A New Hybrid Method for Time Series Forecasting

Tiago A. E. Ferreira, Germano C. Vasconcelos and Paulo J. L. Adeodato

***Abstract*—This paper presents a new method — the Time-delay Added Evolutionary Forecasting (TAEF) method — for time series prediction which performs an evolutionary search of the minimum necessary number of dimensions embedded in the problem for determining the characteristic phase space of the phenomenon generating the time series. The method proposed is inspired in F. Takens theorem and consists of an intelligent hybrid model composed of an artificial neural network (ANN) combined with a modified genetic algorithm (GA). Initially, the TAEF method finds the most fitted predictor model for representing the series and then performs a behavioral statistical test in order to adjust time phase distortions that may appear in the representation of some series. It is shown how this model proposed can boost the performance of time series prediction of both artificially generated time series and real world time series from the financial market. An experimental investigation is conducted with the TAEF method with five different relevant time series and the results achieved are discussed and compared with previous results found in the literature, according to several performance measures, showing the robustness of the proposed approach.**

***Index Terms*—Genetic Algorithms, Neural Network, Time Series, Forecasting.**

I. Introduction

New promising approaches based on artificial neural networks (ANNs) have been proposed for the non-linear modeling of time series [1]. However, in order to define a solution to a given problem, ANNs require the setting up of a series of system parameters, some of them not always easy to determine. The network topology, the number of processings units, the algorithm for network training (and its corresponding variables) are just some of the parameters that require definition. In addition to those, in the particular case of time series prediction, another crucial element that demands definition is the relevant time lags necessary to adequately represent the series.

In this work, a systematic procedure based on an hybrid intelligent system approach is proposed for the automatic search of the important system parameters that solve time series prediction problems. The adopted method consists of a combination of a standard neural network architecture with a modified genetic algorithm (GA) [2] which efficiently searches and defines 1. the minimum number of (and the specific) temporal lags necessary to solve the problem, based on F. Takens theorem [3], 2. the best neural network structure in terms of the number of processing units to be employed, 3. the most fitted training algorithm [4], [5], [6], [7] that boosts the prediction performance, and 4. a behavioral statistical test carried out at the prediction model output to fix relative phase distortions in the series representation. The proposed procedure is described in Section II. It is shown how this procedure can enhance prediction performance making use of a test bed composed of seven relevant time series: Hénon, Sunspot, Dow Jones Industrial Average, S&P500, Nasdaq, Petrobrás Stock Values (Brazilian petroleum company) and a Artificial Random Walk series.

Center for Informatics – Federal University of Pernambuco, Av. Prof. Luiz Freire, s/n, Cidade Universitária, 50732-970, Recife - PE - Brazil — e-mail: {taef, gcv, pjla}@cin.ufpe.br

II. The TAEF Method

The method proposed in this work — Time-delay Added Evolutionary Forecasting (TAEF) method — tries to reconstruct the phase space of a given time series by carrying out a search for the minimum dimensionality necessary to reproduce, to a certain accuracy, the phenomenon generator of the times series and its subsequent values. The proposed procedure is a intelligent hybrid system based on an artificial neural network (ANN) architecture (multilayer perceptron network - MLP) trained with a modified genetic algorithm (GA) [2] which not only searches for a number of the ANN parameters but also for the adequate embedded dimension represented in the lags. The accuracy of the prediction generated is initially set by the user but is automatically changed by the training algorithm if it finds a model with better accuracy.

The scheme describing the proposed algorithm is based on the iterative definition of the three main elements necessary for building an accurate forecasting system: 1. the underlying information necessary to predict the series (the minimum number of time lags adequate for representing the series); 2. the structure of the model capable of representing such underlying information for the purpose of prediction (the number of units in the ANN structure); and 3. the appropriate algorithm for training the model (the most appropriate algorithm among several candidates).

It is important to consider the minimum possible number of time lags in the representation of the series because the larger the number of lags the larger the cost associated with the model training.

