Electrical Power and Energy Systems 32 (2010) 108–116

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

Electricity demand and spot price forecasting using evolutionary computation combined with chaotic nonlinear dynamic model

C. Unsihuay-Vila *, A.C. Zambroni de Souza, J.W. Marangon-Lima, P.P. Balestrassi

Power and Energy Systems Group-GESis, Federal University of Itajubá, MG, Brazil

ARTICLE INFO

Article history:
Received 12 May 2008
Received in revised form 7 April 2009
Accepted 7 June 2009

Keywords:
Electricity markets
Short-term load forecasting
Price forecasting
Evolutionary strategy optimization
Nonlinear chaotic dynamics
Predict (Matlab code)

ABSTRACT

This paper proposes a new hybrid approach based on nonlinear chaotic dynamics and evolutionary strategy to forecast electricity loads and prices. The main idea is to develop a new training or identification stage in a nonlinear chaotic dynamic based predictor. In the training stage five optimal parameters for a chaotic based predictor are searched through an optimization model based on evolutionary strategy. The objective function of the optimization model is the mismatch minimization between the multi-step-ahead forecasting of predictor and observed data such as it is done in identification problems. The first contribution of this paper is that the proposed approach is capable of capturing the complex dynamic of demand and price time series considered resulting in a more accuracy forecasting. The second contribution is that the proposed approach run on-line manner, i.e. the optimal set of parameters and prediction is executed automatically which can be used to prediction in real-time, it is an advantage in comparison with other models, where the choice of their input parameters are carried out off-line, following qualitative/experience-based recipes. A case study of load and price forecasting is presented using data from New England, Alberta, and Spain. A comparison with other methods such as autoregressive integrated moving average (ARIMA) and artificial neural network (ANN) is shown. The results show that the proposed approach provides a more accurate and effective forecasting than ARIMA and ANN methods.

1. Introduction

Accurate forecasting of the electricity demand and spot price is essential in the operation of electric power system, especially in deregulated electricity markets. Along with the forecasted electricity prices, producers can develop bidding strategies to maximize profits and minimize risks, while consumers can allocate purchases between long-term bilateral contracts and spot markets. Accurately forecasting electricity price and load demand are necessary for investors to optimize portfolios [1]. Transmission congestion, maintenance schedule of generation units, fuel or water supply, etc., might affect the electricity price dramatically, complicating the forecasting problem [1–3].

A number of papers dealing with short-term load forecasting have been reported in [1,4–15]. Ref. [4] provides a comprehensive review of many methodological issues and techniques which have become innovative in addressing the problem of forecasting daily loads. The range of approaches for generating forecasts includes exponential smoothing [5] and neural networks [1,6]. Interesting approaches based on chaos theory were proposed in [7–16]. More recently hybrid approaches have been proposed in [17,18].

A considerable number of techniques of forecasting day-ahead prices are described in the literature. Techniques based on ARIMA models were presented in [2,19,20]. Ref. [21] provides a comprehensive review of some main methodological issues and techniques which have become innovative in addressing the problem of forecasting daily loads and prices in the new competitive power markets. In [1,22–24] neural network approaches were proposed to forecast short-term electricity price. A GARCH forecasting model to predict day-ahead electricity prices was proposed in [25]. An approach based on fuzzy classification was shown in [26], whereas wavelet transform models were discussed in [27] and an interesting hybrid approach is presented in [28]. Finally, models based on chaos theory were presented in [29–33].

Some believe that electricity demand seems random, but some believe that it seems chaotic, due to the influence of many complicated facts such as temperature, price of electricity and many other factors [7–16]. Similarly the price of electricity depends on the supply and demand of the market and the operating conditions of the transmission network, which are influenced by many factors, such as the climate, the economic situation, the planning for development, accidents and failure [29–33]. The joint effect of these factors results in complicated dynamics of electricity demand and

* Corresponding author.
E-mail addresses: clodomiro@unifei.edu.br (C. Unsihuay-Vila), zambroni@unifei.edu.br (A.C. Zambroni de Souza), marangon@unifei.edu.br (J.W. Marangon-Lima), pedro@unifei.edu.br (P.P. Balestrassi).

0142-0615/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijepes.2009.06.018

Nomenclature

Indexes

N	length of historical time series
M	number of desired steps-ahead prediction
nTr	dimension of the training set vector

Parameters

μ	number of parents used in evolutionary strategy (ES)
γ	number of offspring used in ES
S	vector of the time series data values
$\hat{y}_{z,i}$	real value of time series
σ	standard deviation vector
TSD	historical time series data (an $N \times 1$ vector dimension)
SSR	the state space reconstruction set
TRN	the training set

Variables

$\mathbf{x}$	vector of variables correspondent to input parameters of PREDICT2
x_j	input parameter j of PREDICT2
$\mathbf{x}_{\max}$, $\mathbf{x}_{\min}$	maximum and minimum limits of vector of variables correspondent to input parameters of PREDICT2
m	dimensionality of embedding space
τ	time delay
k	size of the local neighborhood
λ	the Euclidian distance metric
RF	type of regression functions of local constant models or method of computing the prediction output
$y_{z,i}$	prediction value of time series
Output^i	the $(nTr \times 1)$ column vector whose elements $\hat{y}_{z,i}$ are the nTr-step-ahead prediction results, using PREDICT2 associated to ith individual candidate

price. Usually statistics methods are employed to develop forecasting models for electricity price and load.

Recent developments in nonlinear dynamics have demonstrated that irregular or random behavior in natural systems may arise from purely deterministic dynamics with unstable trajectories. Even though some observations might appear random, there may exist an order or pattern beneath such an appearance. Such types of nonlinear dynamical systems, which are also highly sensitive to initial conditions, are known as chaotic systems [34,35]. Furthermore, if the seemingly random evolution of the electricity demand or price also possesses a chaotic trait, the theory of probability and statistics is not accurate enough to study both as a stochastic variable [7–16,29–35]. Instead, the chaos theory can be used to reveal its intrinsic regularity until more accurate and rational analysis results and prediction models are obtained [7–16,29–35].

The quality of a forecaster based on chaotic dynamics is highly dependent on parameters of the time series dynamics. Generally the parameters are assessed at the system characterization stage. Regarding forecasting precision, most of the chaos based methods have good performance for short-term chaotic time series forecasting. On the other hand, for stochastic time series and time chaotic series under high embedding dimension, the chaos based methods do not present a good behavior. Thus, a judicious time series analysis and/or hybridization with other methods are necessary to improve the time series forecasting to any kind of time series.

This paper presents a novel hybrid nonlinear chaotic dynamic and evolutionary strategy-based approach for multi-step-ahead time series forecasting. The fundamental and novel contribution of the paper is the insertion of a new training or identification stage into a nonlinear chaotic dynamic based predictor. In this training stage, five optimal parameters for a nonlinear chaotic dynamic based predictor are searched through an optimization model based on evolutionary strategy. Hence a common identification objective is used to minimize the mismatch between model prediction and observed data. For this purpose, an optimal time series identification stage is developed to find a model with good prediction capabilities. Thus, this paper proposes a new robust hybrid model for multi-step ahead prediction of time series regardless if the time series follows a chaotic, stochastic, and/or any other type of dynamic behavior.

