Journal of King Saud University – Computer and Information Sciences (2014) **26**, 7–18

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

ORIGINAL ARTICLE

Forecasting of currency exchange rates using an adaptive ARMA model with differential evolution based training

Minakhi Rout [a], Babita Majhi [b,c,*], Ritanjali Majhi [d], Ganapati Panda [e]

[a] *Dept. of CSE, ITER, Siksha O Anusandhan (Deemed to be) University, Bhubaneswar, India*
[b] *Dept. of Automatic Control and Systems Engineering, The University of Sheffield, UK*
[c] *Dept. of CSIT, G.G. Vishwavidyalaya (Central University), Bilaspur, India*
[d] *School of Management, National Institute of Technology, Warangal, India*
[e] *School of Electrical Sciences, Indian Institute of Technology, Bhubaneswar, India*

Received 20 July 2012; revised 4 December 2012; accepted 2 January 2013
Available online 8 January 2013

KEYWORDS

Exchange rate forecasting;
Adaptive auto regressive moving average (ARMA) model;
Forward backward LMS;
Particle swarm optimization (PSO);
Differential evolution (DE);
Cat swarm optimization (CSO) and bacterial foraging optimization (BFO)

Abstract To alleviate the limitations of statistical based methods of forecasting of exchange rates, soft and evolutionary computing based techniques have been introduced in the literature. To further the research in this direction this paper proposes a simple but promising hybrid prediction model by suitably combining an adaptive autoregressive moving average (ARMA) architecture and differential evolution (DE) based training of its feed-forward and feed-back parameters. Simple statistical features are extracted for each exchange rate using a sliding window of past data and are employed as input to the prediction model for training its internal coefficients using DE optimization strategy. The prediction efficiency is validated using past exchange rates not used for training purpose. Simulation results using real life data are presented for three different exchange rates for one–fifteen months' ahead predictions. The results of the developed model are compared with other four competitive methods such as ARMA-particle swarm optimization (PSO), ARMA-cat swarm optimization (CSO), ARMA-bacterial foraging optimization (BFO) and ARMA-forward backward least mean square (FBLMS). The derivative based ARMA-FBLMS forecasting model exhibits worst prediction performance of the exchange rates. Comparisons of different performance measures including the training time of the all three evolutionary computing based models demonstrate that the proposed ARMA-DE exchange rate prediction model possesses superior short and long range prediction potentiality compared to others.

1. Introduction

Accurate prediction of different exchange rates is important as substantial amount of trading takes place through the currency exchange market. The prediction is affected by economic and political factors and also involves uncertainty and nonlinearity. Thus accurate prediction of exchange rates is a complex task. In the literature many interesting publications on exchange rate prediction have been reported as detailed here.

* Corresponding author at: Dept. of Automatic Control and Systems Engineering, The University of Sheffield, UK. Mobile: +44 7778380647/+91 9437048906(Cell); fax: +91 674 2306203.
E-mail addresses: minakhi.rout@gmail.com (M. Rout), babita.majhi@gmail.com (B. Majhi), ritanjalimajhi@gmail.com (R. Majhi), gpanda@iitbbs.ac.in (G. Panda).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

1319-1578 © 2013 Production and hosting by Elsevier B.V. on behalf of King Saud University.
http://dx.doi.org/10.1016/j.jksuci.2013.01.002

Under such conditions the data/features driven forecasting approach has proven to be effective for different financial time series. In a recent paper (Yu et al., 2005) the authors have proposed an improved ensemble forecasting model for foreign exchange rates by integrating generalized linear autoregression and artificial neural network. In another communication (Zhang and Wan, 2007) the authors have developed a novel granular soft computing based forecasting approach to currency exchange rates. The experimental results demonstrate that the fuzzy interval neural network can provide more reliable prediction performance. Using a single layer low complexity nonlinear adaptive model (Majhi et al., 2009b) the authors have proposed an efficient scheme for the prediction of exchange rates between US Dollar and British Pound, Indian Rupees and Japanese Yen. They have also proposed another efficient prediction model by cascading two stages of single layer nonlinear networks. In another study, both parametric and nonparametric self organizing modeling methods have been applied for daily prediction of the American Dollar and the Deutche Mark against the British Pound (Anastasakis and Mort, 2009). They have reported that the combined approach is found to produce promising results. An hybrid model using the rough set theory (RST) and directed acyclic graph support vector machines (DAGSVM) have been suitably combined to analyze the exchange rates (Pai et al., 2010). They have found that the proposed method is a promising alternative for analyzing the exchange rates. Other structures which have been used for forecasting purpose are discussed in sequel.

The Box–Jenkins method using autoregressive moving average (ARMA) (Box and Jenkins, 1976) linear models have extensively been used in many areas of time series forecasting. A typical ARMA model consists of three steps: identification, parameter estimation and forecasting. Among these three steps, the identification step, which involves order determination of the AR and MA parts of ARMA model is important. This step requires statistical information such as the autocorrelation and partial autocorrelation (Box and Jenkins, 1976). The problem of estimating the order and the parameters of an ARMA model is still an active area of research (Rojasa et al., 2008).

In the past, statistical ARMA models have been developed and utilized successfully for analysis and simulation of strong earthquake ground motions (Popescu and Demetriu, 1990), time series forecasting (Chib and Greenberg, 1994; Lees and Matheson, 2007; Stoica, 1984; Poskitt, 2003), forecasting of work piece roundness error in turning operation (Fung and Chung, 1999), river flow (Mohammadi et al., 2006; Koutroumanidis et al., 2009; Kisi, 2010), Electricity load (Nowicka-Zagrajek and Weron, 2000; Pappas et al., 2008), electricity consumption (Taylor, 2006), tourism demand (Andrawis et al., 2011; Chu, 2008; Chu, 2009), hourly electricity price (Cuaresma et al., 2004), wind speed (Erdem and Shi, 2011), weather prediction and global radiation (Voyant et al., 2012), machine health condition (Pham and Yang, 2010), rotate speed signal of one type of aero-engine (Liu et al., 2011).

Variations of ARMA model such as the vector ARMA for forecasting of treasury bill rates and changes in money supply (Aksu, 1991) and seasonal fractionally differenced ARMA model for long range forecasting of revenue of IBM (Ray, 1993) have been reported in the literature. Multivariate ARMA model has been applied to model Canadian money, income and interest rate forecasting (Boudjellaba et al., 1994). In addition clustering of time series data has been attempted using the ARMA model (Xiong and Yeung, 2004).

Although the Box–Jenkins stochastic time series approach can provide accurate forecast results, these models are all based on fixed parameter design. Based on a set of historical data, the model structure as well as its parameters is determined and estimated. The fitted model is then used to forecast the future. In practical situations when new data are added, the parameters require re-estimation and hence this approach provides a limited forecasting accuracy (Chen et al., 1995). One major requirement of the ARMA model is that the time series must be linear and stationary (Wu and Chan, 2011). But real life time series data are nonlinear and non-stationary in nature. In the literature different hybrid ARMA methods have been proposed for forecasting purpose. Use of hybridization of autoregressive with exogenous input (NARX) with ARMA for machine state (Pham et al., 2010), ARMA and neural network for sunspot numbers and trend (Chattopadhyay et al., 2011), gray and ARMA for gyro drift (Zhou and Hu, 2008) and ARMA and TDNN for solar radiation (Wu and Chan, 2011) forecasting have been suggested. Fuzzy logic, artificial neural network (ANN) and ARMA models have been suitably combined for time series forecasting (Rojasa et al., 2008). The radial basis function neural network added with ARMA for time series forecasting has been proposed in (da Silva, 2008). Partially adaptive estimator of ARMA models has been developed (McDonald, 1989) including least absolute deviation and least squares criteria. An adaptive ARMA model for short range load forecasting has also been reported in (Chen et al., 1995).When a sample of a time series depends on present input as well as past outputs, the corresponding time series can be better modeled by a pole-zero or ARMA model. Such time series can also be modeled by conventional all zeros or finite impulse response (FIR) or non-recursive models. But the order of the corresponding model would be large and hence more computational complexity would be involved in training and running the model. For dynamic and nonlinear data the fixed ARMA model yields poor prediction performance as its previously estimated parameters do not perform well for the new situations. Thus, adaptive ARMA in which the parameters can be retrained is more suitable for such time series prediction. In the literature various forms of adaptive ARMA models have been suggested.