Following this principle, the important parameters defined by the algorithm are: 1. **The number of time lags to represent the series:**initially, a maximum number of lags (*MaxLags*) is defined by the user and a GA can choose any number of lags in the interval [1, *MaxLags*] for each individual of the population; 2. **The number of units in the ANN hidden layer:** the maximum number of hidden layer units (*NHiddenmax*) is determined by the user and the GA chooses, for each candidate individual, the number of units in the hidden layer (in the interval [1,*NHiddenmax*]); 3. **The training algorithm for the ANN:** RPROP [4], Levenberg-Marquardt [5], Scaled Conjugate Gradient [6], One Step Secant Conjugate Gradient [7] are candidates for the best algorithm for training the ANN and the GA defines these algorithms as individuals in the population.

The algorithm starts with the user defining a minimum initial fitness value (*MinFit*) which should be reached by at least one individual of the population in a given GA round. The fitness function is defined as,

$$Fitness = \frac{1}{1 + MSE} \quad (1)$$

where *MSE* is the Mean Squared Error of the ANN and will be formally defined in the next section.

In each GA round, a population of M individuals is generated, each of them being represented by a chromosome (in the experiments carried out here $M = 10$). Each individual is in fact a three-layer ANN where the first layer is defined by the number of time lags, the second layer is composed of a number of hidden processing units (sigmoidal units) and the third layer is composed by one processing unit (prediction horizon of one step ahead).

Each individual has distinct network initialization and cross validation. The stopping criteria for each one of the individual are the number of epochs ($NEpochs$), the increase in the validation error (Gl) and the decrease in the training error (Pt).

The best repetition with the smallest validation error is chosen to represent the best individual. Following this procedure, the GA evolves towards a good fitness solution (which may not be the best solution yet), according to the stopping criteria: number of generations created ($NGen$) and fitness evolutions of the best individual ($BestFit$).

After this point, when the GA reaches a solution, the algorithm checks if the fitness of the best individual paired or overcame the initial value specified for the variable *MinFit* (minimum fitness). If this is not the case, the value of *MaxLags* (maximum number of lags) is increased by the unit and the GA procedure is repeated to search for a better solution. The objective here is to increase the possible number of lags in the lag set until a solution of minimum fitness is reached.

However, if the fitness reached was satisfactory, then the algorithm checks the number of lags chosen for the best individual, places this value as *MaxLags*, sets *MinFit* with the fitness value reached by this individual, and repeats the whole GA procedure. In this case, the fitness achieved by the best individual was better than the fitness previously set and, therefore, the model can possibly generate a solution of higher accuracy with the lags of the best individual (and with the *MinFit* reached by the best individual as the new target). If, however, the new value of *MinFit* is not reached in the next round, *MaxLags* gets again the same value defined for it just before the round that found the best individual, increased by the unit (the maximum number of lags is increased by one). The idea here is that if the time lags found in the best individual were not capable of producing a higher fitness than the one previously found this may be because some important lag (or lags) was discarded. The state space for the lag search is then increased by one to allow a wider search for the definition of the lag set. This procedure goes on until the stop condition is reached. After that, the TAEF method chooses the best model found among all the candidates.

In order to conclude the definition of the method a last aspect had to be considered. During the development and test of the method, a peculiar prediction behavior was observed in the prediction model. While the representations of some series were developed by the model with a very close approximation between the actual series and the predicted series ("in-phase" matching), the predictions of some other series were always presented with a one step shift (delayed) with respect to the original data ("out-of-phase" matching). This out-of-phase behavior was always found in the prediction of the financial series, whereas the in-phase matching was observed in all the other types of series (natural phenomena series). An interesting point to observe is that this one step delay behavior with respect to the actual series is similar to a random walk like model. Since it is a common sense in finance and economics that financial times series behave like random walks [8], as a first approximation, it is not strange that predictor models generated for them show this one step time delay distortion.

This observation is also in accordance with some other results reported in the literature. Sitte and Sitte [9] showed that predictions of financial time series represented by an ANN exhibit a characteristic one step shift with respect to the original data (out-of-phase matching). They argued that the financial series is represented by the ANN as it were a random walk.

In any circumstance, in order to make the TAEF method robust for representation of any time series, another element was introduced in the method operation. After the best model is chosen when training is finished, an statistical test is employed to check if the network representation has reached an in-phase or out-of-phase matching. This is conducted by comparing the outputs of the prediction model with the actual series making use of the validation data. If this test (for example the t-test) accepts the in-phase matching hypothesis, the elected model is ready for practical use. Otherwise, the method carries out a new procedure to adjust the relative phase between the prediction and the actual time series. The validation patterns are presented to the ANN and the output of these patterns are re-arranged to create new inputs that are both presented to the ANN and set as the output (prediction) target. The approximation results for both the in-phase and out-of-phase models are measured and the best model (smaller MSE error) is elected as the final model. Figure 1 depicts the complete algorithm for the TAEF model construction.