The proposed approach is applied for multi-step ahead prediction of load and price and it is also compared with traditional techniques like ARIMA and ANN models. The load and electricity price data of New England, Alberta, and mainland Spain are used to corroborate the ideas and to obtain the results.

The paper is organized as follows: Section 2 provide the details of the proposed hybrid nonlinear chaotic dynamics and evolutionary strategy based approach, Section 3 presents numerical results of the simulations and Section 4 discusses the conclusion.

2. An hybrid approach based on nonlinear chaotic dynamic and evolutionary strategy

2.1. Times series prediction by chaotic nonlinear dynamics

The theoretical fundaments of times series forecasting using chaotic nonlinear dynamic methodologies is out of scope of this paper. The interested reader is referred to [34,35] for a better discussion.

A landmark in the chaos signal processing was made with the origin of embedding theorem of Takens [34,35]. This theorem explored the time-lagged vectors to realize the underlying dynamics, whereby, a dynamic of a real process result in a time series $\eta(t) = \{\eta(t_0 + n\tau)\}$ is sampled at intervals τ and initiated at t_0. Consider a dynamical system with a m-dimensional space and an evolving solution $g(t)$. For some observation, the lag vector can be defined as:

$$\eta(t) \equiv \left\{\eta_t, \eta_{t-\tau_1}, \eta_{t-\tau_2}, \eta_{t-\tau_3}, \ldots, \eta_{t-\tau_{m-1}}\right\}. \quad (1)$$

Then, under general conditions, the space of vectors $\eta(t)$ generated by the dynamics contains all of the information of the space of solution vectors $g(t)$. The mapping between them is smooth and invertible. This property is referred to as the embedding theorem. Thus, the study of the time series $\eta(t)$ is also the study of the solutions of the underlying dynamical system $g(t)$ through a particular coordinate system given by the observable η.

The embedding theorem establishes that, given a scalar time series from a dynamical system, it is possible to reconstruct a phase space from this single variable, that is, in theory, an embedded space with dimensions consisting of various time lags of the variable itself. The embedded space can also be created from many dynamic variables. According to the embedding theorem, the underlying structure cannot be seen in the space of the original scalar time series, rather only when unfolded into an embedded (or phase) space. Time series can correspondingly be forecasted based on this structure in the phase space. The purpose of the forecasting is to predict the state of the system $\eta(t)$ at a time horizon T in the future $\eta(t+T)$.

The vector on Eq. (1) represents the nonlinear dynamics in its entity when the embedding dimension m is large enough and the selection of the time delay τ is appropriate. For a good approximate first guess, the values would give a feeling on the efficiency of the embedding dimension. There are many methods available to estimate m and τ. Typically, the selection of m is based on the concept of average mutual information (AMI) and m using false nearest neighbor (FNN) analyses [34,35].

When the "optimal" estimate of m and τ is identified, the phase space H is reconstructed with the elements of $\eta(t)$. The prediction model can then be built in this m-dimensional space.

$$\eta(t+T) = g_T(\eta(t)), \tag{2}$$

where the lag vector $\eta(t)$ is the current state of the system. $\eta(t+T)$ is the system state in a forecast horizon interval of T and g_T is a mapping function. The problem now is limited to find a good expression for g_T. In the local model, $g(t)$ is evaluated among only the most similar points which are locally present near the forecast point. The sub-phase space is identified from the historical time series by choosing the k nearest neighboring points within the m-dimensional space, H. The neighbors can be chosen either within a circle of constant radius (distance) from the forecast point or by specifying the number of nearest neighbors k. There are many algorithms to find the k nearest neighbors in a m-dimensional space occurring in nonlinear time-series analysis, especially for modeling and prediction of time series via time-delay reconstruction. For this purpose, fast nearest neighbor (FNN) searching algorithms have been proposed in [36,37].

Having constructed the phase space and pooled the most similar events in the past corresponding to the present time horizon, T, the desired expected (forecast) value vector, $\Omega(t)$ is formed for each point in the neighborhood domain, say H' where $H' \in H$. The regression is performed using the neighborhood coordinates in the sub-domain $H'()$ as inputs, and their corresponding expected values $\Omega(t)$ as outputs.

2.2. Chaotic dynamics models in times series prediction

The accuracy of nonlinear local dynamics models approximation and resulting forecast depend on many factors, such as dimensionality of embedding space (m), size of the local neighborhood (k), time delay (τ), the Euclidian distance metric (λ), of the nearest trajectory algorithms, the type of regression functions (RF) of local constant models or method of computing the prediction output [34,35].

State space reconstruction using the time delay coordinate method is clearly common to both system characterization and prediction. Conventionally, it is assumed that the state space parameters m and τ for forecasting are the same as those estimated using AMI and FNN method, respectively, for system characterization purposes. Similarly, the choice to $k = m+1$, is often used [34,35]. The λ value is commonly fixed at 1 and the type of regression function (RF) is pre-established before the forecasting stage. Such local dynamics parameters provide robust, but in principle at least sub-optimal choices of embedding parameters, thus resulting in a sub-optimal embedding properties as well as a sub-optimal forecast skill [38,39].

The above consideration would be justified and motivated, since there are no rigorous or foolproof criteria to validate the presence of chaos in real time series. In addition, there are no criteria to evaluate state space parameters precisely and reliably. Many of these criteria are developed based on the assumptions that the time series is sampled at sufficient resolution, not corrupted by noise, and measured over a sufficiently long period of time, which may not be valid in practice. It suffices to say that the limitations of existing criteria for system characterization permit a fairly large degree of latitude in the selection of the appropriate state space parameters. Linking forecasting with system characterization will certainly propagate this uncertainty and affect prediction accuracy [38,39].

The above discussion suggests that it may be more practical to select all the parameters together during the forecasting stage. If the parameters are selected to produce optimum prediction accuracy, then such a direct approach will at least be as accurate as the standard approach as far as forecasting is concerned. From a pragmatic engineering point of view, such a selection procedure for state space parameters is also highly desirable, because accurate prediction is usually the primary motivation for developing engineering models in the first place. Hence an optimal time series identification stage is necessary to identify a model with good prediction capability [38,39].

Thus, in this paper the insertion of a new training stage in a nonlinear chaotic dynamic based predictor (PREDICT2) [37] is proposed to improve the time series modeling and forecasting. This training stage consists in finding five optimal parameters of PREDICT2 using evolutionary strategy.

PREDICT2 is a code of a free software package for signal processing with emphasis on nonlinear time-series analysis (TSTOOL) [40]. Such a function is a state space based prediction using nearest neighbors. The algorithm computes one or more nearest neighbors to an initial state vector. The images of the nearest neighbors are used to estimate to image of the initial state vector [40].

