematical Modelling

Research on the mathematical model, corresponding to the susceptibility, is to determine an unknown function f(x) explicitly, but only at certain frequencies known evaluable by an expensive computation. Our principle in this work is to identify f by a simple function, easy to evaluate. The problem is the existence of an infinite family of solutions by polynomial functions, exponential and trigonometric.

According to Weirstrass Approximation Theorem, any function continuous over a closed interval can be approximated arbitrarily well by polynomials. The theorem, nevertheless, does not tell us how to construct such a polynomial to satisfy a specified error range allowed for approximation. In addition, polynomial interpolation does not fit any continuous function properly. Also in the case of equally spaced nodes, polynomial interpolation may become unstable

In general, *(Tam, K.Y & all*) compared the performance of neural networks to that of other techniques such as interpolation analysis and found that neural networks outperform other techniques.

Modelling by Multilayer Neural Network

To better evaluate the electromagnetic susceptibility of our microcontroller, the model has been developed based on networks neurons as shown in Figure 5.A neural network is a statistical analysis tool to build a model of behaviour from a series of experimental observation, and consists of a directed weighted graph whose nodes symbolize the neurons having an activation function that can influence other neurons in the network. Connections between neurons, known as synaptic connections, spread the activity of neurons with a weighting characteristic of the connection. Synaptic weight is called the weighting of synaptic connections.

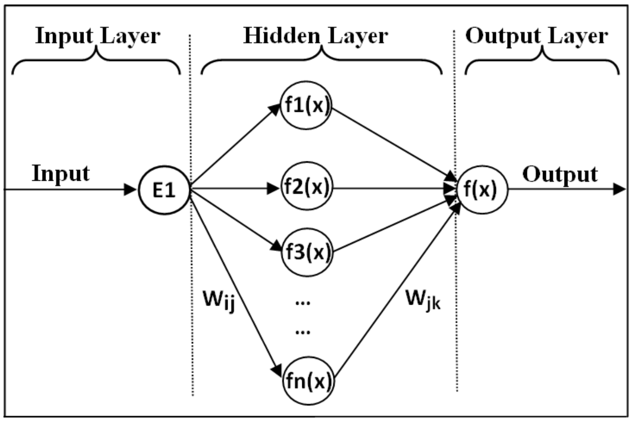

FIG. 5 MULTILAYER NEURAL NETWORK MLP

In this section programs of multilayer neural networks MLP in Matlab have been developed, in which the back propagation gradient is applied to identify different curves that best represent our experimental data of susceptibility.

To evaluate the performances of our model, the Root Mean Squared Error (RMSE) and the regression analysis are employed.

In science field, the use of RMSE is very common and it makes an excellent tool of error metric for numerical predictions.

The popular formula of RMSE is:

$$RMSE = \sqrt{\frac{1}{n}\sum_i (p_i - a_i)^2} \qquad (1)$$

Where: a: Actual target; p: Predicted target.

An MSE of zero, meaning that the estimator p predicts observations of the parameter a with perfect accuracy, is the ideal, but practically never possible. Minimizing MSE is a key criterion in selecting estimators.

While regression analysis is used, to find equations that fit data. Once the equation is acquired, the model can be utilized to make predictions. One type of regression analysis is linear analysis. When, the correlation coefficient shows that data is likely to be able to predict future outcomes.

The most common method for fitting a regression line is the method of least-squares which calculates the best-fitting line for the observed data by minimizing the sum of the squares of the vertical deviations from each data point to the line.

The least squares regression line is a mathematical model for the data, and the line that makes the sum of the squares of the residuals as small as possible. When, *a* residual is a difference between an observed *y* and a predicted *y*, the equation of the least squares regression line of *y* on *x* is:

$$y-\bar{y}=b(x-\bar{x}) \tag{2}$$

Where b is:

$$b=\frac{S_{xy}}{S_{xx}}=\frac{\sum_i (x_i-\bar{x})(y_i-\bar{y})}{\sum_i (x_i-\bar{x})^2}=\frac{\frac{\sum x_i y_i}{n}-\bar{x}\bar{y}}{\frac{\sum x_i^2}{n}-\bar{x}^2} \tag{3}$$

Note that *b* is slope and *a* is the *y*-intercept. where $\bar{x},\bar{y}$ are the mean of the observed *x* and *y* data and n is the number of data pairs (x_i, y_i).

The fraction of the total variation in the dependent variable that is explained by the independent variable is known as the coefficient of determination R^2, which is calculated as:

$$R^2=1-\frac{SEE}{TSS} \tag{4}$$

Where SSE, the sum of squares errors, is given by:

$$SEE=\sum_i^{=}(y_i-\hat{y}_i)^2 \tag{5}$$

And TSS, the total sum of squares, is given by:

$$TSS=\sum_i^{=}(y_i-\bar{y})^2 \tag{6}$$

And where $\hat{y}_i$ are the predicted *y* values at each x_i:

For simple least squares regression (i.e. only one independent variable), the square root of R^2 is equivalent to the simple correlation coefficient *r*. That one may alternatively be calculated as:

$$r=\frac{S_{xy}}{S_{xx}}=\frac{(\sum_i (x_i-\bar{x}))(\sum_i (y_i-\bar{y}))}{\sqrt{(\sum_i (x_i-\bar{x})^2)(\sum_i (y_i-\bar{y})^2)}} \tag{7}$$

r provides a quantitative measure of the linear relationship between *x* and *y*, ranging from -1 to +1: a value of $r = -1$ or $+1$ indicates a perfect linear fit, and $r = 0$ indicates that no linear relationship exists at all.

As $\sum_i (y_i-\hat{y})^2$ the sum of squared errors between the observed and predicted y-values, tends to zero, so r^2 tends to 1 and therefore *r* tends to -1 or +1, its sign depends on whether *b* is negative or positive respectively.

Results and Discussions

Before the use of the classification capabilities of a neural network, the structure has been built for our experimental database (training set and test set), and then the network was set up by using an algorithm to learning the recognition phase *(Gérard Dreyfus and all)*.