III. Performance Evaluation

Most of the works found in the literatire of time series prediction frequently employ only one performance criterion for model evaluation. Most of the times, the measure used is the MSE (mean squared error),

$$MSE = \frac{1}{N} \sum_{j=1}^{N} (target_j - output_j)^2 \quad (2)$$

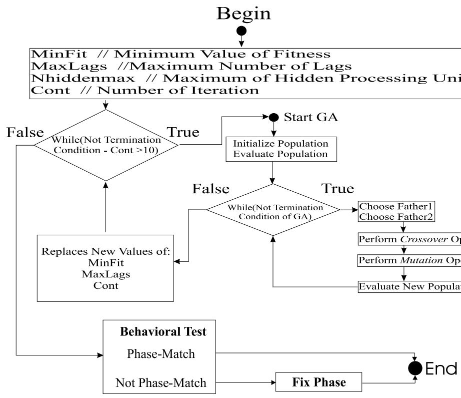

Fig. 1. Algorithm for the TAEF method

where N is the number of patterns, $target_j$ is the desired output for pattern j and $output_j$ is the predicted value for pattern j.

Although the MSE measure may be used to drive the prediction model in the training process, it cannot be considered alone as a conclusive measure for comparison of different prediction models [10]. For this reason, other performance criteria should be considered for allowing a more robust performance assertiveness.

A second relevant measure is the $MAPE$ (Mean Absolute Percentage Error), given by

$$MAPE = \frac{100}{N} \sum_{j=1}^{N} \left| \frac{target_j - output_j}{X_j} \right| \tag{3}$$

where N, $target_j$, and $ouytput_j$ are the same MSE parameters, and X_j is the time series at point j.

A third performance measure is the U of Theil Statistics, or NMSE (Normalized Mean Squared Error), which is given by

$$Theil = \frac{\sum_{j=1}^{N} (target_j - output_j)^2}{\sum_{j=1}^{N} (target_j - target_{j+1})^2} \tag{4}$$

which associates the model performance with a random walk model. If the U of Theil Statistics is equal to 1, the predictor has the same performance of a random walk model. If the U of Theil Statistics is greater than 1, then the predictor has a worse performance than a Random Walk model, and if the U of Theil Statistics is less than 1, the predictor is better than a random walk model.

Another relevant evaluation measure considers the calculation of the correctness of Prediction of Change in Direction, or $POCID$ for short,

$$POCID = 100 \frac{\sum_{j=1}^{N} D_j}{N} \tag{5}$$

where

$$D_j \begin{cases} 1 & if (target_j - target_{j-1})(output_j - output_{j-1}) > 0, \\ 0 & otherwise. \end{cases} \tag{6}$$

Since all measures above do not consider the freedom degrees of the model, two last performance measures can be taken into account. This is relevant because the larger the number of free parameters, the larger the probability of overfitting in the series estimation. The evaluation measures that include the freedom degrees, penalizing the models with additional parameters, are the Akaike (AIC) and Bayesian (BIC) information criteria. The AIC and BIC are approximated by

$$AIC = N \ln(MSE) + 2p \tag{7}$$

$$BIC = N \ln(MSE) + p + N \ln(p) \tag{8}$$

where N is the number of time series points, MSE is the Mean Squared Error and p is the number of freedom degrees.

IV. Experimental Results

A set of five times series was used as a test bed for evaluation of the method proposed. The first series is the known Hénon series (artificial series without noise), and the other series were drawn from real world situations: Sunspot, Dow Jones Industrial Average (DJIA), Nasdaq and Petrobrás Stock Options (Brazilian petroleum company).

All series investigated were normalized to lie within the interval [0,1] and divided in three sets, training set (50% of the points), validation set (25% of the points) and test set (25% of the points). The AG parameters are the same for all the series, with a mutation probability of 10%, and crossover and mutation operations, as those reported in Leung et al [2], composed of father genes and the maximal and minimal parameters allowed for the chromosomes that speeds up the search. For all the experiments carried out, the following system parameters were employed: initialization parameters — $MinFit = 0.99$ ($\sim$ 1% of error), $MaxLags = 4$ and $NHiddenmax = 20$; stopping conditions for the GA — $NGen = 1000$ and $BestFit \leq 0.0001$; Stopping conditions for each individual — $NEpochs = 1000$, $Gl \leq 5\%$ and $Pt \leq 10^{-6}$.