The syntax of PREDICT2 is as follows:

$$\text{Output} = \text{PREDICT2}(S, nTr, m, \tau, k, \lambda, RF) \tag{3}$$

RF variable represents the regression function type alternatives and its numerical equivalence (correspondence) would takes integer values from 0 to 3 [40]: $RF = 0$ (i.e. when the output vectors represent the mean of the images of the nearest neighbors), $RF = 1$ (i.e. when the output vectors represent the distance weighted mean of the images of the nearest neighbors), $RF = 2$ (i.e. when the output vectors based on the local flow uses the mean of the images of the neighbors), $RF = 3$ (i.e. when the output vectors based on the local flow uses the weighted mean of the images of the neighbors).

The *nTr* value is the multi-ahead-step prediction length (number of output values) and S is the time series data values. Finally, *Output* is the set of *nTr* forecast values which in the next section this variable is renamed as dimension of the training set vector.

2.3. The hybrid PREDICT2 and evolutionary strategy (ES) based approach to times series forecasting

In this paper, a hybrid approach, which combines the predictor PREDICT2 with Evolutionary Strategy (ES) to improve the quality of time series forecasting is proposed. A new training or identification stage, which selects all the five PREDICT2 prediction parameters m, τ, k, λ and RF is proposed to optimize the prediction accuracy. The training stage is an optimization process, where the constraints are the same as those associated with the PREDICT2 problem and the objective function is to minimize the mismatch between model prediction and the observed data.

Evolutionary strategies belong to the wide class of evolutionary computation algorithms. Briefly, they consist in selecting a set of μ candidates for the solution of the optimization problem and applying the rules of evolution until an optimal solution is obtained. A typical candidate or individual consists of a pair of vectors, one containing the parametric solution of the system $(\mathbf{x})$ and another containing a vector of standard deviations (σ) which controls the evolution of the individual in the subsequent steps. These initial candidates conforms the parent population (Υ). From these μ solutions a batch of γ offspring are generated according to the mechanisms of recombination and mutation, as described next. These solutions are then evaluated according to the fitness function or an optimization criterion and they are ranked from best to worst. The best ones are chosen following a selection method to form the next parent population, and the process is iterated. The main advantages of this method are the low sensitivity to initial estimates and the ability to escape from local minima. For further details on the workings of evolutionary strategies, the reader should refer to [41].

Before starting the evolutionary strategy algorithm, it is necessary to establish the search space. The search space Ψ is defined by a set of maximum and minimum values for each parameter. It conceives an n-dimensional domain delimited by vectors $\mathbf{x}_{\max}$ and $\mathbf{x}_{\min}$ containing the upper and lower bounds of the n parameters, respectively:

$$\Psi = \left\{\mathbf{x} \in \mathbb{R}^n | x_{\min,j} \leqslant x_j \leqslant x_{\max,j}; \quad j = 1,\ldots,n\right\}. \tag{4}$$

In this paper,

$$\mathbf{x} = [m \quad \tau \quad k \quad \lambda \quad RF]. \tag{5}$$

Because these parameters are the input to PREDICT2, they have the following constraints of maximum and minimum values:

$$\mathbf{x}_{\max} = [50 \quad 50 \quad 50 \quad 1 \quad 3], \tag{6}$$
$$\mathbf{x}_{\min} = [1 \quad 1 \quad 1 \quad 0.1 \quad 0]. \tag{7}$$

According to the general scheme, the description of PREDICT2 and ES based algorithm proposed in this paper is given by the following steps.

(1) *Population initialization:* This is carried out picking μ individuals at random from the search space Ψ as defined in (4). The initial parent population matrix with $(\mu \times 5)$ dimension is defined by: $\Upsilon = (x_{i,j}, \sigma_{i,j})$ for $i = 1,2,\ldots,\mu$ and $j = 1, 2, \ldots, 5$, where:

$$x_{i,j} = x_{\min,j} + U_{i,j}(0,1)(x_{\max,j} - x_{\min,j}) \tag{8}$$

$$\sigma_{i,j} = \left| x_{i,j} - \left(x_{\min,j} + \frac{x_{\max,j} - x_{\min,j}}{2} \right) \right| \frac{1}{\sqrt{5}} \tag{9}$$

where $U_{i,j}(0,1)$ denotes a random variable of uniform distribution in the interval [01] that is sampled for each parameter. The initial values of the standard deviations are obtained as $\sigma_{i,j} = \frac{\Delta x_{ij}}{\sqrt{5}}$ where Δx_{ij} denotes the estimated parametric distance to the optimum, which should lie in the middle of the parameter search range.

(2) *Recombination:* The kind of recombination used in this paper is the *global discrete recombination* [41], where the offspring solution inherits its components from any of the parents. Thus, each of the μ parents has $1/\mu$ chances of being selected to contribute to each component of the new individual offspring solution. Therefore, a jth new component associated with a qth new individual is produced choosing randomly one element from all positions (μ rows) of the jth column of population matrix, as follows:

$$x_{q,j} = rand\begin{bmatrix} x_{i,j} \\ \vdots \\ x_{\mu,j} \end{bmatrix}; \quad j = 1,\ldots,5; \quad q = 1,\ldots,\gamma \tag{10}$$

$$\sigma_{q,j} = rand\begin{bmatrix} \sigma_{i,j} \\ \vdots \\ \sigma_{\mu,j} \end{bmatrix}; \quad j = 1,\ldots,5; \quad q = 1,\ldots,\gamma, \tag{11}$$

where $rand[.]$ returns an element chosen randomly and γ is the offspring population size.

(3) *Mutation:* Mutation consists of slight perturbations in the parameters of the individual offspring after they have been generated by the recombination procedure. Thus in this stage, the γ offspring are mutated using Gaussian mutation. The $\sigma_{i,j}$ standard deviations are mutated first and then the object variables $x_{i,j}$, according to the following expressions:

$$\sigma'_{q,j} = \sigma'_{q,j}\exp(\beta' N(0,1) + \beta N_q(0,1)); \quad j = 1,\ldots,5; \quad q = 1,\ldots,\gamma, \tag{12}$$

where $N(0,1)$ and $N_q(0,1)$ represents a Gaussian random number with mean 0 and variance, $\beta' = (\sqrt[4]{4n})^{-1}$; $\beta = (\sqrt{2n})^{-1}$.Hence the variations of the parameters m, τ, k and RF are in an integer discrete values form. Then:

$$x'_{q,j} = round\Big(x_{q,j} + \sigma'_{q,j}N(0,1)\Big); \quad j = 1,\ 2,\ 3,\ 5; \quad q = 1,\ldots,\gamma, \tag{13}$$

λ varies in continuous values form. Then:

$$x'_{q,j} = x_{q,j} + \sigma'_{q,j}N(0,1); \quad j = 4; \quad q = 1,\ldots,\gamma \tag{14}$$