Indeed, the recognition phase, is based on the Levenberg-Marquardt (Michael Lamptona) in our case, and the network is to present each of the vectors of the test database. Levenberg uses the second derivative of a function to reduce the number of iterations of an optimization algorithm. The corresponding output is calculated by propagating the vector through the network. The response of the array is read directly from the output units and compared to the expected answer. Once the network has acceptable performance, it can be used to meet the need behind its construction. A target data vector is experimental data used on training or validation. Output data vector is response of the model that is composed of three neurons in hidden layer. To evaluate our model, the simulation has been performed in two stages. The first stage is training model where the simulator returned experimental observations as follows: 70% for training, 15% for testing and 15% for the validation stage. In the second stage we call validation which our own verification has been implemented to evaluate the model using experimental observations which were not used in the first stage. Figure 6 shows the results obtained in stage 1 by our model under training. The target data vector is our experimental forward power used for training versus their frequencies. The susceptibility prediction by our model varies from -18 dBm to 10 dBm (forward power).

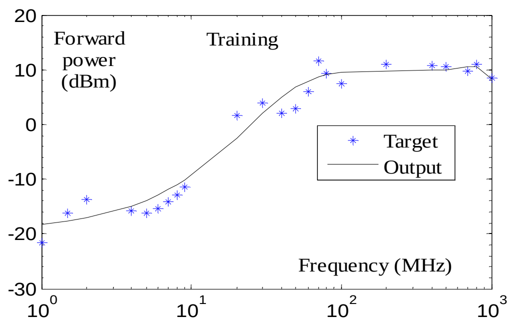

FIG. 6 MLP TRAINING

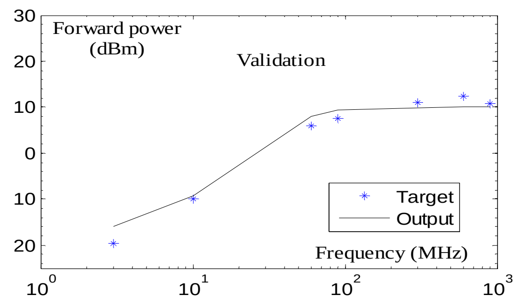

FIG. 7 MLP VALIDATION

In the second stage, the simulation results of validation model are presented in the figure 7 showing that our model follows the experimental observations.

Figure 8 shows the error estimating the deviation between the experimental observations (Target) and outputs predicted by the MLP during both stages.

Curve (a) describes the error in training stage that ranges from -4 to 4dBm with 0.71dBm of RMSE, while curve (b) shows the error in the validation stage that ranges from -3.4dBm to 2dBm with 23.10^{-3}dBm of RMSE.

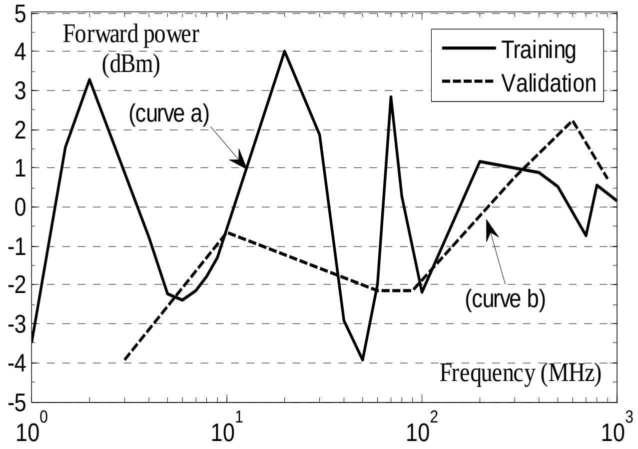

FIG. 8 COMPARISON OF MODEL ERRORS AT BOTH STAGES

The figure 9 shows the graphs of regression model R for the different phases of the simulation stage 1: R is 0.98 for the training phase, 0.99 for validation and testing phases. After adjusting the parameters (weights) was made, the network neurons has a capacity of generalization of data not present in the training set evaluated by 0.98 of regression.

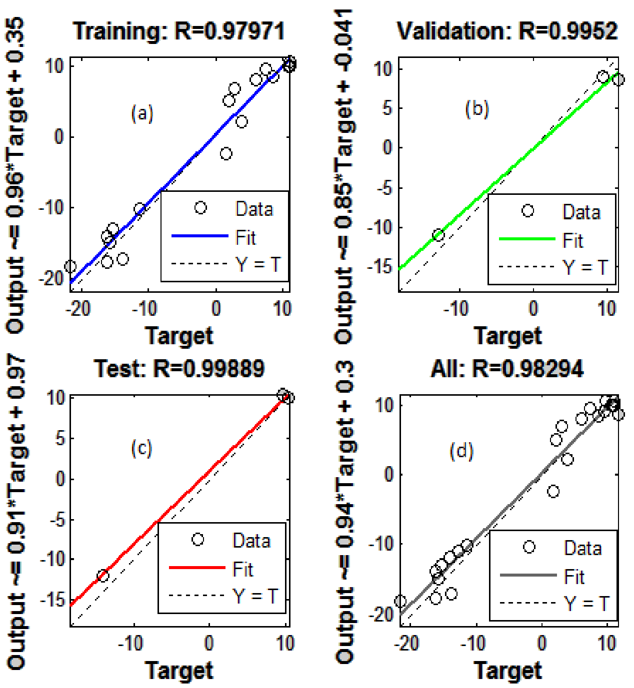

FIG. 9.REGRESSION OF MODEL IN FIRST STAGE

MLP is fast when running determined with nonlinear functions of linear combinations *(C.L. Liu)*. This allows the use of feature vectors large enough, increasing the discriminating power of the classifier. Moreover, they allow obtaining the output of probabilities, which is valuable to interpreting the results.

The implementation of an MLP is easy, and usually gives satisfactory results. Neural networks, due to their structure, are able to determine the boundaries of complex decision.

The big problem is that the MLP cannot know a priori the size of the network for a given problem. Experience shows that it is not necessary to have more than one hidden layer for the best results *(J. Liu)*. However, we cannot determine in advance the number of neuron in the hidden layer required for a given problem.The adjustment of connection weights is also difficult to optimize, and it is necessary to set these parameters by trial and error, ie we look at the behaviour of the network based learning, and choose the best configuration. In addition, the number of iterations during the learning phase is a parameter which must be found empirically. This one is important as it appears after a certain number of iterations the well-known phenomenon of over-learning, in which the MLP begins to learn by heart the examples in the training set and loses its ability to generalize. Using a validation dataset, we can calculate the generalization error of the network depending on the number of iterations. In addition the number of iterations has been naturally selected that minimizes the generalization error. Thus a limitation of the MLP is that it requires substantial foundation for learning.