The forward and backward least mean square (FBLMS) algorithm and recursive least square (RLS) (Widrow and Strearns, 1985) algorithm have been used for obtaining ARMA model in an iterative manner. But these algorithms are derivative based and hence its parameters have a tendency to fall into the local minima solution (Widrow and Strearns, 1985). To avoid such situation adaptive ARMA models have been proposed to be trained using derivative free learning algorithms. In the recent past the Genetic algorithm (GA) has been employed to estimate the structure and parameters of ARMA model for time series forecasting (Flores et al., 2012) and the PSO-ARMA model has been suggested for sales forecasting (Majhi et al., 2009a). The use of GA in training of the parameters has certain deficiencies. The first one is the difficulty in choosing proper crossover and mutation probability. The increase in population size in a generation involves more computation. In binary GA, the conversion of chromosome values from binary to decimal for fitness evaluation also requires more time.

Recently a number of evolutionary computing techniques such as particle swarm optimization (PSO) (Kennedy et al., 2001), Differential Evolution (DE) (Storn and Price, 1995), Bacterial Foraging Optimization (BFO) (Passino, 2002) and Cat Swarm Optimization (CSO) (Chu and Tsai, 2007) have been successfully applied to many fields. Out of these algorithms the DE is found to be a simple and useful alternative to GA and has been observed to perform better for various applications such as parameter identification (Ursem and Vadstrup, 2003), image processing (Falco et al., 2006; Omran et al., 2005), data clustering (Paterlini and Krink, 2005), optimal designing (Babu and Munawar, 2007), scheduling (Nearchou and Omirou, 2006) and stock market prediction (Rout et al., 2011). In this paper an in depth investigation has been made for forecasting various exchange rates using adaptive ARMA as the basic architecture and DE as a training tool for updating the model parameters. The DE algorithm involves less computations compared to the GA, CSO and BFO algorithms. Further, it requires the choice of only two parameters which is relatively easier to set. Hence updating of the weights of the ARMA model by DE is advantageous compared to that performed by other bioinspired methods. For comparison purpose adaptive ARMA models are also trained using FBLMS, PSO, BFO and CSO algorithms under similar conditions. The paper has developed a promising forecasting model for prediction of exchange rates using DE based adaptive ARMA structure. The new model has been demonstrated to exhibit a superior exchange rate prediction performance compared to conventional FBLMS as well as bioinspired tools such as PSO, BFO and CSO based forecasting models.

The rest of the paper is organized as follows: Section 1 deals with literature review, formulation of the research problem and the motivation behind the proposed work. The adaptive ARMA based forecasting model is developed in Section 2. An introduction to differential evolution as a training algorithm is dealt in Section 3. The DE based adaptive ARMA forecasting model is designed in Section 4. The design of real life input data of the model and the details of simulation study are presented in Section 5. For assessing the potentiality of new model, its performance is compared with that obtained by FBLMS, BFO and CSO algorithms. Finally the conclusion of the paper is drawn in Section 6.

2. Adaptive auto regressive-moving average (ARMA) based forecasting model

The proposed adaptive ARMA model for prediction of a financial time series particularly various exchange rate predictions is shown in Fig. 1 in three stages. The first stage of development is the training phase in which the model parameters of ARMA are trained using Differential Evolution (DE) based optimization algorithm. The details of the training strategy have been depicted in Fig. 1(a). The ARMA prediction model essentially consists of feed-forward and feed-back linear combiners. The feed-forward portion acts as moving average (MA) or all-zero network whereas the feedback portion functions as an autoregressive (AR) or all-pole network. Thus the ARMA model contains both feed forward and feedback coefficients which need to be properly trained using appropriate learning algorithm. Conventionally in training an adaptive model, the raw time series data are directly used as input to the model. In many cases the raw data take more time to train the model as there is redundancy present in the data. Secondly proper training of the model is not achieved when the raw data are used as input and hence prediction performance becomes poor.

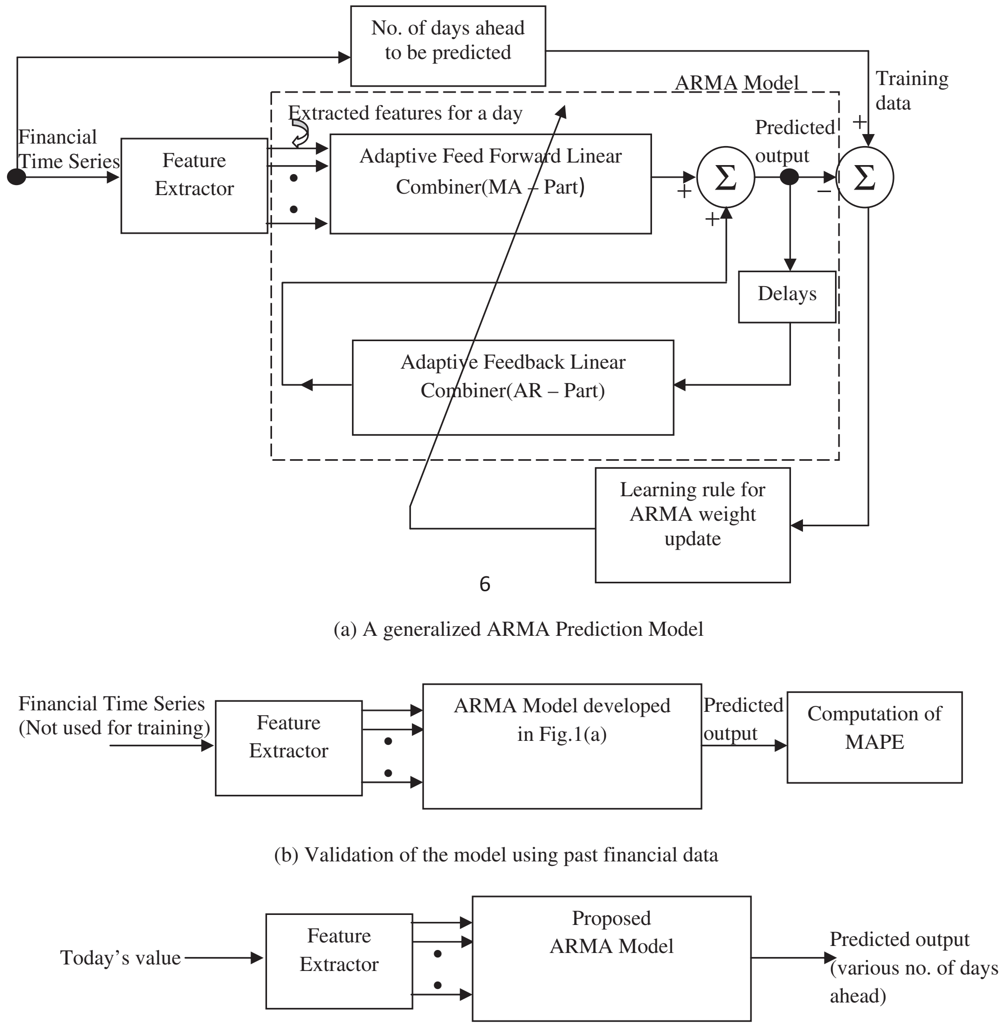
To alleviate these problems features are extracted from the financial time series and are used as input to the ARMA model. Further the future exchange rates not only depend on the features of the past data but also on the past predicted values. Hence a feed-forward and feed-back model like the ARMA has been chosen as the required network which possesses such feature. The amount of delays on the feed back side is suitably selected so as to provide the best possible prediction performance. The training sample is selected from the past time series depending on the number of days ahead the exchange rate to be predicted. The predicted exchange rate is compared with the training sample to produce the error or mismatch value. The feed forward and the feed back parameters are updated by a suitable learning rule such that in few iterations the cost function which is the mean square in this case progressively decreases and attains the least possible value.