(4) *Fitness evaluation and selection:* The fitness function for evaluating the fitness of each individual in the population must be defined. Hence a common identification objective is used to minimize the mismatch between model prediction and the observed data. This paper uses the error prediction between the output of PREDICT2 and the observed data. Thus the prediction result of PREDICT2 associated with each individual in the population is

$$\text{Output}^i = \begin{bmatrix} \hat{y}_{1,i} \\ \vdots \\ \hat{y}_{z,i} \\ \vdots \\ \hat{y}_{nTr,i} \end{bmatrix} = \text{PREDCIT2}(S, nTr, \mathbf{x}^i); \quad i = 1,\ldots,\mu+\gamma. \tag{15}$$

where Output^i is a $(nTr \times 1)$ column vector whose elements $\hat{y}_{z,i}$ are the nTr-step-ahead prediction results using PREDICT2 associated to ith individual. S is $(N - nTr \times 1)$ column vector whose elements are time series data. N is the number of elements of time series input data and nTr is the number of training data points considered.Thus the fitness function considered in this paper is the mean absolute percentage error (MAPE). The fitness value in (%) *Fitness* for ith individual of the offspring population is given by,

$$\text{Fitness}^i = \frac{100}{nTr}\sum_{z=1}^{nTr}\left|\frac{y_{z,i} - \hat{y}_{z,i}}{y_{z,i}}\right|; \quad i = 1,\ldots,\mu+\gamma \tag{16}$$

where $\hat{y}_{z,i}$ is the observed value. The evolution strategy $(\mu+\gamma)$-ES [41] is used. In this class of strategy, after applying the mutation operator, a new batch of offspring is obtained. In this paper elitist selection operator is used. Elitism dictates that the old parent individuals will be pooled together with the new offspring individuals and then the ranking of all $\mu+\gamma$ individuals will be performed according to their fitness value. The best-fitted μ individuals, selected from the pool, will substitute the old parent population.

(5) *Termination:* In order to halt the evolutionary process and to accept the best found individual as the solution to the optimization problem, one or several criteria have to be established. Often, convergence is imposed on evolutionary algorithms by setting an external parameter of a maximum number of generations.

2.4. Summary of the PREDICT2-ES algorithm

The proposed approach PREDICT2-ES has been developed for multi-ahead step prediction (M-ahead step). The input data to the proposed model is a historical time series data (TSD), an $N \times 1$ vector dimension. For example a M-step-ahead prediction using the proposed approach is desired. Then the TSD is divided in two adjoining segments. The first segment is used to reconstruct the state space for predicting the data in the second segment. These two segments are named as the state space reconstruction set (SSR), a $(N - nTr) \times 1$ vector dimension and training set (TRN) a $(nTr \times 1)$ vector dimension, respectively. Notice that at the training stage there is no overfitting problem, since the dimension of the training set vector (nTr) of each individual is much greater than 1. Hence, in this paper is being trained the multi-step-ahead prediction ability of PREDICT2. The optimal value for nTr must be fixed before going to the prediction process. However, the optimal value for nTr must be determined. In this paper an empirical value for nTr is given as $nTr = KM$, where M is the number of steps-ahead desired to forecast, and K is a discrete positive number ranging from 1 to M. After many sensitivity tests varying the K values, it was observed that an appropriate choice for the training dimension set vector (nTr) was twice the desired M-step-ahead prediction. That is, for $K = 2$ the prediction error (MAPE) was smaller than for other values of K. Thus in this paper, $nTr = 2M$, and M is equal to 168 h, which means that $nTr = 336$.

As explained above, in fact the proposed approach is mainly for multi-step (long-range) application forecasting. The one-step-ahead prediction can be considered here as a simplified case, hence the one-step-ahead prediction is a sub problem of the multi-step-ahead prediction.

Next the proposed evolutionary strategy optimization algorithm is then used to tune the prediction parameters so that the calibration set can be predicted from the state space reconstruction set with maximum accuracy.

Fig. 1 shows the flowchart of the M-step-ahead prediction procedure of times series using the proposed PREDICT2-ES approach:

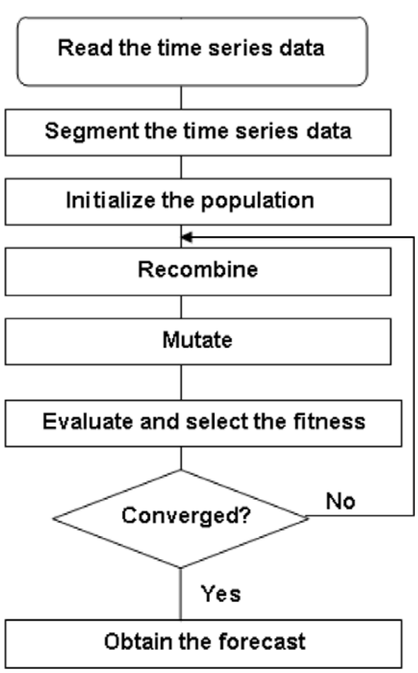

Fig. 1. Flowchart of the proposed PREDICT2-ES approach.

- Read the TSD ($N \times 1$ vector dimension).
- Divide the time series data in two sets: The TRN set is a $2M \times 1$ vector dimension, whose elements are obtained from $N - 2M + 1$ to N position of TSD column vector. The SSR set is a $(N - 2M) \times 1$ column vector whose elements are obtained from 1 to $N - 2M$ position of TSD column vector.
- Obtain optimal parameters to PREDICT2 using the evolutionary strategy proposed algorithm: A maximum number of generations of 100, $\mu = 5$ and $\gamma = 1$ are settled in this paper for all cases. Then, $\hat{y} = \text{TRN}$.
- Update PREDICT2 input: SSR is an $N \times 1$ vector dimension. Then obtain the M-step-ahead prediction using PREDICT2 considering optimal parameters obtained in the above step.

The multi-ahead prediction error of the proposed approach can be evaluated using the M real values.

2.5. Final considerations

In summary, the approach involves the selection of all prediction parameters of a nonlinear chaotic dynamic based predictor PREDICT2 at the forecasting stage rather than at the system characterization stage. A simple and efficient evolutionary strategy is used to find five optimal prediction parameters so that the training set can be predicted from a state space reconstruction set with optimal accuracy.

The proposed hybrid approach PREDICT2-ES is more logical than the standard approach for two reasons. First, system characterization should be relegated to a supporting/ verification role because there are neither necessary nor sufficient conditions available at present to unequivocally identify chaos [38,39]. Existing selection criteria also allow fairly wide latitude in the choice of state space parameters for system characterization. Even if these diverse selection criteria produce similar results, it is still uncertain how such convergence will lead to an accurate prediction. It is more sensitive to use prediction accuracy as a unifying criterion for parameter selection that has direct engineering significance. Second, the standard approach implicitly assumes that a single set of state space parameters is applicable to both system characterization and forecasting [39].

The proposed new training stage performs well regardless the time series follows chaotic or stochastic behavior, or even presents other patterns of dynamics behavior. As a consequence, the proposed approach PREDICT2-ES is capable of effectively capturing the complex dynamic of time series considered. In real time series, this dynamic complex is unknown and it can be any chaotic, stochastic, etc, or a combination of them. However, the technique reported in this paper is restricted to predict times series including no strong spikes. If the times series under consideration suffers from cyclical occurrence of spikes, specific procedures to estimate such strong spikes are required.