Conclusion

In this work, we set the standard method of measuring the electromagnetic susceptibility of a microcontroller used for critical applications with elevated level of safety, and then the MLP network with one hidden layer of only three neurons to model the phenomenon of susceptibility of our microcontroller under test. Graphs illustrating both developed simulation stages show the stability of this model. Once the elaboration phase of the model and learning taking some time are made, the electromagnetic susceptibility is predicted. According to the results obtained, it can be noted that our model based on artificial neural networks is the best tool to model susceptibility and immunity EMC electromagnetic integrated circuits.

ACKNOWLEDGMENT

The authors would like to thank Etienne Sicard and

Alexandre Boyer (INSA, Toulouse, France) for their valuable discussions and support, as well as Stephane Baffreau (IUT Tarbes, France) for their help with setting up the DPI test bench.

REFERENCES

C.L. Liu, K. Nakashima, H. Sako, H. Fujisawa. "Handwritten digit recognition using state-of-the-art techniques", IEEE, 2002.

Dominique Paret, "Réseaux multiplexés pour systèmes embarqués CAN, LIN, FlexRay, Safe-by-Wire", Dunod, 2004.

Freescale, Microcontrôleur S12X, www.freescale.com, 2005.

Gérard Dreyfus and all "Apprentissage statistique Réseaux de neurones - Cartes topologiques - Machines à vecteurs supports", Edition EYROLLES Collection, Algorithmes, 2008.

"Guide d'application de la directive Européenne sur la CEM 2004/108/CE", http:// cmrt.centrale-marseille.fr/ electromagnetisme/ veille/guide_2004_108_francais.pdf, (27 février 2007).

H.A. Thompson, "A CAN bus-based safety-critical distributed systems", Microprocessors and Microsystems, 345-355, 2006.

IEC 62132- 2 "Direct RF power injection to measurement method", 2005.

J. Liu, P. Gader, "Neural networks with enhanced outlier rejection ability for off-line handwritten word recognition," Pattern Recognition, 2002.

Michael Lamptona "Damping–undamping strategies for the Levenberg–Marquardt nonlinear least-squares method", Center for EUV Astrophysics, University of California, Berkeley, California 94720-5030, 1996.

Tam, K.Y. and M.Y Kiam, "Managerial applications of neural networks: the case of bank failure predictions", Management science, Vol. 38 n 7, 926-947, July 1992.

Rachid Alaoui is a professor at the ESTA (Ecole Supérieure de Technologies d'Agadir), Morocco. He received the BS degree in electronics engineering from ENSET, Rabat, Morrocco, in 1986, He received the MS degree in process engineering from ENSA, Agadir, Morrocco, in 2006, and the PhD degree in defensive software, from Ibn Zohr University, Agadir, Morrocco. He is a research associate at the Centre for Compatibility Electromagnetic of Integrated Circuits, Lattis, INSA, and Toulouse, France. His research interests include Fault-Tolerant computing, Reliability, and Processor Susceptibility.

Lahoussine Bouhouch is a professor at the ESTA (Ecole Supérieure de Technologies d'Agadir), Morocco. He holds a Masters Degree and the PhD in Electronic engineering, from the University of Nancy I, France, in 1985 and 1988 respectively. He also holds a state thesis in Electronic, from the University Ibn Zohr of Agadir, Morocco in 2007.

In research, it is interested in electromagnetic compatibility (EMC), while characterizing and by modelling the interaction between magnetic materials and the processors.

He works in close cooperation with the platform of research in EMC component of the LATTIS, INSA, Toulouse, France.

Fuzzy Systems Neural Networks and Makov Switching AR Model for Prediction of Exchange Rates

A.F.M. Khodadad Khan[1], Mohammed Anwer[2], Shipra Banik[3]

School of Engineering and Computer Science, Independent University, Bangladesh, Dhaka, Bangladesh

khoda@secs.iub.edu.bd[1]; manwer@secs.iub.edu.bd[2]; shiprabanik@yahoo.com.au[3]

Abstract

Many international agents (e.g. money managers, investment banks, investors, funds makers and others) are very concerned about predicted values of exchange rates because it often moves dustically and generally affects the profits. This paper forecasted the daily Bangladeshi and Canadian exchange rates for the period of October 1996 to January 2013. With attention paid to recently developed econometric noises, the widely-used forecasting model the fuzzy extension of artificial neural network is considered and compared results with the Markov switching autoregressive forecasting model. Root mean square error and correlation coefficient are used as an evaluation measures. It has been found that the fuzzy extension of the artificial neural network model is a superior predictor compared to the other selected predictor for the Bangladeshi series and the reverse observed for the Canadian series. It is believed that the findings of this paper will be helpful for multinational organizations wanting to make wise policy about these two country's exchange rates.

Keywords

Exchange Rate Dynamics; Time Series Prediction; Non-Linearities; Econometric Noises; Aritificial Intelligence; Markov Switching Model

Introduction

Rate at which one currency is converted into another currency (e.g. for the purpose of travel, for engaging in speculation or trading in the foreign exchange market and others) is known as exchange rate. There are a variety of factors (e.g interest rates, inflation, the state of politics and the economy in each country and others) generally influencing the exchange rates. For this reason, it often moves significantly and commonly affects the profits of Multinational Companies (MNCs) engaged in various international business activities. Thus, understanding and forecasting exchange rate tendencies are very important to make a wide range of decisions for MNCs (investors, money managers, investments banks, hedge funds and others).