A. Hénon Series

The Hénon series is a very popular example of time series investigated by several researchers due to its complex nature and chaotic dynamics. An interesting work that employed this series was conducted by D.B.Murray [11]. Such as in the present work, Murray was interested in proposing a model to represent the phase space of the temporal lags. He developed his approach based on the idea of building this phase space of embedded dimensions from a metric tensor whose components are adjusted in order to the minimize the prediction error (root mean square error, RMSE). The best prediction results obtained by Murray corresponded to 3.7e-3 (RMSE), or 1.4e-5 (1.4e-3 %), if considered the mean square error (MSE) .

The Hénon series considered in this work was the same as that used by Murray, being composed of 10.000 points generated from Equation (9) with parameters $a = 1.4$ and

$b = 0.3$. This series is generated without the inclusion of any noise (the r terms are null) .

$$X_t = 1 - a(X_{t-2} - r_{t-2})^2 + b(X_{t-4} - r_{t-4}) + r_t \qquad (9)$$

For the prediction of Henon series (with 1 step ahead of prediction horizon), the TAEF method identified the lags 2, 3, 5 and 7 as the relevant to the problem, defined 14 processing units in the hidden layer of the network, elected the Levenberg-Marquardt algorithm as the most fitted for the ANN training and classified the prediction model as "in-phase" matching. Table I shows the results with all the performance measures presented in Section III for both cases: "in-phase" matching and if the prediction model had been chosen as "out-of-phase" matching. The prediction results obtained by the proposed method were significantly better than those reported in the work of D.B. Murray, with an MSE prediction of $3.1678 \cdot 10^{-11}$ for the test set.

Figure 2 shows a comparative graph of the actual Henon series (solid lines) and the prediction generated by the TAEF method (dash lines) for the last 100 points of the test set, for both cases of prediction hypotheses (in-phase matching and out-of-phase matching).

TABLE I

Experimental Results for the Hénon Series

	In-phase Matching	Out-of-phase Matching
MSE	$3.1678 \cdot 10^{-11}$	1.0445
MAPE	0.0061 %	305.09%
U of Theil	$1.9836 \cdot 10^{-10}$	0.9996
POCID	100.00%	41.51%
AIC	-55251.0	556.7
BIC	-53722.6	2085.0

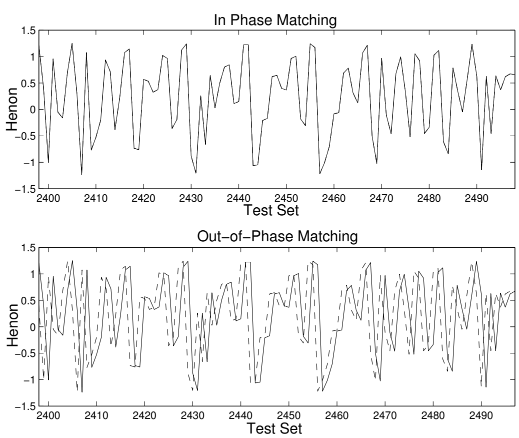

Fig. 2. Prediction results for the Hénon series (test set): actual values (solid lines) and predicted values (dashed lines).

B. Sunspot Series

The sunspot series used consisted of the total annual measures of the sun spots from the years of 1700 to 1988, generating a database of 289 examples.

N.Terui and H.K.Van Dijk [12] developed a work where a combination of some linear and non-linear models were employed for times series prediction. Among the series investigated, Terui and Van Dijk employed the sunspot series from the years 1720 to 1989 to test their method based on the combination of the AR, TAR and ExpAR models The best experimental results reported with their proposed method (best model combination) corresponded to an MSE error of 0.0390.