An additional advantage of the proposed approach is its running on-line manner, since the search of the optimal parameters and prediction are executed automatically. It is an improved manner in comparison with the ARIMA and ANN-based models, where the choice of their input parameters are carried out off-line, following qualitative/experience-based recipes.

3. Experiments, results and comparisons

The proposed approach PREDICT2-ES is applied for multi-step ahead prediction of the electricity demand of the New England and Alberta market. The New England data set consists of hourly electricity loads from January 1, 2002 through December 31, 2002 [42], whereas Alberta data set consists of hourly electricity loads from January 1, 2004, to December 31, 2004 [44]. PREDICT2-ES is also applied for multi-step ahead prediction of the electricity price of the New England and Spanish market. The New England data set consists of hourly electricity prices from January 1, 2002 through December 31, 2002 [43], whereas Spanish data set consists of hourly electricity prices from January 1, 2004 to December 31, 2004 [44]. The results of PREDICT2-ES are compared with the best results obtained by ARIMA and ANN models by using the commercial software Statistica [45].

3.1. ARIMA

The autoregressive integrated moving average (ARIMA) methodology developed by Box and Jenkins [46] has gained enormous popularity in many areas and research practice confirms its power and flexibility. The methodology is recursive and requires some expertise and knowledge to obtain the right model. In summary, the ARIMA methodology has the following main processes:

- *Identification:* Consists of identifying which models best fit the time series behavior. In ARIMA $(p, d, q)(P, D, Q)S, d$ and D are the order of the nonseasonal and seasonal differences, respectively, p and P represents the order of the nonseasonal and seasonal autoregressive term, and q and Q represent the order of the nonseasonal and seasonal moving average terms. The seasonal difference is represented by the S lag. The main tools used in this process are the autocorrelation (ACF) and partial autocorrelation (PACF) functions.
- *Verification:* Consists of evaluating whether the estimated model is adequate to describe the time series behavior. Several different models were tested for the electricity time series using the program Statistica [45]. In all of the obtained models a daily seasonality of lag = 24 and a weekly difference of lag = 168 was obtained from the ACF and PACF graphs. A model ARIMA (1,1,0) (0,0,1) was enough to present a reasonable margin of error for forecasting most of the time. In general, the ARIMA model gives better results when compared to regression models but it has a drawback related to the time-varying variance, as well as a problem when heteroscedasticity is present.

3.2. ANN

The following characteristics were addressed for the time series prediction and a great number of simulations were necessary to estimate reasonable ANNs for this specific problem. Design of experiments (DOE) was the main tool to conduct the training of the ANNs [47]. The procedure adopted in this paper related to the use of ANN, can be summarized as:

Type of problem: Generally a time series forecasting problem using ANN can be addressed with two different strategies (according to the method of training and the number of variables to be considered for the ANN). The first strategy consists of using ANN for a specific problem of time series. In this case, the ANN training considers past values as input and forecast values as response. Here, the definition of seasonality is a very important point for the convergence of the ANN training. Usually the number of neurons of the input layer is determined by seasonality. The second method consists of making the series prediction using the data as a regression problem. In this case, explanatory variables are used. Usually the number of past values is small, not directly related to the seasonality lag, but there are extra time series that try to explain the presence of seasonality. The simulations pursued the best ANN for most of the time series, but the outcome showed that no single ANN could represent the behavior of all time series. For a great number of points, the seasonality is better estimated when an ANN is used as a time series problem.

- *Stopping Criteria:* Two criteria were tested on the ANN training – Minimal error and error against diversity. Minimal error has the tendency to be better once the ANN is well defined. Error against diversity is indicated when the simulation is exploratory.
- *ANN's architecture:* Multilayer feed forward and radial basis function ANNs were tested. It is hard to generalize the best architecture, and trial and error is in fact a common procedure in spite of a great number of theoretical procedures available.
- *Sampling techniques:* The respective amounts of data for training, selection and testing were established under the rule (2:1:1).
- *Complexity:* The number of layers and neurons is one of the most difficult problems to be estimated when training an ANN. A mechanism called Grid Search based on Genetic Algorithm was utilized most of the time to optimize the mentioned number for each time series.
- *Seasonality:* The number of lags for training the ANNs can be obtained using fast Fourier transform and autocorrelation functions. Usually, lags 24 and 168, representing a daily and week seasonality were considered.
- *Activation functions:* For all time series the sigmoid function was considered the standard for all time series.
- The ANNs were obtained using the software Statistica with its user-friendly toolbox of neural network. The software was considered an excellent resource for short-term forecasting (especially when automated features are necessary). Some features such as Grid Search and Data Mining are valuable tools for estimating the ANN.

3.3. Results and comparisons

Table 1 presents the comparative electricity market forecasting results employing the proposed PREDICT2-ES approach and the traditional ANN and ARIMA methods. The hourly data set used for training and testing consider the four seasons and an exclusive 168 step ahead time series representing a week interval during each season. These 168 points were not used during the training phase. In attempting to make a fair comparison, the fourth week of February, May, August, and October were selected. This accounting for reality results in an uneven accuracy distribution throughout the year. To assess the prediction capacity of the PREDICT2-ES, ARIMA and ANN model, the average prediction error was computed using the traditional mean absolute percentage error (MAPE), given as:

$$\mathrm{MAPE}\,(\%) = \frac{100}{H}\sum_{h=1}^{H}\left|\frac{P_h - \widehat{P}_h}{P_h}\right|, \tag{17}$$

where $H = 168$, and P_h and $\widehat{P}_h$ are the respective actual and forecasted hourly load or prices.

The Model column reveals the obtained prediction model for the three employed methods, as follows:

(i) The ARIMA model reveals the (p, d, q) (P, D, Q) parameters already described on Section 3.1.
(ii) The profile of the network architecture is described by the form In:H–H–H:Ou, where In is the number of input variable, Ou the number of output variables, and H the number of units in each layer. Example: 2:4–6–3:1 indicates a network with 2 input variables, 1 output variable, 4 input neurons, 6 hidden neurons, and 3 output neurons. For a time series network, the steps factor is prep ended to the profile, and signified by an s. Example: s10 1:10–2–1:1 indicates a time series network with steps factor (lagged input) 10.
(iii) The PREDICT-ES reveals the optimal parameters $\mathbf{x} = (m \quad \tau \quad k \quad \lambda \quad RF)$ described on Section 2.2.

Table 1
Weekly forecasting error comparison for the analyzed cases.