Various learning rules have been reported in the literature. These can be broadly classified into two types: derivative based and derivative free. The derivative based class of learning algorithm like the FBLMS (Widrow and Strearns, 1985; Majhi and Panda, 2009) has the disadvantage of being trapped by local minima solution. In the recent past many evolutionary computing based learning algorithms such as the genetic algorithm (GA), the differential evolution (DE) etc. have been reported and extensively used for single and multi-objective optimization purposes. Out of the class of evolutionary computing algorithms, the DE is chosen because it is simple but powerful as well as it is computationally faster than the GA. The ARMA prediction model is considered as an adaptive optimizer in which the feed-forward and feed-back coefficients are suitably altered to minimize the squared error of the model. Then the DE is used as an efficient optimizer to reduce the mean square error to the least possible value.

After the training is complete the weights are frozen to their final values and the DE based ARMA model is ready for forecasting future exchange rate values when the desired features of present exchange rate are applied as input. But before it is used as an exchange rate predictor, its performance is validated. Referring to Fig. 1(b), the features of remaining 20% of old exchange rates are used as inputs and the model predicts the future exchange rate. Since these are past data, the desired exchange rates are known and hence the percentage of error is obtained in case of each input. Finally to have a consistent comparison of the prediction performance of various models the conventional mean average percentage of error (MAPE) is computed. The MAPE of the prediction model is computed according to

$$\mathrm{MAPE} = \left(\frac{\text{Sum of percentage of errors obtained by all test inputs}}{\text{No. of test inputs}}\right) \times 100$$

The MAPE is a fair indicator of a predictor model. When the designer is satisfied with the computed MAPE of the model then the model is subjected to prediction of various exchange rates. This situation is depicted in Fig. 1(c). The advantage of the adaptive prediction model is its flexibility. With little



(a) A generalized ARMA Prediction Model

(b) Validation of the model using past financial data

(c) ARMA model for prediction of time series data

Figure 1 Stages of development of ARMA prediction model for time series prediction.

effort the same model can be retrained to predict a different exchange rate as well as can be used for predicting exchange rate values for different days in future. This can be achieved by providing suitable input or desired values to the model during the training phase.

2.1. Actual ARMA model used for exchange rate prediction

In the previous subsection the basis of selection of the adaptive model and the evolutionary learning rules are discussed. Further it has dealt with the phases involved in achieving the final prediction model. In this subsection the details of the actual prediction model employed in this paper is dealt.

The block diagram of an adaptive ARMA based prediction model is shown in Fig. 2. The model is an adaptive pole-zero structure and is described by the recursive difference equation given in (1).

$$y(n) = \sum_{m=1}^{N-1} a_m(n)y(n-m) + \sum_{m=0}^{M-1} b_m(n)x(n,m) \tag{1}$$

where $x(n)$ and $y(n)$ represent the nth input pattern and output of the model respectively. The current estimated output $y(n)$ depends on the past estimated output samples $y(n-m)$, m = 1, 2, ...$N-1$ and the features $x(n, m)$ of the current financial input. The coefficients $\{a_m(n), b_m(n)\}$ are adjusted using some learning rules until the appropriate model is developed. $d(n)$ is the desired or target financial value. The pole and zero parameters of the ARMA model are a_m and b_m, respectively. Referring to Fig. 2, the predicted output, $y(n)$ is given by

$$y(n) = \left(\frac{\sum_{m=0}^{M-1} b_m(n)x(n,m)}{1 - A(n,z)x(n)} \right) \tag{2}$$

where

$$A(n,z) = \sum_{m=1}^{N-1} a_m(n)z^{-m}. \tag{3}$$

The output error is computed as $e(n) = d(n) - y(n)$ and is generated by subtracting the model output in (1) from the true value, $d(n)$. The weights of the ARMA model are updated iteratively using some learning algorithm to minimize the squared error value. The minimization process leads to optimum

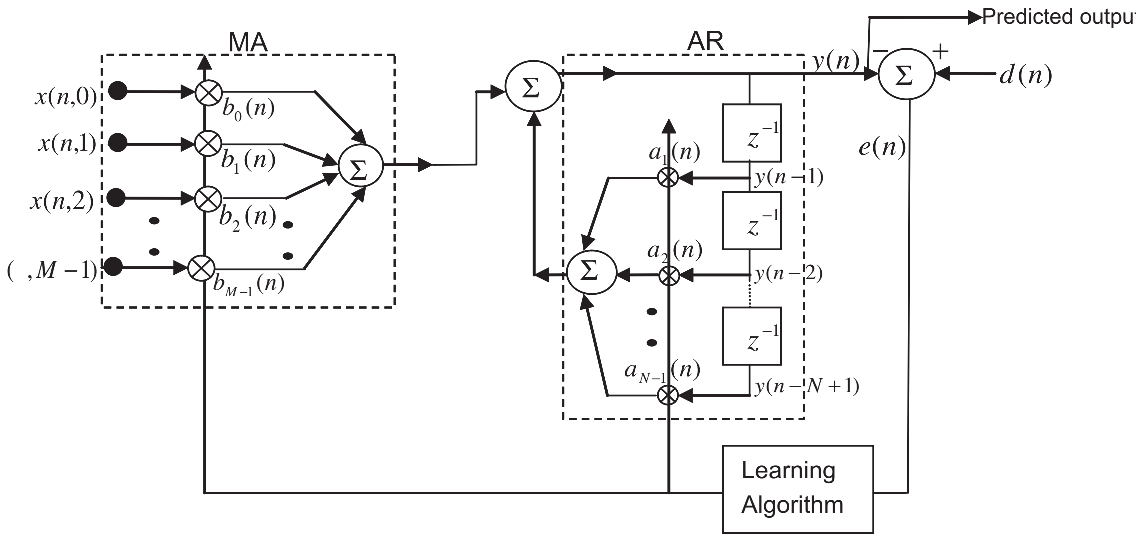


Figure 2 Adaptive ARMA based prediction model.

weights of the ARMA based prediction model. The feed forward and backward weights of the ARMA model are usually updated by the FBLMS algorithm given by (10). The aggregate coefficient vector is given as

$$\widehat{W}(n) = [\hat{b}_0(n) \ldots\ldots \hat{b}_{M-1}(n), \hat{a}_1(n), \ldots\ldots \hat{a}_{N-1}(n)]^T \tag{4}$$

The corresponding data vector is represented as

$$S(n) = [x(n,0), \ldots\ldots x(n, M-1), y(n-1), \ldots y(n-N+1)]^T \tag{5}$$

The output of the ARMA model at the nth iteration is

$$y(n) = W^T(n) * S(n) \tag{6}$$

The estimated gradient vector is given by

$$\hat{\nabla}(n) = -2(d(n) - y(n)) \times [\alpha_1(n) \ldots\ldots \alpha_{N-1}(n)\beta_0(n) \ldots\ldots \beta_{M-1}(n)] \tag{7}$$

where

$$\alpha_m(n) = \frac{\partial y(n)}{\partial a_m} = x(n, m) + \sum_{m=1}^{N-1} a_m(n)\,\alpha_m(n-m) \tag{8}$$

and

$$\beta_m(n) = y(n-m) + \sum_{m=0}^{M-1} b_m(n)\beta_m(n-m) \tag{9}$$

Finally the forward backward LMS (FBLMS) update algorithm is given by

$$W(n+1) = W(n) - \mu * \hat{\nabla}(n) \tag{10}$$

This update algorithm very often leads to non-optimum solution of weights. Hence in this paper, population based DE is employed to overcome this difficulty in proper training of the ARMA model. To compare the prediction performance of the proposed model particle swarm optimization (PSO), bacterial foraging optimization (BFO) and cat swarm optimization (CSO) algorithms based training schemes have been employed and the corresponding results have been obtained through simulation. In the next section a brief overview of DE is presented.