A large number of research [(Kodogiannis and Lolis, 2002), (Kuan and Liu, 1995), (Lisi and Schiavo, 1999), (Zhang and Hu, 1998), (Ismail and Isa, 2006), (Ping-Feng et al, 2006), (Tae and Steurer, 1995), (Dueker and Neely, 2007), (Hung, 2007), (Engel and Hamilton, 1990)] have been published in literature to find an optimal (or nearly optimal) prediction model for the exchange rate series. Many researches [(Kodogiannis and Lolis, 2002), (Kuan and Liu, 1995), (Lisi and Schiavo, 1999), (Zhang and Hu, 1998), (Ismail and Isa, 2006), (Ping-Feng et al, 2006), (Tae and Steurer, 1995), (Jang, 1993), (Banik et al, 2009)] have shown that the behavior of this series cannot be modelled solely by the linear time series models (e.g. regression model, time series model proposed by (Box and Jenkins, 1970) and others) because exchange rates are quite complex, non-linear, and unconstant in nature. Thus, developing a model for forecasting needs an iterative process of knowledge discovery, system improvement through data mining as well as trial and error experimentation.

To address this problem, in recent years [(Kodogiannis and Lolis, 2002), (Kuan and Liu, 1995), (Lisi and Schiavo, 1999), (Zhang and Hu, 1998), (Ismail and Isa, 2006), (Ping-Feng et al, 2006), (Tae and Steurer, 1995), (Dueker and Neely, 2007), (Hung, 2007), (Engel and Hamilton, 1990)], an increasing interest of scholars have been witnessed modeling data as nonlinear models including: artificial intelligence (AI) model, fuzzy logic model, genetic algorithm model, hybridization of ANN and fuzzy system model, Markov switching (MS) model, conditional heteroskedastic models, rough set theory, ant colony method, bee colony method and others.

In literature, we have noticed that there is a growing interest in using AI models [(Kodogiannis and Lolis, 2002), (Kuan and Liu, 1995), (Lisi and Schiavo, 1999), (Zhang and Hu, 1998), (Ping-Feng et al, 2006), (Tae and Steurer, 1995), (Jang, 1993), (Banik et al, 2009) and many others] to forecast exchange rate series. The reason for this rising popularity is that these models pay particular attention to non-linearities and learning processes both of which can help companies to improve their predictions for complex variables. The most common AI model (e.g. fuzzy extension of aritificial neural network (ANN)) is particularly useful for future predictions for variables, which is subject to non-linearities. Although ANN based models are found to perform better compared to conventional statistical models, the main drawback of ANN models is that their prediction capabilities deteriorate over a short period of time especially when data are very much chaotic. It has been proposed by many authors [(Kodogiannis and Lolis, 2002), (Lisi and Schiavo, 1999), (Banik et al, 2009)] to overcome this drawback and develop a reliable prediction of time series data. We have chosen the fuzzy extension of AI model and compared our results with the widely used MS model [(Ismail and Isa, 2006), (Dueker and Neely, 2007), (Hung, 2007), (Hamilton, 1989)] to predict Bangladeshi (BER), and Canadian (CER) exchange rates series in order to see whether selected models can help companies to raise predictive power. By analyzing applied models validity and precision, our plan is to develop the model which can best predict the selected series.

A lot of works have been done in literature to predict exchange rate based on statistical and AI models. For example, (Ismail and Isa, 2006) used the MS model to predict the exchange rates for three ASEAN countries (Malaysia, Singapore and Thailand), and compared their results with the autoregressive (AR) model of order p. Results show that the MS model predicted well than the AR model. (Kodogiannis and Lolis, 2002) examined the forecasting ability of daily exchange rate values of the US Dollar vs. British Pound using the neuro-fuzzy systems and compared with various networks. They found that the neuro-fuzzy systems outperform the other networks. (Kuan and Liu, 1995) have investigated the forecasting ability of feed-forward and recurrent neural networks based Japanese yen, British pound, Canadian dollar, Deutsche mark etc. Their results show that the selected network models have significant market timing ability relative to the random walk model. (Zhang and Hu, 1998) have successfully used neural network for predicting British pound/US dollar exchange rates. Their results show that the neural network outperforms the linear model. (Tae and Steurer,1995) compared the Deutsche mark monthly and weekly exchange rates using the neural networks and the statistical model, leading to the result that the neural net outperforms the statistical model. (Lisi and Schiavo, 1999) investigated the monthly exchange rates of the four major European currencies from 1973 to 1995 using the neural network and compared with the random walk model, and got the results that neural network turn out to be statistically better than the random walk model.

The aim of this paper is to investigate whether our selected models can serve useful tools to describe the behavior of BER and CER series more efficiently. To our knowledge, with caring econometric noises forecasting selected BER and CER daily series under the powerful nonlinear models are yet considered in the existing literature which have considered this research in this paper. Thus it is believed that findings of the paper will be useful for those who are interested to make wise policies about the complex variables BER and CER. The paper is outlined as follows: section 2 describes about the data set and numerical properties. Considered forecasting models are reviewed in section 3. Section 4 contains experimental designs and findings. Concluding remarks and some future research plans are given in section 5.

Data, Forecasting Model, Numerical Properties and Econometric Noises

We considered the daily BER and CER series over the period of October 1996 to January 2013 for a total of 5934 observations. BER and CER are the local currency against the US dollar, collected from the site of http://www.oanda.com/currency/historical-rates/. The training and testing data sets and for understanding of changes, series are depicted in Fig. 1 (BER series) and Fig. 2 (CER series) respectively. It is very clear that there is an increasing trend w.r.t to time for BER and a decreasing trend for CER series. There have some reasons why these sorts of trends exist. See above website for details. It is also observed from these plots that the behaviors of series are non-linear, meaning that series can appear unconstant with moves that look chaotic. Some sort of non-linearity can also present in the selected series.