Electricity market	Data type	Season	Data set training	Data set test	Method	Model	MAPE
New England	Load	Winter	1/1/2002–2/15/2002	2/16/2002–2/22/2002	ARIMA	(1,1,3)(1,0,0)	5.560
					ANN	RBF s168 1:168-261-1:1	7.470
					PREDICT2-ES	(20, 10, 15, 1, 3)	2.685
		Spring	1/1/2002–5/10/2002	5/11/2002–5/17/2002	ARIMA	(1, 1, 3) (5, 0, 0)	2.740
					ANN	MLP s168 1:168-7-1:1	5.330
					PREDICT2-ES	(21, 25,15, 1, 3)	1.765
		Summer	1/1/2002–8/15/2002	8/16/2002–8/22/2002	ARIMA	(1, 1, 3) (6, 0, 0)	14.430
					ANN	MLP s168 1:168-7-1:1	9.150
					PREDICT2-ES	(30, 17, 20, 1, 3)	4.156
		Fall	1/1/2002–10/25/2002	10/25/2002–10/31/2002	ARIMA	(1, 1, 2) (3, 0, 0)	3.790
					ANN	MLP s168 1:168-8-1:1	3.200
					PREDICT2-ES	(28, 7, 8, 0.865640, 3)	2.087
New England	Price	Winter	1/1/2002–2/15/2002	2/16/2002–2/22/2002	ARIMA	(1,1,0)(1,0,1)	13.360
					ANN	MLP 168 1:168-3-1:1	14.120
					PREDICT2-ES	(17, 6, 9, 0.834833, 0)	8.127
		Spring	1/1/2002–5/10/2002	5/11/2002–5/17/2002	ARIMA	(1, 1, 3) (2, 0, 2)	13.310
					ANN	RBF 168 1:168-40-1:1	13.280
					PREDICT2-ES	(12, 20, 13, 0.991530, 2)	10.632
		Summer	1/1/2002–8/15/2002	8/16/2002–8/22/2002	ARIMA	(1, 1, 1)(1, 0, 2)	38.800
					ANN	MLP 168 1:168-9-1:1	27.130
					PREDICT2-ES	(10, 3, 13, 0.738849, 1)	15.103
		Fall	1/1/2002–10/25/2002	10/25/2002–10/31/2002	ARIMA	(1, 1, 1) (1, 0, 0)	12.740
					ANN	RBF s168 1:168-12-1:1	9.740
					PREDICT2-ES	(23, 11, 11, 0.968260, 2)	9.418
Alberta	Load	Winter	1/1/2004–2/15/2004	2/16/2004–2/22/2004	ARIMA	(1, 1, 1)(3, 0, 0)	1.440
					ANN	MLP s168 1:168-8-1:1	2.130
					PREDICT2-ES	(30, 20, 12, 1, 2)	0.945
		Spring	1/1/2004–5/10/2004	5/11/2004–5/17/2004	ARIMA	(1, 1, 1) (1, 0, 1)	1.070
					ANN	RBF 168 1:168-25-1:1	1.100
					PREDICT2-ES	(26, 15 4, 0.894527, 2)	0.812
		Summer	1/1/2004–8/15/2004	8/16/2004–8/22/2004	ARIMA	(1, 1, 0) (2, 0, 0)	2.540
					ANN	MLP s168 1:168-10-1:1	2.130
					PREDICT2-ES	(21, 10, 3, 1, 3)	1.272
		Fall	1/1/2004–10/25/2004	10/25/2004–10/31/2004	ARIMA	(1, 1, 0) (3, 0, 0)	1.500
					ANN	RBF s168 1:168-24-1:1	0.820
					PREDICT2-ES	(15, 19, 5, 0.938630, 2)	0.745
Spain	Price	Winter	1/1/2004–2/15/2004	2/16/2004–2/22/2004	ARIMA	(1,1,0)(1,0,0)	15.030
					ANN	RBF s168 1:168-276-1:1	10.490
					PREDICT2-ES	(23, 2, 13, 0.944092, 1)	7.638
		Spring	1/1/2004–5/10/2004	5/11/2004–5/17/2004	ARIMA	(1, 1, 1) (1, 0, 1)	14.790
					ANN	RBF s168 1:168-284-1:1	25.450
					PREDICT2-ES	(30, 29, 11, 1, 1)	10.519
		Summer	1/1/2004–8/15/2004	8/16/2004–8/22/2004	ARIMA	(1, 1, 1) (2, 0, 0)	8.730
					ANN	MLP s168 1:168-7-1:1	8.720
					PREDICT2-ES	(29, 22, 13, 0.941816, 2)	8.691
		Fall	1/1/2004–10/25/2004	10/25/2004–10/31/2004	ARIMA	(2, 1, 0) (2, 0, 0)	21.360
					ANN	RBF s168 1:168-284-1:1	11.180
					PREDICT2-ES	(26, 18, 10, 0.998016, 2)	10.797

From Table 1 it was found that the proposed approach PREDICT2-ES was able to outperform ARIMA and ANN models for all scenarios considering MAPE.

The well-known statistical paired t hypotheses test is appropriated for testing the mean difference between paired observations when the paired differences follow a normal distribution. It is used to compute a confidence interval and perform a hypothesis test of the mean difference between paired observations in the population. A paired *t*-test matches responses that are dependent or related in a pairwise manner. This matching allows one to account for variability between the pairs usually resulting in a smaller error term, thus increasing the sensitivity of the hypothesis test or confidence interval. *p*-Values are often used in hypothesis tests where one either accepts or rejects a null hypothesis. The *p*-value represents the probability of making a Type 1 error, or rejecting the null hypothesis when it is true. The smaller the *p*-value, the smaller is the probability that you would be making a mistake by rejecting the null hypothesis. A cutoff value often used is 0.05, that is, reject the null hypothesis when the *p*-value is less than 0.05. Here, the null hypothesis is that the methods are equal and the paired *t*-test (one tail) presents p-value < 0.01 comparing PREDICT2-ES against ANN and also against ARIMA. ANN and ARIMA were found not to be statistically different using a paired *t*-test (p-value > 0.1).

The boxplots display graphically patterns of variation. It is used to identify families of variation, such as variation within a subgroup, between subgroups. The spread gives you some idea of the variation. Figs. 2 and 3 show the boxplots for the considered markets and seasons. As shown in those figures, PREDICT2-ES error was significantly smaller in terms of MAPE than those obtained by ARIMA and ANN models in all markets and seasons. It was noted that spot prices caused mild spikes and the time series did not follow a well-defined pattern, making forecasting very complicated. Despite that, the forecasting results of PREDICT2-ES were more accurate than the ARIMA and ANN methods. In all seasons PREDICT2-ES method outperformed ANN and ARIMA. For the fall period ANN and PREDICT2-ES present similar results. It was also noticed that the summer week cases presented unstable behavior, making forecasting hard. Nevertheless one can see how the forecast obtained from the PREDICT2-ES was able to successfully predict the trend of the 168-step-ahead time series.

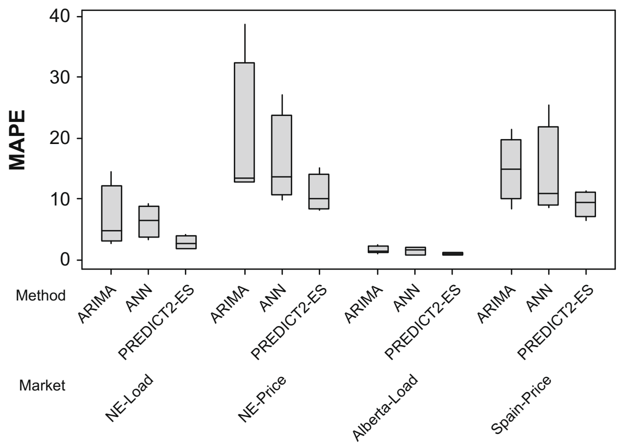

Fig. 2. Boxplots of MAPE of markets.