TABLE II

Experimental Results for the Sunspot Series

	In-phase Matching	Out-of-phase Matching
MSE	$8.600 \cdot 10^{-3}$	0.3070
MAPE	34.03 %	82.55 %
U of Theil	0.3218	1.2225
POCID	84.29%	65.22%
AIC	-289.6	-195.9
BIC	-211.3	-117.9

For the prediction of the Sunspot series (with 1 step ahead of prediction horizon), the TAEF method identified the lags 1 to 4 as the relevant to the problem, defined 4 processing units in the hidden layer of the network, elected the Levenberg-Marquardt algorithm as the most fitted for the ANN training and classified the model as "in-phase" matching. Table II shows the results with all the performance measures for both cases: "in-phase" matching and if the prediction model had been chosen as "out-of-phase" matching.

It can be seen that the proposed method produced a prediction error of $8.600 \cdot 10^{-3}$, consistently better than that observed in the work of Terui and Van Dijk (0.0390).

Figure 3 shows a comparative graph of the actual Sunspot series (solid lines) and the prediction generated by the TAEF method (dash lines) for the test set, for both cases of prediction hypotheses (in-phase matching and out-of-phase matching). It is seen that the "in-phase" matching model chosen by TAEF method was the correct choice.

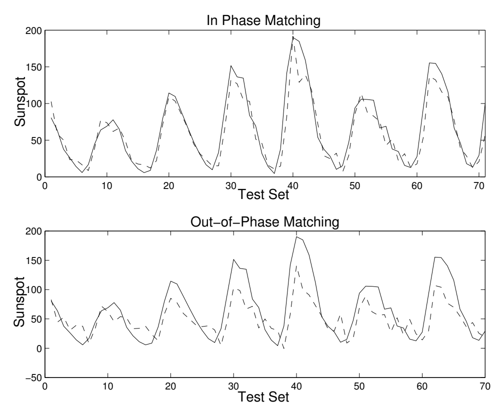

Fig. 3. Prediction results for the Sunspot series (test set): actual values (solid lines) and predicted values (dashed lines).

C. Down Jones Series

The Dow Jones Industrial Average Index (DJIA) series corresponds to daily observations from 1st January 1998 to 26th of August 2003 of the DJIA index, constituting a database of 1420 points.

The hybrid model proposed automatically chose the lags 2, 4, 8, 6, 9 and 10 as the relevant lags for the series representation, defined 10 processing units for the hidden layer of the ANN, once again selected the algorithm Levenberg-Marquardt as the most fitted for the ANN training and classified the model as "out-of-phase" matching. Table III shows the results with all the performance measures for both cases: out-of-phase matching and if the prediction model had been chosen as "in-phase" matching. Of particular interest to this financial series are the measures shown by the statistics U of Theil (0.03), denoting a far better result than a random walk (see Section III), and by the POCID which presents a 97% of series approximation.

TABLE III

Experimental Results for the DJIA Series

	In-phase Matching	Out-of-phase Matching
MSE	$8.4183 \cdot 10^{-4}$	$2.6841 \cdot 10^{-5}$
MAPE	1.15 %	0.20%
U of Theil	1.0006	0.0318
POCID	47.58%	97.14%
AIC	-2206.1	-3408.5
BIC	-1510.6	-2713.4

Figure 4 shows a comparative graph of the actual DJIA (solid lines) and the prediction generated by the TAEF method (dash lines) for the last 100 points of the test set, for both cases of prediction model classification (in-phase matching and out-of-phase matching). It is seen that the "out-of-phase" matching model chosen by TAEF method was the correct choice.

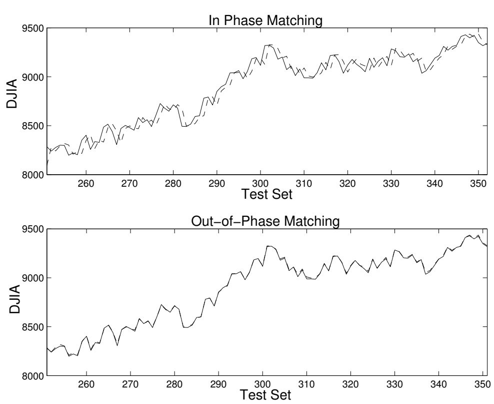

Fig. 4. Prediction results for the DJIA series (test set): actual values (solid lines) and predicted values (dashed lines).

D. Nasdaq Series

The Nasdaq series (National Association of Securities Dealers Automated Quotation) corresponds to daily observations from 2nd February 1971 to 18th of June 2004 of the Nasdaq index, constituting a database of 8428 points.