Numerical Properties

To understand behaviors of daily BER and CER series, summary statistics are reported in Table 1. It is noticed that exchange rate patterns for BER and CER do not follow the normal distribution. Since our series are time series, so we have selected most commonly used time series model, namely, the autoregressive (AR) model of order p. The model is defined for BER and CER series are as follows:

$$BER_t = \alpha + \beta t + \rho_i\, BER_{t-i} + e_t, \quad t = 1,2,3,\ldots,n \qquad (2.1)$$

$$CER_t = \alpha + \beta t + \rho_i\, CER_{t-i} + e_t, \quad t = 1,2,3,\ldots,n \qquad (2.2)$$

where α is an intercept, β is the deterministic trend, t is the time variable, ρ_i are the lag orders of the AR(p) model and $e_t \sim N(0,\sigma^2)$. The appropriate lags of the series are selected by the Bayesian Information Criterion (BIC). It is started by calculating BIC (e.g. Akaike Information Criterion(AIC), Sibata Information Criterion(SIC) and others can also be used) to find the number of lags used in AR process for BER and CER. To calculate BIC, we followed the following steps:

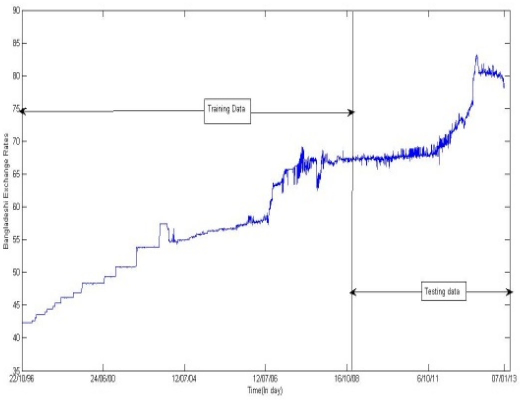

FIG. 1 TIME PLOTS OF BER SERIES

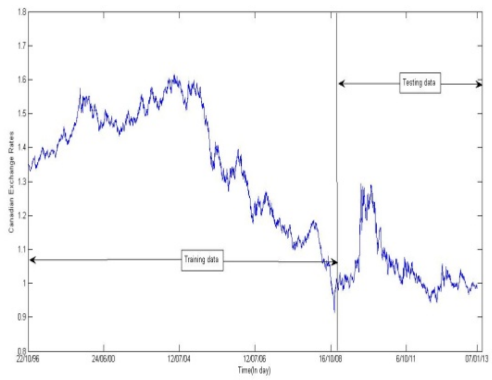

FIG. 2 TIME PLOTS OF CER SERIES

Various AR models have been utilized by increasing the order of AR, p. The formula is defined as

$$-2L + X_t \ln(n)$$

where n is the sample size, L is the maximized log-likelihood of the model and X_t includes an intercept and lags of series. Then we have chosen the model that gives the smallest value of BIC. It is noted that the lag length of series is selected based on a maximum lag of 8. It has been found that the number of lags to be used is 1 and 4 for the considered series. Based on this information, AR(1) for BER series and AR(4) model is considered for AR(4) for CER series. See Table 2 for proposed AR(p) model.

TABLE 1 DESCRIPTIVE STATISTICS OF BER AND CER FOR OCTOBER 1996 TO JANUARY 2013

Series	Statistical measures						
	n	Min Rate	Max rate	Mean Rate	SD Rate	Skew	Kur
BER	5934	42.35	83.15	60.38	10.09	0.077	2.216
CER	5934	0.914	1.615	1.269	0.208	0.008	1.516

TABLE 2 STATIONARITY TEST RESULTS

Rates	Proposed AR(p) Model	ADF(p) statistic	p-value (critical value) for ADF test	PP(L) statistic	p-value (critical value) for PP test
BER	AR(1)	-0.988(1)	0.141 (-3.414)	-1.13(8)	0.768 (-3.414)
CER	AR(4)	-1.233(4)	0.759 (-3.414)	-1.29(5)	0.2627 (-3.414)

Note: 'p' and 'L' indicate lag order to remove serial correlation.

Decision rule: If p-value<level of significance (α), then accept null hypothesis

Econometric Noises

Numerous studies (e.g. [(Nelson and Plosser, 1982), (Mitchell, 1999), (Banik,1999), (Banik and Silvapulle, 1999), (Said and Dickey, 1984), (Phillips and Perron, 1988)] and many others) have suggested that most time series are non-stationary (contains a unit root). Therefore, assumption of stationarity is unrealistic. Thus, prior to model specifications and estimations, stationary property of data series is routinely tested (see Banik,1999, details), otherwise the study can yield unrealistic results. That's why appropriate forecasting model should be selected for BER and CER series, and at first, stationarity property of the series was tested.

Many stationarity tests are available in time series literature (details, see [(Green, 2008), (Banik,1999)] and others). To test the stationarity behavior of our considered models (2.1)-(2.2), commonly applied unit root tests, namely the ADF test proposed by (Said and Dickey, 1984) and the PP test proposed by (Phillip and Perron, 1988)] are used (for test procedures, see (Banik,1999)). Note that under the null hypothesis of ADF and PP tests, series is assumed non-stationary and under the alternative hypothesis, series is stationary. Results (Table 2) show that BER and CER series are non-stationary (because p-value>α, α=0.05). Then we have taken the first difference of the series to remove non-stationarity and applied again ADF test and PP test. Our results show that in first differences considered BER and CER series are stationary (these results are not reported for spaces but are available on request). The effect of these tests will be shown when our forecasting model was used, in first differences.

Considered statistical tests ((Table 2) results show that BER and CER series are non-stationary and non-linear [Figs 1-2], respectively. To remove non-stationarity from selected series, we have used series in first differences. Thus, based on above findings, nonlinear forecasting models have been selected to forecast BER and CER which have also ability to capture chaotic behavior. A brief description of the considered forecasting models is given below.

Forecasting Models

Fuzzy Extension of Neural Network Model

Based on the theory of fuzzy set and logic, this architecture (also known as adaptive neural fuzzy inference system (ANFIS)) proposed by (Jang, 1993) is a combination of neural network (NN) system and fuzzy inference system (FIS) in such a way that the NN learning algorithm is used to determine the parameters of FIS. NN is non-linear statistical data modeling tool, which can capture and model any intput-output relationships. In addition, FIS, the process of formulating the mapping from a given input to an output using fuzzy logic, involves: membership functions (mfs), fuzzy logic operators and if-then-rules.