3. Introduction to differential evolution

Differential evolution (DE) (Storn and Price, 1995) is a population based stochastic meta-heuristic global optimization tool in continuous domains. Due to its simplicity, effectiveness and robustness, the DE has been successfully applied for solving complex optimization problems arising in different practical applications. A population in DE consists of P vectors represented as $\bar{x}_{i,G}, i = 1, 2, \ldots P$, where G is the number of generations. To keep the population within some bounds it is randomly initialized from a uniform distribution between the lower and the upper bounds defined for respective variables. These bounds are problem dependent. The possible solutions known as target vectors are represented with D -dimensional vectors as

$$\bar{x}_{i,G} = (x_{i,1,G}, x_{i,2,G}, \ldots, x_{i,D,G}) \tag{11}$$

The initial population is changed in each generation using subprocesses such as mutation, crossover and selection operators. In a simple DE algorithm mutant vector $\bar{v}$ for every target vector $\bar{x}_{i,G}$ is computed as

$$\bar{v}_{i,G} = \bar{x}_{r_1,G} + F(\bar{x}_{r_2,G} - \bar{x}_{r_3,G}), r_1 \neq r_2 \neq r_3 \tag{12}$$

where F is a mutation control parameter with its value between 0 and 2 and r_1, r_2 and r_3 are randomly chosen numbers within the population size. After mutation, the crossover operator generates a trial vector, $\bar{u}_{i,G}$ using (6)

$$u_{i,j,G} = \begin{cases} v_{i,j,G}, & if \quad rand_j \leqslant CR\, or\, j = rn(j) \\ & x_{i,j,G}, \ otherwise \end{cases} \tag{13}$$

where j, (dim *ension number*) = 1, 2,........, D; $rand_j$ a random number between 0 and 1; $rn(j)$ a randomly chosen index from 1, 2,......., D and CR the crossover constant between 0 and 1.

Differential evolution uses a greedy selection operator as

$$\bar{x}_{i,G+1} = \begin{cases} \bar{u}_{i,G}, & if \quad f(\bar{u}_{i,G}) < \quad f(\bar{x}_{i,G}) \\ & \bar{x}_{i,G}, \ otherwise \end{cases} \tag{14}$$

where $f(\bar{u}_{i,G})$ is the fitness value of the trial vector and $f(\bar{x}_{i,G})$ = fitness value of the target vector.

The number of generations is continued until the cost function almost remains constant and decreases further.

4. Development of DE based ARMA forecasting model

This section deals with the designing of DE based ARMA forecasting model. The ARMA model is constructed by considering it as a DE based optimization model in which the mean square error is minimized. Since ARMA model has a feedback path, it has a tendency to become unstable during training by conventional method. However, the DE based training overcomes this difficulty. The stepwise DE based weight update rule proceeds as follows:

1. The target vectors of DE are assumed to be the weights of the ARMA model. Let there be M target vectors each with D dimensions. Each time one vector is used as the initial value of the pole–zero parameters of the model.
2. The prediction model is fed with K input patterns successively. Each pattern has three independent values i.e. the mean, variance and actual exchange rate value corresponding to a month.
3. Each input component of input pattern is weighted with the zero-parameters, $b_m(n)$ to provide the output of the feed forward path. The output of the model, $y(n)$ is delayed, weighted with the pole parameters, $a_m(n)$ and added with the output of the feed forward path to give the final output of the ARMA model.
4. Each output, $y(n)$ is compared with the target value, $d(n)$ to give error value, $e(n)$. In this way after the application of all patterns K number of errors is obtained.
5. The fitness function which is the mean of squared error (MSE) of the pole-zero prediction model (corresponding to nth target vector) is calculated using (15)

$$MSE(n) = \frac{\sum_{j=1}^{K} e_j^2}{K} \tag{15}$$

1. The steps 2–5 are repeated for all target vectors and M numbers of MSE are generated. This completes one experiment and the Mean of MSE (MMSE) is calculated and used as the cost function to be optimized.
2. The elements of the target vector are then changed following mutation, crossover and selection processes as described in the previous section.
3. At the end of each generation the mean of MSE (MMSE) and the corresponding target vector are chosen. The relation between the number of generations and the MMSE is plotted to show the training characteristics of the model.
4. When the MMSE reaches the possible minimum value the training process is stopped.
5. The pole-zero parameters attained after training represent the coefficients of the ARMA based prediction model.

5. Simulation study

For simulation purpose real life data of three different exchange rates, Indian Rupees, British Pound and Japanese Yen have been collected for the period of 1-1-1973–1-10-2005, 1-1-1971–1-1-2005 and 1-1-1971–1-1-2005, respectively from the website www.forecasts.org. The data show the average of daily figures (noon buying rates in New York City) on the 1st day of each month. The numbers of data are 393, 418 and 418 for Rupees, Pound and Yen, respectively. Each set of data is normalized to lie between 0 and 1 by dividing each value of a set by the maximum value of the corresponding set. An initial window of size 12 containing the present and previous 11 data is used. The normalized value of 12th number data, the mean and variance of each group of 12 data are calculated and used as first input pattern of features. Subsequently the sliding window is shifted by one position to extract the second input pattern. A window size of 12 is chosen as it provides the best performance in the simulation experiment. This process is then repeated until all features are extracted. In this way a total of 382 feature patterns for Rupees and 407 patterns for each of Pound and Yen are extracted. Out of these patterns 80% are used for the training purpose and the remaining are used for validation of the model. The ARMA prediction model shown in Fig. 2 is used for simulation to assess its prediction performance.

The target vectors are initialized as the random numbers lie between 0 and 1. Since each input pattern has three features the number of weights of MA part is three. The number of weights of AR part is also taken as three after various trials as this combination provides the best possible prediction results. Each target vector of DE based ARMA has a total of six dimensions and its population size is 30. The other simulation parameters used for DE, PSO, BFO and CSO algorithms are given in Table 1. The convergence coefficient used in the FBLMS model is set at 0.05.

The training patterns are applied in sequence as input to the ARMA model, the corresponding outputs are obtained from the model and the resulting error values are recorded. The weights of the model are updated using the DE rule described in Section 4 until the minimum MMSE is reached. The MMSE

Table 1 Value of different parameters of algorithms used in simulation.

DE	PSO	BFO	CSO
Population size = 30	Population size = 30	Population size = 8–16	Population size = 30
$F = 0.9$	$c_1 = 1.042$	Probability of elimination–dispersion = 0.25	Seeking memory pool(SMP) = 5
$CR = 0.9$	$c_2 = 1.042$	Run length unit = 0.0075	Seeking range of selected dimension(SRD) = 0.2
Max. Iterations = 500	Inertia weight, (w) = linearly decreases between 0.9 and 0.4	Swimming length = 3	Counts of dimensions to change(CDC) = 0.8
Ensample average = 10	$v_{\max} = 1$	No. of chemotactic loops = 5	Mixture ratio(MR) = 0.1
	Max. Iterations = 500	No. of reproduction loops = 100–140	$c_1 = 2.0$
	Ensample average = 10	No. of elimination-dispersion loops = 5	$v_{\max} = 3.0$
		Max. Iterations = 500	Inertia weight, (w) = linearly decreases between 0.9 and 0.4
		Ensample average = 10	Max. Iterations = 500
			Ensample average = 10

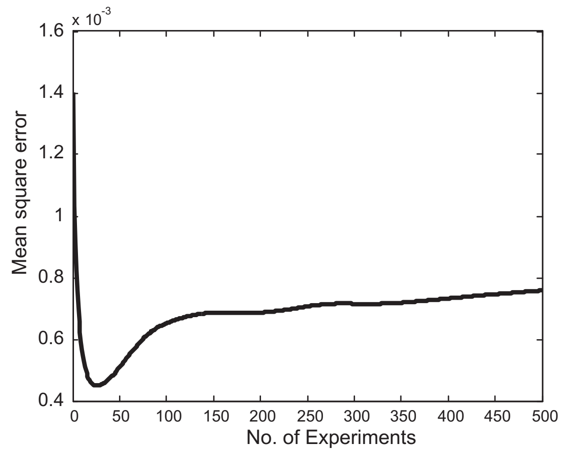


Figure 3a Convergence characteristics of ARMA-LMS for Rupees Exchange rate for 12 months ahead prediction.