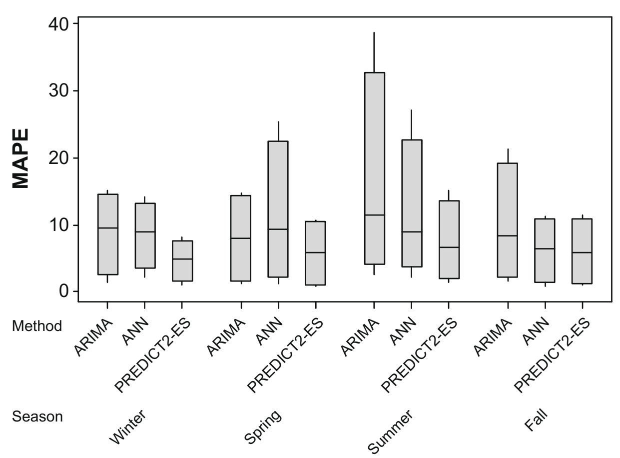

Fig. 3. Boxplots of MAPE of seasons.

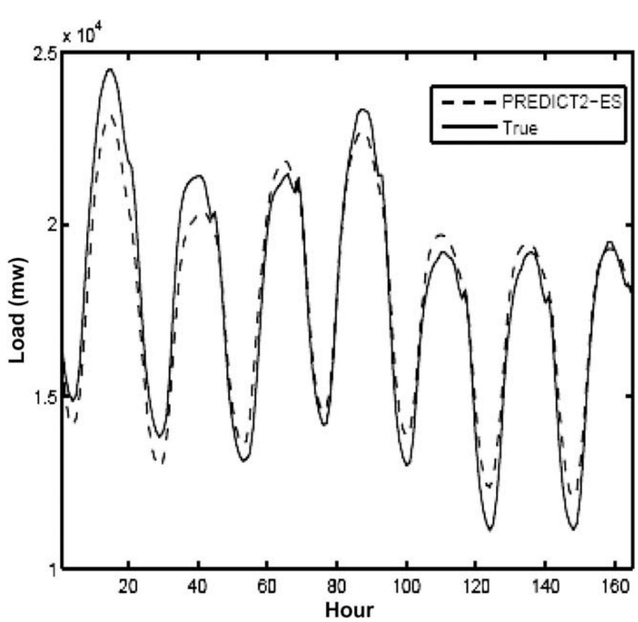

Fig. 4. Summer week: Actual loads and PREDICT2-ES estimates in megawatt- New England Market.

In Fig. 4, it can be seen that the summer week load case of New England market was also unstable, making forecasting hard. Despite that, one can see how the load forecast obtained from the PREDICT2-ES approach is able to successfully predict the trend of the 168-step ahead real load. Even so, the performance of the PREDICT2-ES technique for the summer is very good, with a weekly error below 4.156%. Weekly error for the ARIMA and ANN technique is below 14.430% and 9.150%, respectively. Analyzing the different plots represented in Figs. 2 and 3 and Table 1, it can be concluded that the performance of the proposed technique is superior to the performance of the ARIMA and ANN methods. These experimental results confirm the notion that in this case the combined chaotic dynamic model and evolutionary strategy optimization capture more effectively the demand and prices time series dynamics. The superior predicting behavior of the proposed technique is apparent in all the weeks analyzed.

The proposed PREDICT2-ES was implemented in Matlab language. The CPU running time, including training and forecasting, was below two minutes for each one of the cases presented. All the results were obtained on a 2.80 GHz Pentium IV PC with 960 MB of RAM.

As a further investigation we have to compare the method against others ANN methods besides MLPs and RBF's, used in this study.

4. Conclusions

A hybrid approach which combines nonlinear chaotic dynamics and evolutionary strategy techniques was presented and applied to short-term load and day-ahead electricity price forecasting. The main idea is developing a new training or identification stage to a nonlinear chaotic dynamic based predictor, in such away that the time series modeling and forecasting are improved significantly. In this training stage, five optimal parameters for a nonlinear chaotic dynamic based predictor are searched through an optimization model based on evolutionary strategy. Hence a common identification objective is to minimize the mismatch between model prediction and observed data, and then it is used as the objective function in the optimization model based on evolutionary strategy.

The proposed approach PREDICT2-ES is capable of effectively capturing the complex dynamic (without strong spikes) of time series, since in real time series, this dynamic complex is unknown i.e. it can be any chaotic, stochastic, etc., or a combination of them. An additional advantage of the proposed approach is its running on-line manner, since the search of the optimal parameters and prediction are executed automatically. It is an improved manner in comparison with the ARIMA and ANN-based models, where the choice of their input parameters are carried out off-line, following qualitative/experience-based recipes.

Many cases study and comparisons with ARIMA and ANN model were presented and the results showed that the proposed approach provides a more accurate and effective forecasting.

Acknowledgements

The authors would like to thank the support of the Brazilian Institutions of CAPES (Project 023/05), CNPq and FAPEMIG.

References

[1] Mandal P, Senjyu T, Funabashi T. Forecast several hours ahead electricity prices and loads in deregulated markets. Energy Convers Manage 2006;47:2128–42.
[2] Conejo AJ, Plazas MA, Espinola R, Molina AB. Day-ahead electricity price forecasting using the wavelet transform and ARIMA models. IEEE Trans Power Syst 2005;20(2):1035–42.