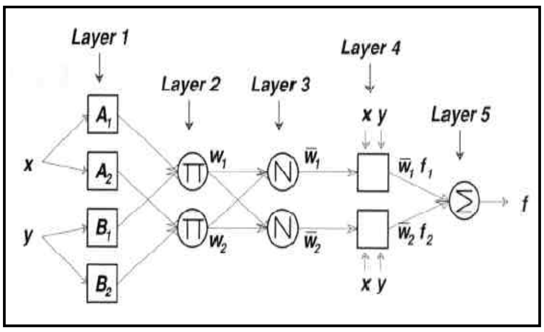

FIG. 3 AN ANFIS ARCHITECTURE WITH A 2-INPUT, 2-RULE FIRST-ORDER SUGENO MODEL

A typical ANFIS architecture is given in Fig. 3 showing that ANFIS has 1 input layer, 3 hidden layers (that represents mfs and fuzzy rules) and 1 output layer. It uses Sugeno-fuzzy inference model to be thelearning algorithm. In Fig. 3, the circular nodes represent fixed nodes whereas the square nodes are nodes that have parameters to be learnt. Each layer in this figure is associated with a particular step in FIS. The following concepts concerns the process, where the input vector is fed through the ANFIS network layer by layer:

Layer 1*: Fuzzy layer (generates mfs grades):* Input x to A_1 and A_2 and input y to B_1 and B_2 respectively, where A_1, A_2, B_1 and B_2 [fuzzy sets] are the lingiustic expressions which are used to distinguish the mfs represented by the premise parameters (PP). The relationship between the input-output mfs: $O^1_{A_i} = \mu_{A_i}(x)$; $O^1_{B_i} = \mu_{B_i}(y)$, i=1,2, where $\mu_{A_i}(x)$ and $\mu_{B_i}(y)$ denote mfs. Any continuous and piecewise differentiable functions such as generalised bell-shape mfs, gaussain-shaped mfs, triangular-shaped mfs, trapezoidal-shaped mfs can be used in this layer.

Layer 2*: Production layer (generate the firing strenghts):* It is marked as Π and output is defined as $O^2_i = W_i = \mu_{A_i}(x) \times \mu_{B_i}(y)$, where W_i, i=1,2 are weight functions of the next layer. Here the t-norm "product" is used as fuzzy operator.

Layer 3*: Normalized layer (normalize the firing strength):* It is marked as N and used to normalize W_i (i.e. it calculates the ratio of firing strength of the rules with the total firing strenghts). It is defined as: $O^3_i = \overline{W}_i = W_i / \sum_{i=1}^{2} W_i,\ i = 1, 2$.

Layer 4*: Defuzzy layer (calculates output rules):* Here an adaptive node $\overline{W}_i$ is outputs and $\{p_i, q_i, r_i\}$ is the parameter sets [known as consequent parameters (CP)] in this layer. The relationship between input and output is: $O^4_i = \overline{W}_i f_i = \overline{W}_i(p_i x + q_i y + r_i)$, i=1,2.

Layer 5*: Total output layer (sum all inputs from the layer 4):* Its node is marked as Σ, computes the overall output. It can be expressed as: O^5_i =Output =

$$\sum_{i=1}^{2} \overline{W}_i f_i = \overline{W}_i(p_i x + q_i y + r_i)$$

Next step is to examine how the ANFIS learns PP and CP for the mfs and the rules. From Fig. 3, it is clear that the final layer output can be expressed as a linear combination of the CP. In symbols, f = $(\overline{W}_1 x)p_1 + (\overline{W}_1 y)q_1 + (\overline{W}_1)r_1 + (\overline{W}_2 x)p_2 + (\overline{W}_2 y)q_2 + (\overline{W}_2)r_2$. Thus, we have a set of total ANFIS parameters S, which is calculated as S = S1$\cup$S2, where S1 = set of PP(nonlinear) and S2 = set of CP(linear). The learning algorithm of ANFIS is a hybrid algorithm, which combines the gradient descent (GD) method and the least square estimation (LSE) for an effective search of PP and CP, which means that ANFIS uses a two pass of learning algorithm to reduce the error: (i) Forward pass and (ii) backward pass. The hidden layer is computed by the GD method of the feedback structure and the final output is estimated by the LSE.

The output equation from the layer 5 can be rearranged as more usable form: Y = XW, where X=[$\overline{W}_1 x$ $\overline{W}_1 y$, $\overline{W}_1$, $\overline{W}_2 x$, $\overline{W}_2 y$, $\overline{W}_2$] and W = [p_1, q_1, r_1, p_2, q_2, r_2]^T. When the input-output training pattern exists, the vector W can be solved using the ANFIS learning algorithm .

The MS Model

Changes in regime happen quite suddenly in real world. For example, exchange rate appears to follow long swings, which means that rate drifts upward for a considerable period of time and then switches to a long period with a downward drift. To model this dramatic change, a more practical model is the Markov switching autoregressive (MS_AR) model, developed by (Hamilton, 1989). To understand the model clearly, consider the BER and CER at time t and S_t is an unobservable discrete state variable that takes values of 1 (appreciation period- an increase in the value of domestic currency relative to foreign currency) or 2 (depreciation period-a decrease in the value of domestic currency relative to foreign currency).

The MS_AR model with two possible states is defined as follows:

$$BER_t = \alpha_{S_t} X_t + e_t, \quad S_t \in \{1, 2\}, \ t = 1,2,\ldots,n$$

$$CER_t = \alpha_{S_t} X_t + e_t, \quad S_t \in \{1, 2\}, \ t = 1,2,\ldots,n$$

where X_t includes an intercept (denoted by μ_i, i =1,2) and lags of the dependent variable BER_t and CER, α_{S_t} are the corresponding parameters and $e_t \sim N(0, \sigma^2_{S_t})$ a random variable with a state dependent variance $\sigma^2_{S_t}$. The changes in states are rules by transition probabilities which are governed by a first order Markov process as follows:

$$P(S_t = 1 | S_{t-1} = 1) = p_{11}$$

$$P(S_t = 1 | S_{t-1} = 2) = p_{12}$$

$$P(S_t = 2 | S_{t-1} = 1) = p_{21}$$

$$P(S_t = 2 | S_{t-1} = 2) = p_{22}$$

with $p_{11} + p_{21}=1$ and $p_{12} + p_{22}=1$, where p_{ij}(i=1,2 and j=1,2) are the transition probabilities for switching from one state to other state. As S_t is unobserved, the parameter vector (say) $\theta = (\alpha_{S_t}, \sigma_1, \sigma_2, p_{11}, p_{12}, p_{21}, p_{22})$ is estimated by the maximum likelihood method using the EM algorithm developed by (Hamilton, 1989). Here fitted series will be calculated by the probability of S_t=1or 2 based on the observed series.