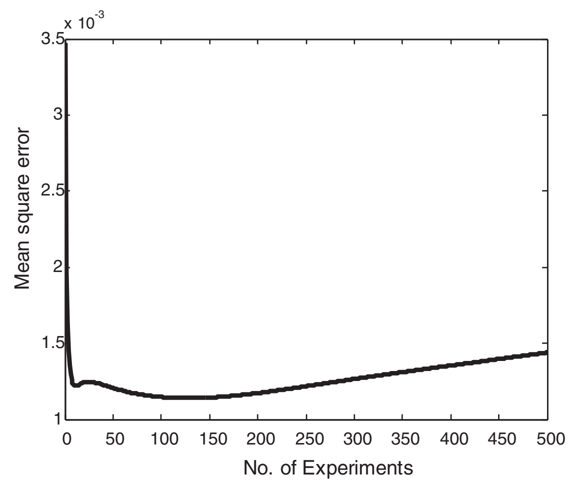


Figure 3b Convergence characteristics of ARMA-LMS for Pound Exchange rate for 12 months ahead prediction.

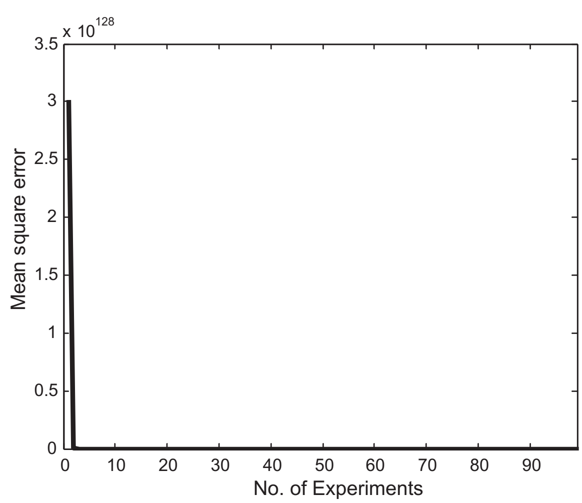


Figure 4a Convergence characteristics of ARMA-DE for Rupees Exchange rate for 12 months ahead prediction.

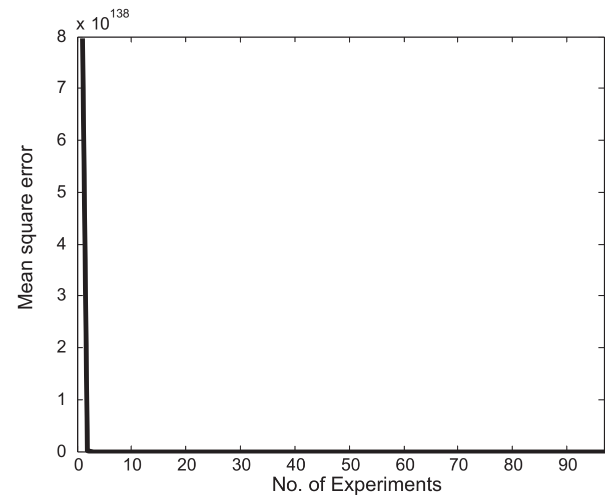


Figure 4b Convergence characteristics of ARMA-DE for Pound Exchange rate for 12 months ahead prediction.

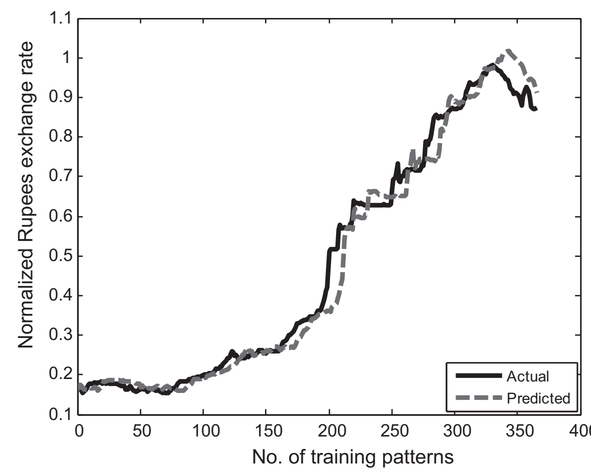


Figure 5a Comparison of actual and predicted values of Rupees exchange rate for 12 months ahead prediction using ARMA-DE during training.

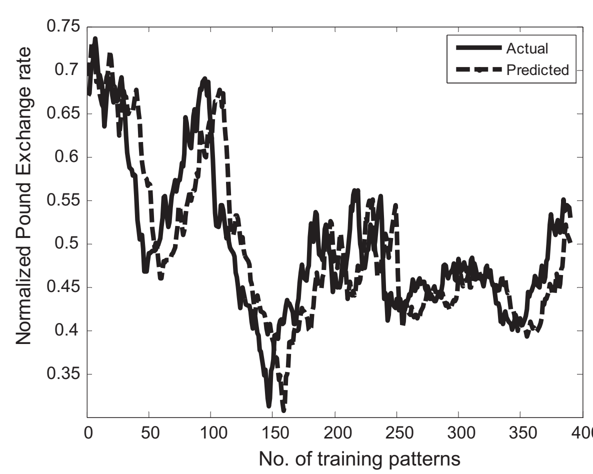


Figure 5b Comparison of actual and predicted values of Pound exchange rate for 12 months ahead prediction using ARMA-DE during training.

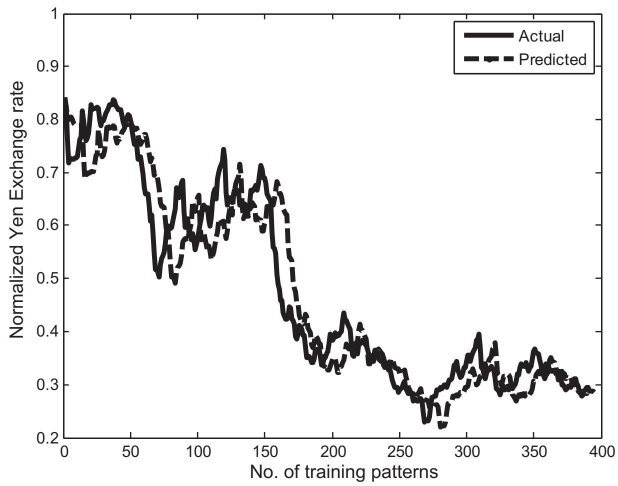


Figure 5c Comparison of actual and predicted values of Yen exchange rate for 12 months ahead prediction using ARMA-DE during training.

obtained from all the four evolutionary computing based models are given in Tables 4, 7 and 10 for rupees, pound and yen exchange rates, respectively for 1, 3, 6, 9, 12 and 15 months ahead prediction. Figs. 3a and 3b show the convergence characteristics of ARMA-FBLMS forecasting models for rupees and pound exchange rate prediction for 12 months' ahead respectively. The identical convergence characteristics for ARMA-DE model are depicted in Figs. 4a and 4b. From these plots it is observed that the FBLMS based training model shows divergence of the mean square error. Thus such model cannot be used for the purpose of exchange rate prediction. On the other hand the proposed DE training based ARMA prediction model exhibits excellent and fast convergence characteristics even for 12 months ahead prediction. To assess the training behavior of the ARMA-DE model, the matching performance is obtained during simulation and is plotted in Figs. 5a–5c for rupees, pound and yen exchange rates, respectively. Excellent agreement is observed in both cases even for 12 months' ahead prediction. After the MMSE reached its prefixed minimum value the training process is stopped and the test patterns are then applied for the validation of the ARMA prediction model. The performance of the model is evaluated by calculating few performance measures such as the Mean average percentage error (MAPE) and Root mean square error (RMSE). These are defined as

$$MAPE = \left(\frac{1}{N}\sum_{n=1}^{N}(A_n - P_n)/A_n \times 100\right) \quad (16)$$

$$RMSE = \sqrt{MSD} \quad (17)$$

where

$$MSD = \left(\frac{1}{N}\sum_{n=1}^{N}(A_n - P_n)^2\right) \quad (18)$$

where A_n = actual exchange rate, P_n = predicted exchange rate and N = No. of patterns applied for validation.