For the prediction of the Nasdaq series (with 1 step ahead of prediction horizon), the TAEF method identified the lags 3, 4 6 and 8 as the relevant to the problem, defined 11 processing units in the hidden layer of the network, elected the Levenberg-Marquardt algorithm as the most fitted for the ANN training and classified the model as "out-of-phase" matching. Table IV shows the results with all the performance measures for both cases: "out-of-phase" matching and if the prediction model had been chosen as "in-phase" matching. Of particular interest to this financial series are the measures shown by the statistics U of Theil (0.17), denoting a far better result than a random walk (see Section III), and by the POCID which presents a 89.6% of series approximation.

TABLE IV

Experimental Results for the Nasdaq Series

	In-phase Matching	Out-of-phase Matching
MSE	$2.1449 \cdot 10^{-5}$	$3.2374 \cdot 10^{-6}$
MAPE	0.20 %	0.08%
U of Theil	1.1441	0.1720
POCID	52.71%	89.63%
AIC	-22342.4	-26310.1
BIC	-21391.2	-25358.9

Figure 5 shows a comparative graph of the actual Nasdaq series (solid lines) and the prediction generated by the TAEF method (dash lines) for the last 100 points of the test set, for both cases of prediction hypotheses (in-phase matching and out-of-phase matching).It is seen that the "out-of-phase" matching model chosen by TAEF method was the correct choice.

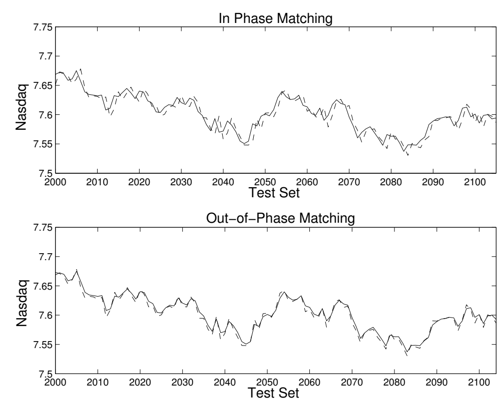

Fig. 5. Prediction results for the Nasdaq series (test set): actual values (solid lines) and predicted values (dashed lines).

E. Petrobrás Stock Values Series

The Petrobrás series corresponds to the daily records of the Brazilian Petroleum Company stock values from the 1st of January 1995 to the 3rd of July 2003, totalizing 2060 points.

For the prediction of the Petrobrás series (with 1 step ahead of prediction horizon), the TAEF method identified the lags 3, 4 and 7 as the relevant to the problem, defined 17 processing units in the hidden layer of the network, elected the Levenberg-Marquardt algorithm as the most fitted for the ANN training and classified the model as "out-of-phase" matching. Table V shows the results with all the performance measures for both cases: "out-of-phase" matching and if the prediction model had been chosen as "in-phase" matching. Of particular interest to this financial series are the measures shown by the statistics U of Theil (0.30), denoting a far better result than a random walk (see Section III), and by the POCID which presents a 97.7% of series approximation.

TABLE V
EXPERIMENTAL RESULTS FOR THE PETROBRÁS SERIES

	In-phase Matching	Out-of-phase Matching
MSE	$7.5951 \cdot 10^{-5}$	$1.9049 \cdot 10^{-5}$
MAPE	0.55 %	0.29 %
U of Theil	1.2077	0.3023
POCID	52.79%	97.68%
AIC	-4286.4	-4997.7
BIC	-2589.4	-3298.4

Figure 6 shows a comparative graph of the actual Petrobrás series (solid lines) and the prediction generated by the TAEF method (dash lines) for the last 100 points in the test set, for both cases of prediction hypotheses (in-phase matching and out-of-phase matching).It is seen that the "out-of-phase" matching model chosen by TAEF method was the correct choice.

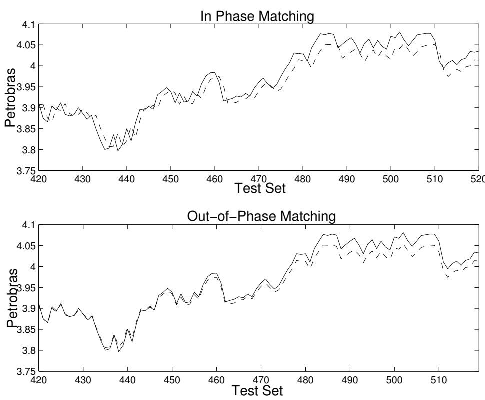

Fig. 6. Prediction results for the Petrobrás series (test set): actual values (solid lines) and predicted values (dashed lines).