Experimentation, Results and Discussion

Experimentation

The first 70% observations for daily BER and CER series are used as the training period and the rest as the testing period [see Figs 1-2]. All computational works were carried out using the programming code of MATLAB.

Results

(1) The ANFIS Model

A trial and error approach is used to design the topology of ANFIS. The best performance is obtained by a network consisting of: 4 inputs with 2 mfs (type Gaussian-shaped) with each input, 8 if-then fuzzy rules were learned, total parameters (44)=premise parameters(12) + consequent parameters (32), where premise parameters are calculated by number of inputs×number of mfs×number of parameters of Gaussian distribution and consequent parameters is calculated by number of mfs×number of parameters of Gaussian distribution × number of fuzzy rules.

The training data has been used with the MATLAB command Genfis1 in order to create a FIS. Thus, the following ANFIS forecasting models was selected to forecast BER and CER values:

$$P_{5th\,day\_BER} = f(day1\_BER, day2\_BER, day3\_BER, day4\_BER)$$

$$P_{5th\,day\_CER} = f(day1\_CER, day2\_CER, day3\_CER, day4\_CER)$$

(2) The MS_AR Model

It is started by calculating BIC to find the number of lags used in AR process for BER and CER. It was found that the number of lags to be used is 1 and 4 for BER and CER. Based on this information, a MS with AR(1) for BER series and a MS with AR(4) model are considered for AR(4) and CER series and all parameters have been estimated using the maximum likelihood method.

TABLE 3A PARAMETERS ESTIMATES FOR THE MS_AR(1) MODEL FOR STATE 1 AND BER SERIES

	μ_1	σ_1	α_{11}
Parameters estimates	0.0020 (0.0005)	0.0324 (0.0030)	-0.1213 (0.0124)

Note: Standard errors are in parenthesis

TABLE 3B PARAMETERS ESTIMATES FOR THE MS_AR(1) MODEL FOR STATE 2 AND BER SERIES

	μ_2	σ_2	α_{21}
Parameters estimates	0.0041 (0.013)	0.0213 (0.0012)	0.2012 (0.0345)

Note: Standard errors are in parenthesis

TABLE 3C TRANSITION PROBABILITY MATRIX

$$\begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} = \begin{bmatrix} 0.46 & 0.35 \\ 0.54 & 0.65 \end{bmatrix}$$

TABLE 4A PARAMETERS ESTIMATES FOR THE MS_AR(4) MODEL FOR STATE 1 AND FOR CER SERIES

	μ_1	σ_1	α_{11}	α_{12}	α_{13}	α_{14}
Parameters estimates	0.0002 (0.0001)	0.0045 (0.0003)	-0.0065 (0.0245)	-0.3429 (0.0134)	-0.5468 (0.0312)	-0.1302 (0.0528)

Note: Standard errors are in parenthesis

TABLE 4B PARAMETERS ESTIMATES FOR THE MS_AR(4) MODEL FOR STATE 1 AND FOR CER SERIES

	μ_2	σ_2	α_{21}	α_{22}	α_{23}	α_{24}
Parameters estimates	0.0009 (0.002)	0.4138 (0.0112)	0.3000 (0.0720)	-0.0400 (0.1260)	0.0057 (0.0287)	0.0078 (0.0487)

Note: Standard errors are in parenthesis

TABLE 4C TRANSITION PROBABILITY MATRIX

$$\begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} = \begin{bmatrix} 0.78 & 0.29 \\ 022 & 0.71 \end{bmatrix}$$

Results are reported in Tables 3A-3C for BER series and Tables 4A-4C for CER series. These parameters estimates are used to forecast daily BER and CER.

Discussion of Results

Forecasting performances are evaluated against two widely used statistical metrics, namely, root mean square error (RMSE) and correlation coefficient (CORR). The formulas are defined as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathrm{Actual_i} - \mathrm{Predicted_i})^2}$$

$$\mathrm{CORR} = \frac{\mathrm{Cov(Actual - Predicted)}}{\sigma_{\mathrm{Actual}}\sigma_{\mathrm{Predicted}}}$$

Smaller values RMSE indicate higher day accuracy in forecasting. While higher CORR indicates better prediction.

To evaluate predicted and actual exchange rates, prediction performance is measured in terms of RMSE and CORR over the training and testing data. Tables 5 and 6 summarize the performances of considered forecasting models for BER series in Tables 7 and 8 for CER series. In terms of measures of RMSE and CORR (see Table 5), our training results show that the ANFIS forecasting model outperforms (noted smallest RMSE value and highest CORR value) the MS_AR(1) forecasting model. The highest CORR values (i.e. good match between actual and predicted BER) again indicated that the ANFIS forecasting model outperforms the MS_AR(1) model. For example, the accuracy of prediction for the ANFIS forecasting model is 95.46% and for the MS_AR(1) forecasting model 92.65% respectively. After the considered models have been built using the training data, BER series is forecasted over the testing data and performance measures are reported in Table 6. The testing results (see Table 6) show that BER prediction capability of ANFIS forecasting model is higher again (found the lowest RMSE value and highest CORR values) compared to the that of MS_AR(1) model.

TABLE 5 PERFORMANCE MEASURES FOR TRAINING DATA AND FOR BER SERIES

Performances metrics	ANFIS	MS_AR(4)
RMSE	1.2359	4.2180
CORR	0.9546	0.9265

TABLE 6 PERFORMANCE MEASURES FOR TESTING DATA AND FOR BER SERIES

Performances metrics	ANFIS	MS_AR(4)
RMSE	1.1045	3.8796
CORR	0.9672	0.9349

TABLE 7 PERFORMANCE MEASURES FOR TRAINING DATA AND FOR CER SERIES

Performances metrics	ANFIS	MS_AR(4)
RMSE	5.0301	2.5298
CORR	0.8994	0.9336

TABLE 8 PERFORMANCE MEASURES FOR TESTING DATA AND FOR CER SERIES

Performances metrics	ANFIS	MS_AR(4)
RMSE	4.6780	2.0970
CORR	0.9999	0.95365

In order to find out how well our considered models fitted to the actual CER data, Table 7 and Table 8 have been added. For training data, it shows clearly that the MS_AR(4) forecasting model outperforms (noted smallest RMSE value and highest CORR value) the ANFIS forecasting model. The testing results (see Table 8) show that CER prediction capability of MS_AR(4) forecasting model is higher again (found the lowest RMSE value and highest CORR value) compared to that of the ANFIS model, meaning more significant differences in the ANFIS model observed between the actual CER and that predicted by the ANFIS than in the MS_AR(4) forecasting model.