Comparison of the MAPE value of ARMA-DE and ARMA-LMS models for different exchange rates for various months ahead predictions is given in Table 2. Comparison of the MAPE and RMSE of different models using derivative free algorithms for various months ahead predictions is given in Tables 3, 6 and 9 for rupees, pound and yen respectively. The comparison of computation times is also presented in Tables 5, 8 and 11 respectively. From these tables it is observed that the proposed DE-ARMA model outperforms all other models based on PSO, BFO and CSO algorithms.

Some critical observations on the simulation results are presented to assess the efficiency of DE-ARMA based exchange rate predictor. Results obtained from four different models

Table 2 Comparison of MAPE value of different exchange rates between ARMA-LMS and ARMA-DE.

Months ahead	Dollar to rupees		Dollar to pound		Dollar to yen	
	ARMA-FBLMS	ARMA-DE	ARMA-FBLMS	ARMA-DE	ARMA-FBLMS	ARMA-DE
1	1.7624	**0.7984**	2.0403	**1.8197**	1.8505	**1.3973**
3	4.6913	**2.4515**	4.8753	**2.7962**	4.3914	**3.1382**
6	10.1602	**4.5135**	7.6012	**4.8292**	6.5531	**4.9493**
9	13.4810	**6.5204**	11.0738	**4.3443**	8.2282	**4.9442**
12	18.4978	**6.6273**	15.0995	**2.6389**	9.9773	**5.1621**
15	28.8139	**5.9148**	25.2697	**2.0328**	17.7972	**7.3081**

Table 3 Comparison of MAPE and RMSE for dollar to rupees exchange rate using derivative free algorithms.

No. of months ahead prediction	ARMA-PSO		ARMA-DE		ARMA-BFO		ARMA-CSO	
	MAPE (%)	RMSE	MAPE (%)	RMSE	MAPE (%)	RMSE	MAPE (%)	RMSE
1	0.8576	0.4928	**0.7984**	0.5115	1.0942	0.8124	2.5747	2.0124
3	3.8594	2.7789	**2.4515**	1.3903	3.5218	2.4895	3.5284	2.4433
6	5.8828	4.6984	**4.5135**	2.3095	5.8095	3.7736	4.6657	2.5142
9	9.5470	3.2752	**6.5204**	3.0937	6.5774	3.2013	6.5438	3.1186
12	9.0676	3.8112	**6.6273**	3.2951	8.1749	3.7562	6.9796	3.3876
15	8.7612	2.9599	**5.9148**	2.9364	8.6512	2.9329	8.4575	2.6407

Table 4 Comparison of MMSE obtained for dollar to rupees exchange rate.

No. of months ahead prediction	ARMA-PSO	ARMA-DE	ARMA-BFO	ARMA-CSO
1	7.0896×10^{-5}	7.4027×10^{-5}	6.5827×10^{-5}	7.4407×10^{-5}
3	2.6684×10^{-4}	2.5384×10^{-4}	2.5684×10^{-4}	2.6932×10^{-4}
6	6.3039×10^{-4}	5.3490×10^{-4}	5.9341×10^{-4}	6.0978×10^{-4}
9	0.0010	0.0007	0.0009	0.0009
12	0.0017	0.0014	0.0015	0.0015
15	0.0024	0.0020	0.0022	0.0022

Table 5 Comparison of computation time for dollar to rupees exchange rate.

No. of months ahead prediction	ARMA-PSO	ARMA-DE	ARMA-BFO	ARMA-CSO
1	1.1019	**0.9888**	4.1790	9.0299
3	1.1484	**1.0806**	4.3074	5.2927
6	1.1396	**1.0910**	4.2526	5.3096
9	1.1344	**1.0863**	4.2240	5.0855
12	1.1443	**0.9945**	4.2040	5.2688
15	1.1401	**0.9945**	4.1998	5.1570

Table 6 Comparison of MAPE and RMSE for dollar to pound exchange rate using derivative free algorithms.

No. of months ahead prediction	ARMA-PSO		ARMA-DE		ARMA-BFO		ARMA-CSO	
	MAPE (%)	RMSE	MAPE (%)	RMSE	MAPE (%)	RMSE	MAPE (%)	RMSE
1	**1.7567**	0.0372	1.8197	0.0384	1.8800	0.0392	1.8063	0.0385
3	3.9496	0.0879	**2.7962**	0.0653	3.2732	0.0764	3.2637	0.0748
6	5.5479	0.1152	**4.8292**	0.0940	5.4289	0.1143	5.4074	0.1134
9	6.1947	0.1383	**4.3443**	0.0892	6.0138	0.1370	5.3664	0.1157
12	4.2578	0.1078	**2.6389**	0.0666	3.5312	0.0974	3.0142	0.0971
15	3.9897	0.0818	**2.0328**	0.0383	2.9282	0.0654	2.5904	0.0651

Table 7 Comparison of MMSE obtained for dollar to pound exchange rate.

No. of months ahead prediction	ARMA-PSO	ARMA-DE	ARMA-BFO	ARMA-CSO
1	1.5225×10^{-4}	1.5046×10^{-4}	1.4567e-004	1.4537e-004
3	6.8946×10^{-4}	6.7141×10^{-4}	6.7786×10^{-4}	6.7497×10^{-4}
6	0.0015	0.0015	0.0015	0.0015
9	0.0022	0.0021	0.0022	0.0022
12	0.0030	0.0030	0.0030	0.0030
15	0.0039	0.0039	0.0039	0.0039

Table 8 Comparison of computation time for dollar to pound exchange rate.

No. of months ahead prediction	ARMA-PSO	ARMA-DE	ARMA-BFO	ARMA-CSO
1	1.1445	**1.0795**	4.4931	5.8285
3	1.1253	**1.1006**	4.5282	5.7554
6	1.1331	**1.0600**	4.4559	5.7341
9	1.1294	**1.0582**	4.4699	5.7928
12	1.2095	**1.0517**	4.5214	5.7733
15	1.1539	**1.1391**	4.5076	5.7822

for test data for rupee, pound and yen are shown in Figs. 6–8 indicate that the DE based predictors offer more accurate exchange rates compared to that of others. Further the proposed model predicts better exchange rates of rupees and pound compared to that of yen. Therefore to achieve improved performance of yen exchange rate alternative features need to be extracted from the time series and then applied to the model. Analyzing Tables 3–11, it is observed that in terms of all three

Table 9 Comparison of MAPE and RMSE for dollar to yen exchange rate using derivative free algorithms.

No. of months ahead prediction	ARMA-PSO		ARMA-DE		ARMA-BFO		ARMA-CSO	
	MAPE (%)	RMSE	MAPE (%)	RMSE	MAPE (%)	RMSE	MAPE (%)	RMSE
1	1.4585	1.9890	**1.3973**	1.9496	1.5987	2.1198	1.8256	2.6370
3	3.6430	3.9984	**3.1382**	3.8193	3.6389	3.9757	3.5917	3.8789
6	5.6706	6.8207	**4.9493**	6.2901	5.4101	6.7825	5.3999	6.5413
9	5.8112	8.6761	**4.9442**	7.3673	5.7623	8.2542	5.6617	8.0092
12	6.8786	7.5996	**5.1621**	6.5777	6.0299	7.5533	5.1708	6.5935
15	8.2968	7.9965	**7.3081**	6.6578	8.2303	7.9876	7.6802	7.9706

Table 10 Comparison of MMSE obtained for dollar to yen exchange rate.