V. Conclusions

This paper has presented an intelligent hybrid system approach, the Time-delay Added Evolutionary Forecasting (TAEF) method, which consists of an artificial neural network combined with a modified genetic algorithm and a behavior test of phase match hypotheses carried out at the model's output for the solution of time series forecasting problems.

The experimental results using a set of consistent performance measures composed with six different metrics (MSE, MAPE, U of Theil Statistics, POCID, AIC and BIC) showed that this system can boost the performance of time series prediction on both artificially generated time series and real world (financial market and natural phenomena) time series. The experimental validation of the method was carried out on five complex and relevant time series: four real world time series (with all their dependence on exogenous and uncontrollable variables) and the artificially generated Hénon series (with its non-linear relations and chaotic characteristics).

With the introduction of the behavior test for identifying whether the prediction model is "in-phase" or "out-of-phase" with the series to be forecasted, the TAEF method was able to classify if a given time series tends or not to a Random Walk like model, thus adjusting the model if necessary. Such adjustment is conducted on the model constructed without the use of any additional training phase nor the use of any additional training data (the same original validation data is employed). Only one additional epoch is used for presenting the original validation data and deciding which of the models generated (in-phase or out-of-phase) produces the best series approximation.

When compared to the best (most recent) results found in the literature, the TAEF Method presented a superior performance in all the comparisons made. However, a systematic study is yet necessary to determine any possible limitations of the method when dealing with other types of components found in other different real world time series such as trends, seasonality, impulses, steps, and other non-linearities. Taking that into account, other time series with those components are being collected to carry out a broader investigation.

References

[1] G. Zhang, B. Eddy Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks: The state of the art," *International Journal of Forecasting*, vol. 14, pp. 35–62, 1998.
[2] F. H. F. Leung, H. K. Lam, S. H. Ling, and P. K. S. Tam, "Tuning of the structure and parametrs of the neural network using an improved genetic algorithm," *IEEE Transaction on Neural Networks*, vol. 14, no. 1, pp. 79–88, January 2003.
[3] Floris Takens, "Detecting strange attractor in turbulence," in *Dynamical Systems and Turbulence*, A. Dold and B. Eckmann, Eds., New York, 1980, vol. 898 of *Lect. Notes in Maths.*, pp. 366–381, Springer-Verlag.
[4] M. Reidmiller and H. Braun, "A direct adaptive method for faster backpropagation learng: The rprop algorithm," in *Proceedings of the IEEE Int. Conf. on Neural Networks (ICNN)*, San Francisco, 1993, pp. 586–591.
[5] J. J. Mor, "The levenberg-marquardt algorithm: Implementation and theory," in *Proceedings of Springer-Verlag in Numerical Analysis (Lecture Notes in Mathematics)*, 1977, pp. 105–116.
[6] M. F. Moller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Networks*, vol. 6, pp. 525–533, 1993.
[7] R. Battiti, "One step secant conjugate gradient," *Neural Computation*, vol. 4, pp. 141–166, 1992.
[8] Terence C. Mills, *The Econometric Modelling of Financial Time Series*, Cambridge University Press, Cambridge, 2003.
[9] Renate Sitte and Joaquin Sitte, "Neural networks approach to the random walk dilemma of financial time series," *Applied Intelligence*, vol. 16, no. 3, pp. 163–171, May 2002.
[10] Michael P. Clementes and David F. Hendry, "On the limitations of comparing mean squared forecast erros," *Journal of Forecasting*, vol. 12, no. 8, pp. 617–637, 1993.
[11] D. B. Murray, "Forecasting a chaotic time series using an improved metric for embedding space," *Physica D*, vol. 68, pp. 318–325, 1993.
[12] N. Teru and H. K. Van Dijk, "Combined forecasts form linear and nonlinear time series models," *Intenational Journal of Forecasting*, vol. 18, pp. 421–438, 2002.
[13] C. A. Casas, "Tatical asset allocation: an artificial neural network based model," in *IJCNN'2001*. IEEE, 2001, pp. 1844–1816.