[3] Conejo A, Contreras J, Espínola R, Plazas M. Forecasting electricity prices for a day-ahead pool-based electric energy market. Int J Forecasting 2005;21:435–62.
[4] Metaxiotis K, Kagiannas A, Askounis D. Artificial intelligence in short-term electric load forecasting: a state of-the-art survey for the researcher. Energy Convers Manage 2003;4:1525–34.
[5] Dillon TS, Morsztyn K, Phua K. Short term load forecasting using adaptive pattern recognition and self-organizing techniques. In: Proceedings 5th world power system computation conference (PSCC-5). Cambridge; 1975. p. 1–15.
[6] Sestito S, Leung S, Dillon TS. Short term load forecasting using an adaptive neural network. J Electr Power Energy Syst 1991;13(4):186–92.
[7] Choi J-G, Park J-K, Kim K-H, Kim J-C. A daily peak load forecasting system using a chaotic time series. In: Proceedings of the international conference on intelligent systems applications to power systems ISAP; 1996. p. 283–7.
[8] Mori H, Urano S. Short-term load forecasting with chaos time series analysis. In: International conference on intelligent systems applications to power systems ISAP; 1996. p. 283–7.
[9] Iokibe T, Fujimoto Y, Kanke M, Suzuki S. Short-term prediction of chaotic time series by local fuzzy reconstruction method. J Intell Fuzzy Syst 1997;5:3–21.
[10] Drezga I, Rahman S. Phase-space based short-term load forecasting for deregulated electric power industry Neural Networks. Int Joint Conf IJCNN 1999;5:3405–9.
[11] Michanos SP, Tsakoumis AC, Fessas P, Vladov SS, Mladenov VM. Short-term load forecasting using a chaotic time series. Int Symp Signals, Circuit Syst SCS 2003;2:437–40.
[12] Kawauchi H, Sasaki H, Ishikawa F, Sugihara H. A development of real time load forecasting system based on chaos theory. In: 6th international conference on advances in power system control, operation and management, vol. 2; 2003. p. 541–6.
[13] Jiang C, Wang C, Ma Y. Short-term load nonlinear forecasting with high-embedded dimensions using wavelet decomposing and chaos theory. The IASTED-2004 Conf Eur Power Energy Syst 2004:1–6.
[14] Zhang H-Shan WS. Power load forecasting with least squares support vector machines and chaos theory. Int Conf Power Syst 2005:1–6.
[15] Liao G-C. Hybrid chaos search genetic algorithm and meta heuristics method for short-term load forecasting. Electr Eng 2006;88:1–6.
[16] Liu Z, Xie X, Zhang D, Liu H. Local partial least squares multi-step model for short-term load forecasting. IEICE Trans Fundam 2006;E89-A:1–6.
[17] Tian L, Noore A. Short-term load forecasting using optimized neural network with genetic algorithm. In: 8th international conference on probabilistic methods applied to power systems. Iowa: Iowa State University Ames; September 2004. p. 12–6.
[18] Li Y, Li B, Fang T. Short-term load forecast based on fuzzy wavelet support vector machines. In: Proceedings of the 5th world congress on intelligent control and automation. Hangzhou, PR China; 15–19 June 2004.
[19] Contreras J, Espínola R, Nogales FJ, Conejo AJ. ARIMA models to predict next-day electricity prices. IEEE Trans Power Syst 2003;18(3):1014–20.
[20] Nogales FJ, Contreras J, Conejo AJ, Espínola R. Forecasting next-day electricity prices by time series models. IEEE Trans Power Syst 2002;17(2):342–8.
[21] Aggarwal SK, Saini LM, Kumar A. Electricity price forecasting in deregulated markets: A review and evaluation. Int J Electr Power Energy Syst 2009;31(1):13–22.
[22] Szkuta BR, Sanabria LA, Dillon TS. Electricity price short-term forecasting using artificial neural networks. IEEE Trans Power Syst (PES) 1999;14(3):851–7.
[23] Rodríguez CP, Anders GJ. Energy price forecasting in the Ontario competitive power system market. IEEE Trans Power Syst 2004;19(1):366–74.
[24] Pao H-T. Forecasting electricity market pricing using artificial neural networks. Energy Convers Manage 2007;48(3):907–12.
[25] Garcia RC, Contreras J, van Akkeren M, Garcia JBC. GARCH forecasting model to predict day-ahead electricity prices. Trans Power Syst 2005;20(2):867–74.
[26] Pingan Z, Xiaohong G. Fuzzy modeling for electricity market price forecasting. In: Proceedings of the 3rd world congress on intelligent control and automation, vol. 3; 28 June–2 July 2000. p. 2262–6.
[27] Kim C-I, Yu I-K, Song YH. Prediction of system marginal price of electricity using wavelet transform analysis. Energy Convers Manage 2002; 43:1839.
[28] Amjady N, Keynia F. Day ahead price forecasting of electricity markets by a mixed data model and hybrid forecast method. Int J Electr Power Energy Syst 2008;30(9):533–46.
[29] Yang H, Duan X. Chaotic characteristics of electricity price and its forecasting model. IEEE-CCECE Can Conf Electr Comput Eng 2003;1:659–62.
[30] Guangjian L, Sangao H, Junliang D. The chaotic property of system marginal price and its forecasting. Proc CSEE 2003;23(5):6–8.
[31] Wu W, Zhou J-Z, Yu J, Zhu C-J, Yang J-J. Prediction of spot market prices of electricity using chaotic time series. In: Proceeding of the 3rd international conference on machine learning and cybemetics. Shanghai; 26–29 August 2004. p. 1–6.
[32] Liu Z, Yang H, Lai M. Electricity price forecasting model based on chaos theory. In: Power engineering conference, IPEC; 2005. p. 1–5.
[33] Varadan V, Leung H, Bossé É. Dynamical Model Reconstruction and Accurate Prediction of Power-Pool Time Series. IEEE Trans Instrum Meas 2006;55(1):1–10.
[34] Kantz H, Schreiber T. Nonlinear time series analysis. Cambridge University Press; 2002.
[35] Abarbanel HDI, Parlitz U. Nonlinear analysis of time series data. In: Schelter Björn, Winterhalder Matthias, Timmer Jens, editors. Handbook of time series analysis. Weinheim: Wiley-VCH Verlag; 2006. p. 5–33.
[36] Merkwirth Christian, Parlitz Ulrich, Lauterborn Werner. Fast exact approximate nearest neighbor searching for nonlinear signal processing. Phys Rev E 2000;62(2):2089–97.
[37] McNames J. Innovations in local modeling for time series prediction. PhD thesis, Stanford University; May 2000.
[38] Babovic V, Keijzer M, Stefansson M. Optimal embedding using evolutionary algorithms. In: Proceedings of 4th international conference on hydroinformatics. Iowa City, Iowa: Iowa Institute of Hydraulic Research; 2000. p. 1–6.
[39] Phoon KK, Islam MN, Liaw CY, Liong SY. Practical inverse approach for forecasting nonlinear hydrological time series. J Hydrol Eng 2002;7(2): 116–28.
[40] Merkwirth C, Parlitz U, Wedekind I, Lauterborn W. Matlab software package for nonlinear time series analysis tstool. Advances in enzymology, vol. 2. Germay; 2008. Available from: http://www.physik3.gwdg.de/tstool/.
[41] Bäck T. Evolutionary algorithms in theory and practice. 1st ed. New York: Oxford University Press; 1996.
[42] Indepent system operator – ISO New England. Available from: http://www.iso-ne.com.
[43] The Alberta electric system operator. Available from: http://www.aeso.ca.
[44] Compañía Operadora del Mercado Español de Electricidad (Spanish market operator). Available from: http://www.omel.es.
[45] Data mining, statistical analysis, quality control – STATISTICA. Software, StatSoft Inc.; 2005. Available: http://www.statsoftinc.com/
[46] Box GEP, Jenkins GM. Time series analysis: forecasting and control. San Francisco, CA: Holden-Day; 1976.
[47] Balestrassi PP, Popova E, Paiva AP, Marangon-Lima JW. Design of experiments on neural network's training for nonlinear time series forecasting. J Neurocomput; 2009. doi:10.1016/j.neucom.2008.02.002.