According to our findings, the ANFIS forecasting model appears to be more suitable for BER series modeling and the MS_AR forecasting model for CER series.

Conclusion

In this paper, the popular forecasting models (i.e. the ANFIS model and the MS_AR model) have been considered to predict daily BER and CER series for the period October 1998 to January 2013. The forecasting performances of selected models were measured by widely used measures RMSE and CORR. Our findings suggested that the ANFIS forecasting model can forecast the daily BER series closely as compare to the MS_AR forecasting model. The MS_AR forecasting model appeared to be more suitable for CER series modeling. It was believed that our findings will be useful for researchers in making wise decisions about BER and CER series.

Our next step is to improve forecasting results using recently widely used the rough set forecasting model, where stock data, market demand and supply data could be regarded as input variables to predict more accurately daily BER and CER. This is focus of future research.

ACKNOWLEDGEMENT

We greatly acknowledge comments and useful suggestions from anonymous referees. We are very thankful to the Editor (publications), Elva Grande, International Journal of Computer Science and Application for his very valuable cooperation. An earlier version of this paper is presented in 2013 IFSA World Congress NAFIPS Annual Meeting held in Edmonton, Canada, June 24-28.

REFERENCES

Banik, S. "Testing for Stationarity, Seasonality and Long Memory in Economic and Financial Time Series". Unpublished Ph.D. thesis, School of Business, La Trobe University, Australia, 1999.

Banik, S. and Silvapulle, P. "Testing for Seasonal Stability in Unemployment Series: International Evidence", Empirica, Springer, 26(2), 123-139, 1999.

Banik, S., Chanchary, F.H., Khan, K., Rouf, R.A. and Anwer, M. "Neural Network and Genetic Algorithm Approaches for Forecasting Bangladeshi Monsoon Rainfall". International Technology Management Review, vol. 2(1), 1-18, 2009.

Box, G.E.P. and Jenkins, G.M. (1970), *Time Series Analysis: Forecasting and Control*, San Francisco : Holden-Day.

Dueker, M. and Neely, C.J. "Can Markov Switching Models Predict Excess Foreign Exchange Returns?". Journal of Banking and Finance, vol.31, 279-296, 2007.

Engel, C. and Hamilton, J.D. "Long Swings in the Dollar: Are They in the Data and do Markets Know It?".American Economic Review, vol.80, 689-713, 1990.

Greene, W.H., Econometric Analysis, Prentice Hall, seventh edition, Upper Saddle River, NJ, 2008.

Hamilton, J.D. "A New Approach to the Economic Analysis of Non-Stationary Time Series and the Business Cycle". Econometrica, vol.57, 357-384, 1989.

Huang, Y. "The Persistence of Taiwan's Output Fluctuations: An Empirical Study Using Innovation Regime-Switching Model". Applied Economics, vol.39, 2673-2679, 2007.

Ismail, M.T. and Isa, Z. "Modeling Exchange Rates Using Regime Switching Model". Sains Malaysiana, vol.35, 55-62, 2006.

Jang, J.S.R. "ANFIS: Adaptive-network-based Fuzzy Inference Systems". IEEE Transactions on Systems, Man, and Cybernetics, 665-685, 1993.

Kodogiannis, V. and Lolis, A. "Forecasting Financial Time Series Using Neural Network and Fuzzy System-Based Techniques". Neural Computing and Applications, vol.11, 90-102, 2002.

Kuan, C.M. and Liu, T., "Forecasting Exchange Rates Usingfeed forward and Recurrent Neural Networks". Journal of Applied Econometrics, vol.10, 347-364, 1995.

Lisi, F. and Schiavo, R.A."A comparison between Neural Networks and Chaotic Models for Exchange Rate Prediction".Computational Statistics and Data Analysis, vol.30, 87-102, 1999.

Mitchell, W.F. "Testing for Unit Roots and Persistence in OECD Unemployment Rates". Applied Economics,vol.25, 1489-1501, 1999.

Nelson, C.R. and Plosser, C.I. "Trends and Random Walks in Macroeconomics Time Series". Journal of Monetary

Economics, 10, 139-162, 1982.

Phillips, P.C.B. and Perron, P., Testing for Unit Root in Time Series Regression, Biometrica, 79, 335-346, 1988.

Ping-Feng, W.C.H., Chih-Shen, P. L. and Chen, C.T. "A Hybrid Support Vector Machine Regression for Exchange Rate Prediction". Information and Management Sciences, vol.17, 19-32, 2006.

Tae, H.H. and Steurer, E. "Exchangr Rate Forecasting: Neural Networks vs. Linear Models Using Monthly and Weekly Data". Neurocomputing, vol.10, 323-339, 1995.

Said, S.E. and Dickey, D.A. "Testing for Unit Roots in Autoregressive Moving Average Models of Unknown Order", Biometrica, 71, 599-607,1984.

Zhang, G. and Hu, M.Y. "Neural Network Forecasting of the British Pound/US Dollar Exchange Rate". International Journal of Management Science, vol.26, 495-506, 1998.

A.F.M. Khodadad Khan obtained his M.S. in Mathematics from the University of Arizona, USA in 1975. Currently, he is a Professor at the School of Engineering and Computer Science, Independent University, Bangladesh.

Mohammed Anwer obtained his PhD in Mechanical Engineering from the Arizona State University, USA in 1989. Currently, he is a Professor at the School of Engineering and Computer Science, Independent University, Bangladesh.

Shipra Banik received her PhD in Applied financial Econometrics from the La Trobe University, Australia in 2000. Currently, she is an Associate Professor at the School of Engineering and Computer Science, Independent University, Bangladesh.