No. of months ahead prediction	ARMA-PSO	ARMA-DE	ARMA-BFO	ARMA-CSO
1	1.9201×10^{-4}	1.9799×10^{-4}	1.8856×10^{-4}	1.9353×10^{-4}
3	0.0008	0.0008	0.0008	0.0008
6	0.0019	0.0019	0.0019	0.0019
9	0.0029	0.0029	0.0029	0.0029
12	0.0040	0.0038	0.0039	0.0039
15	0.0050	0.0050	0.0050	0.0050

Table 11 Comparison of computation time for dollar to yen exchange rate.

No. of months ahead prediction	ARMA-PSO	ARMA-DE	ARMA-BFO	ARMA-CSO
1	1.1448	**1.0616**	8.9155	5.8347
3	1.2225	**1.0332**	9.0179	5.8776
6	1.1710	**1.1154**	9.0930	5.8872
9	1.1196	**1.0241**	9.0663	5.8994
12	1.1918	**1.1307**	10.8738	5.8396
15	1.1076	**1.1066**	10.8010	5.9049

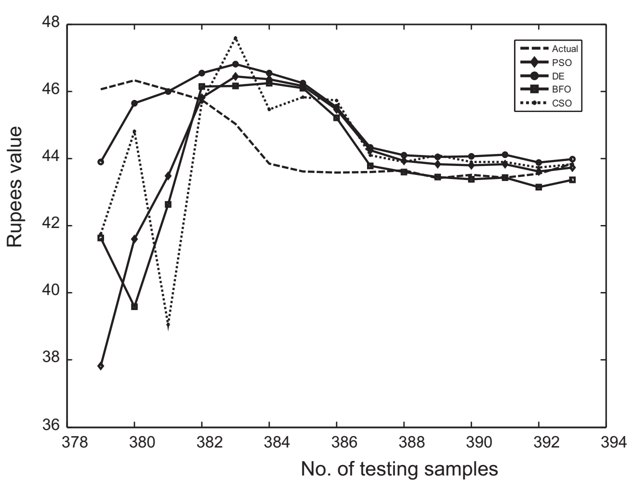


Figure 6 Comparison of actual and predicted values for dollar to rupees exchange rates for 3 months ahead prediction during testing.

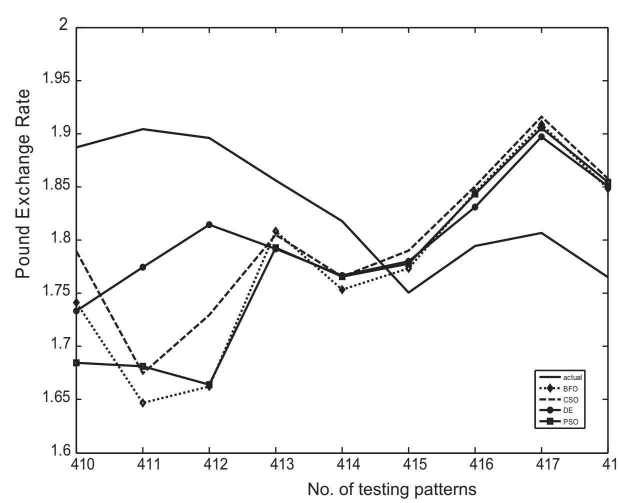


Figure 7 Comparison of actual and predicted values for dollar to pound exchange rates for 9 months ahead prediction during testing.

measures the DE-ARMA shows a superior performance compared to those achieved by other three models. Thus considering all aspects the exchange rate prediction models can be ranked in sequence as ARMA-DE, ARMA-CSO, ARMA-BFO and ARMA-PSO. Another interesting observation marked is on the computational time required for the training of various models. The results presented in Tables 5, 8 and 11 show that the proposed DE based ARMA takes the

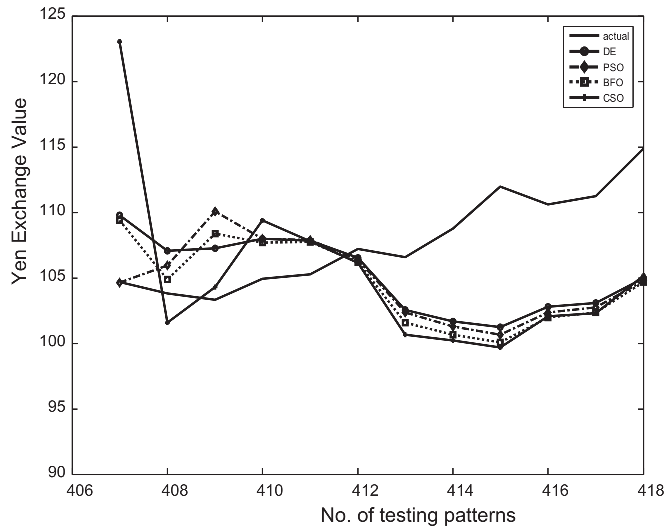


Figure 8 Comparison of actual and predicted values for dollar to yen exchange rates for 6 months ahead prediction during testing.

least time for training followed by ARMA-PSO, ARMA-BFO and finally ARMA-CSO. Thus through various simulation studies it is demonstrated that the proposed ARMA-DE combination based prediction model outperforms all other similar hybrid models studied in this paper.

6. Conclusion

The paper has developed an efficient exchange rate prediction scheme using an ARMA structure and DE based adaptive parameter update strategy. The prediction performance of rupees, yen and pound exchange rates with respect to US dollar of the new model has been evaluated. It is shown that the proposed model offers the best performance for predicting exchange rates compared to those offered by other three similar models studied. The FBLMS based model is observed to show worst prediction performance as the corresponding weight update mechanism is unstable and results in divergent learning characteristics. To further enhance the forecasting performance, particularly for a long range prediction it is suggested to use other additional hidden features of the financial time series as input to the model as well as to explore the use of other promising adaptive models. To enable satisfactory prediction when abrupt fluctuations of exchange rate take place due to political turmoil of a country, natural hazards or such unforeseen reasons, more in-depth investigation is required in terms of selection of features, model and learning algorithm. Our future study will focus on these critical issues in developing the prediction models.

References

Aksu, Celal, 1991. Forecasting with vector ARMA and state space methods. International Journal of Forecasting 7, 17–30.

Anastasakis, L., Mort, N., 2009. Exchange rate forecasting using a combined parametric and nonparametric self-organising modelling approach. Expert systems with applications 36, 12001–12011.

Andrawis, Robert R., Atiyaa, Amir F., El-Shishiny, Hisham, 2011. Combination of long term and short term forecasts with application to tourism demand forecasting. International Journal of Forecasting 27, 870–886.

Babu, B.V., Munawar, S.A., 2007. Differential evolution strategies for optimal design of shell-and-tube heat exchangers. Chemical Engineering Science 62 (14), 2739–3720.

Boudjellaba, Hafida, Dufour, Jean-Marie, Roy, Roth, 1994. Simplified conditions for noncausality between vectors in multivariate ARMA models. Journal of Econometrics 63, 271–287.

Box, G.E.P., Jenkins, G.M., 1976. Time Series Analysis – Forecasting and Control. Holden-Day Inc., San Francisco.

Chattopadhyay, Surajit, Jhajharia, Deepak, Chattopadhyay, Goutami, 2011. Trend estimation and univariate forecast of the sunspot numbers: development and comparison of ARMA, ARIMA and autoregressive neural network models. Comptes Rendus Geoscience 343, 433–442.

Chen, Jiann-Fuh, Wang, Wei-Ming, Huang, Chao-Ming, 1995. Analysis of an adaptive time series auto regressive moving average (ARMA) model for short term load forecasting. Electric Power Systems Research 34, 187–196.

Chib, S., Greenberg, E., 1994. Bays inference in regression models with ARMA (p, q) errors. Journal of Econometrics 64, 183–206.

Chu, Fong-Lin, 2008. A fractionally integrated autoregressive moving average approach to forecasting tourism demand. Tourism Management 29, 79–88.

Chu, Fong-Lin, 2009. Forecasting tourism demand with ARMA-based methods. Tourism Management 30, 740–751.

Chu, S.C., Tsai, P.W., 2007. Computational intelligence based on the behavior of cats. International Journal of Innovative Computing, Information and Control 3, 163–173.

Cuaresma, Jesús Crespo, Hlouskova, Jaroslava, Kossmeier, Stephan, Obersteiner, Michael, 2004. Forecasting electricity spot-prices using linear univariate time-series models. Applied Energy 77, 87–106.

da Silva, Carlos Gomes, 2008. Time series forecasting with a nonlinear model and the scatter search meta-heuristic. Information Sciences 178, 3288–3299.

Erdem, Ergin, Shi, Jing, 2011. ARMA based approaches for forecasting the tuple of wind speed and direction. Applied Energy 88, 1405–1414.

Falco, I. De, Cioppa, A. Della and Tarantino, A., 2006. Automatic classification of hand segmented image parts with differential evolution. In: Rothlauf F., et al., (Eds.), Evo Workshops, LNCS 3907, pp. 403-414.

Flores, Juan J., Graff, Mario, Rodriguez, Hector, 2012. Evolutive design of ARMA and ANN models for time series forecasting. Renewable Energy 44, 225–230.

Fung, Eric H.K., Chung, Allison P.L., 1999. Using ARMA models to forecast work piece roundness error in a turning operation. Applied Mathematical Modelling 23, 567–585.

Kennedy, J., Eberhart, R.C., Shi, Y., 2001. Swarm Intelligence. Morgan Kaufmann Publishers, San Francisco.

Kisi, O., 2010. Wavelet regression model for short term stream flow forecasting. Journal of Hydrology 389, 344–353.

Koutroumanidis, T., Sylaios, G., Zafeiriou, E., Tsihrintzis, V.A., 2009. Genetic modelling for the optimal forecasting of hydrologic time series: application in Nestos river. Journal of Hydrology 368, 156–164.

Lees, K., Matheson, T., 2007. Mind your ps and qs! improving ARMA forecasts with RBC priors. Economics Letter 96, 275–281.

Liu, Zhi-guo, Cai, Zeng-jie, Tan, Xiao-ming, 2011. Forecasting research of Aero-engine rotate speed signal based on ARMA model. Procedia Engineering 15, 115–121.

Majhi, B., Panda, G., 2009. Identification of IIR systems using comprehensive learning particle swarm optimization. International Journal of Power and Energy Conversion 1 (1), 105–124.

Majhi, R., Majhi, B., Rout, M., Mishra, S., Panda, G. 2009. Efficient sales forecasting using ARMA-PSO model. In: Proc. of IEEE International Conference on Nature and Biologically Inspired, Computing, pp. 1333–1337.

Majhi, R., Panda, G., Sahoo, G., 2009b. Efficient prediction of exchange rates with low complexity artificial neural network models. Expert systems with applications 36, 181–189.

McDonald, James B., 1989. Partially adaptive estimation of ARMA time series models. International Journal of Forecasting 5, 217–230.

Mohammadi, K., Eslami, H.R., Kahawita, R., 2006. Parameter estimation of ARMA model for river flow forecasting using goal programming. Journal of Hydrology 331, 293–299.

Nearchou, A.C., Omirou, S.L., 2006. Differential evolution for sequencing and scheduling optimization. Journal of Heuristics 12, 395–411.

Nowicka-Zagrajek, J., Weron, R., 2000. Modeling electricity loads in California: ARMA models with hyperbolic noise. Signal Processing 82, 1903–1915.

Omran, M.G.H., Engelbrecht, A.P., Salman, A., 2005. Differential evolution methods for unsupervised image classification. Proceedings of IEEE Congress on Evolutionary Computation 2, 966–973.

Pai, P., Chen, S., Huang, C., Chang, Y., 2010. Analysing foreign exchange rates by rough set theory and directed acyclic graph support vector machines. Expert systems with applications 37, 5993–5998.

Pappas, S.S.P., Ekonomou, L., Karamousantas, D.Ch., Chatzarakis, G.E., Katsikas, S.K., Liatsis, P., 2008. Electricity demand loads modeling using autoregressive moving average (ARMA) models. Energy 33, 1353–1360.

Passino, K.M., 2002. Biomimicry of bacterial foraging for distributed optimization and control. IEEE control system magazine 22 (3), 52–67.

Paterlini, S., Krink, T., 2005. Differential evolution and particle swarm optimization in partitional clustering. Computational Statistics & Data Analysis 50 (5), 1220–1247.

Pham, Hong Thom, Yang, Bo-Suk, 2010. Estimation and forecasting of machine health condition using ARMA/GARCH model. Mechanical Systems and Signal Processing 24, 546–558.

Pham, Hong Thom, Tran, Van Tung, Yang, Bo-Suk, 2010. A hybrid of nonlinear autoregressive model with exogenous input and autoregressive moving average model for long-term machine state forecasting. Expert Systems with Applications 37, 3310–3317.

Popescu, T.H.D., Demetriu, S., 1990. Analysis and simulation of strong earthquake ground motions using ARMA models. Automatica 26 (4), 721–737.

Poskitt, D.S., 2003. On the specification of co-integrated autoregressive moving-average forecasting systems. Journal of Forecasting 19, 503–519.

Ray, Bonnie K., 1993. Long range forecasting of IBM product revenues using a seasonal fractionally differenced ARMA model. International Journal of Forecasting 9, 255–269.

Rojasa, I., Valenzuelab, O., Rojasa, F., Guillena, A., Herreraa, L.J., Pomaresa, H., Marquezb, L., Pasadas, M., 2008. Soft-computing techniques and ARMA model for time series prediction. Neurocomputing 71, 519–537.

Rout, M., Majhi, B., Majhi, R., Panda, G. 2011. Novel stock market prediction using a hybrid model of adaptive linear combiner and differential evolution. In: Proceeding of 2nd International Conference on Recent Trends in Information, Telecommunication and Computing, pp. 187–191 (partial autocorrelation).

Stoica, P., 1984. Uniqueness of estimated K-step prediction models of ARMA processes. Systems and Control Letters 4, 325–331.

Storn, R., Price, K., 1995. Differential Evolution – A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. International Computer Science Institute, Berkeley, TR-95–012.

Taylor, James W., 2006. Density forecasting for the efficient balancing of the generation and consumption of electricity. International Journal of Forecasting 22, 707–724.

Ursem, K., Vadstrup, P., 2003. Parameter identification of induction motors using differential evolution. Proceedings of IEEE Congress on Evolutionary Computation 2, 790–796.

Voyant, Cyril, Muselli, Marc, Paoli, Christophe, Nivet, Marie-Laure., 2012. Numerical weather prediction (NWP) and hybrid ARMA/ANN model to predict global radiation. Energy 39, 341–355.

Widrow, B., Strearns, S.D., 1985. Adaptive Signal Processing. Englewood Cliffs, NJ, NJ, Prentice-Hall.

Wu, Ji, Chan, Chee Keong, 2011. Prediction of hourly solar radiation using a novel hybrid model of ARMA and TDNN. Solar Energy 85, 808–817.

Xiong, Yimin, Yeung, Dit-Yan, 2004. Time series clustering with ARMA mixtures. Pattern Recognition 37, 1675–1689.

Yu, L., Wang, S., Lai, K.K., 2005. A novel nonlinear ensemble forecasting model incorporating GLAR and ANN for foreign exchange rates. Computers & Operations Research 32, 2523–2541.

Zhang, Y., Wan, X., 2007. Statistical fuzzy interval neural networks for currency exchange rate time series prediction. Applied Soft Computing 7, 1149–1156.

Zhou, Zhi-Jie, Hu, Chang-Hua, 2008. An effective hybrid approach based on grey and ARMA for forecasting gyro drift. Chaos, Solitons and Fractals 35, 525–529.