Security Constrained Unit Commitment Based Load and Price Forecasting Using Evolutionary Optimized LSVR

Ehab E. Elattar, *Member, IEEE*, and Tamer A. Farrag

Abstract— In this paper, a local predictor approach based on proven powerful regression algorithm which is support vector regression (SVR) combined with space reconstruction of time series is introduced. In addition, real value genetic algorithm (GA) has been utilized in the proposed method for optimization of the parameters of the SVR. In the proposed approach, the embedding dimension and the time delay constant for the load and price data are computed firstly, and then the continuous load and price data are used for the phase space reconstruction. Subsequently, the reconstructed data matrix is subject to the local prediction algorithm. Then the forecasted loads and price are fed into IEEE 30 bus test system for security constraint unit commitment to show the reactions of unit commitment to load and price forecasting errors. The proposed model is evaluated using real world dataset. The results show that the proposed method provides a much better performance in comparison with other models employing the same data.

Index Terms— Load forecasting, price forecasting, local predictors, security constrained unit commitment, support vector regression, genetic algorithm, state space reconstruction.

1 Introduction

SHORT term load forecasting (STLF) is a vital part of the operation of power systems. STLF aims to predict electric loads for a period of minutes, hours, days, or weeks. STLF has always been a very important issue in economic and reliable power systems operation such as unit commitment, reducing spinning reserve, maintenance scheduling, etc.

Several STLF methods including traditional and artificial intelligence-based methods have been proposed during the last four decades. The relationship between electric load and its exogenous factors is complex and nonlinear, making it quite difficult to be modeled through traditional techniques such as linear or multiple regression [1], autoregressive moving average (ARMA), exponential smoothing methods [2], Kalman-filter-based methods [3], etc. On the other hand, various artificial intelligence techniques were used for STLF; among these methods, artificial neural networks (ANNs) have received the largest share of attention. The ANNs that have been successfully used for STLF are based on multilayered perceptrons [4]. The neural fuzzy network has also been used for load forecasting [5]. Radial basis functions (RBFs) [6] have been also used for day-ahead load forecasting, giving better results than that of the conventional neural networks.

Accurate forecasting of the electricity price has become a very valuable tool. This is because of the upheaval of deregulation in electricity market. Short-term price forecasting in a competitive electricity market is still a challenging task because of the special electric price characteristics [7], [8], such as high-frequency, non-stationary behavior, multiple seasonality, calendar effect, high volatility, high percentage of unusual prices, hard non-linear behavior etc.

In the literature, several techniques for short-term electricity prices forecasting have been reported, namely traditional and AI-based techniques. The traditional techniques include autoregressive integrated moving average (ARIMA) [9], wavelet-ARIMA [10] and mixed model [11] approaches. Although, these techniques are well established to have good performance, they cannot always represent the non-linear characteristics of the complex price signal. Moreover, they require a lot of information, and the computational cost is very high.

On the other hand, AI-based techniques have been used by many researchers for the price forecasting in electricity markets. These methods can deal with the non-linear relation between the influencing factors and the price signal, therefore the forecasting precision is raised. These techniques include neural network (NN) [12], radial basis function NN [13], fuzzy neural network (FNN) [14] and hybrid intelligent system (HIS) [15]

Recently, SVR [16], [17] has also been applied successfully to STLF and price forecasting. SVR replaces the empirical risk minimization which is generally employed in the classical methods such as ANNs, with a more advantageous structural risk minimization principle. SVR has been shown to be very resistant to the over fitting problem and give a high generalization performance in forecasting problems [18].

All the above techniques are known as global predictors in which a predictor is trained using all data available but give a prediction using a current data window. The global predictors suffer from some drawbacks which are discussed in our previous work [19], [20]. To overcome these drawbacks, the local

- *This work was supported byTaif University, KSA under grant 3267-435-1.*
- *The Authors are with the department of the Electrical Engineering, Faculty of Engineering, Taif University, Kigdom of Saudi Arabia.*
- *Ehab E. Elattar on leave from the Department of Electrical Engineering, Faculty of Engineering, Menofia University, Shebin El-Kom, Egypt (e-mail: dr.elattar10@yahoo.com).*
- *Tamer A. Farrag is on leave from the Department of Communications and Electronics, Misr high institute of Engineering and Technology, Egypt.*

ISSN 2229-5518

SVR predictor is proposed in our previous work [19]–[21] and can be used to solve the STLF and price forecasting problem.

Phase space reconstruction is an important step in local prediction methods. The traditional time series reconstruction techniques usually use the coordinate delay (CD) method [22] to calculate the embedding dimension and the time delay constant of the time series [23].

Although local SVR (LSVR) method gives good prediction accuracy when it is applied to STLF and price forecasting, it has a serious problem. This problem is that there is a lacking of the structural methods for confirming the selection of SVR's parameters efficiently. So, in this paper, a local predictor approach based on proven powerful regression algorithm which is SVR combined with space reconstruction of time series is introduced. In addition, real value genetic algorithm (GA) has been utilized in the proposed method for optimization of the parameters of the SVR. The proposed algorithm is called evolutionary optimized LSVR (EOLSVR).

Unit commitment problem (UC) is a nonlinear, mixed integer combinatorial optimization problem. The UC problem is the problem of deciding which electricity generation units should be scheduled economically in a power system in order to meet the requirements of load and spinning reserve. It is a difficult problem to solve in which the solution procedures involve the economic dispatch problem as a sub-problem. Since UC searches for an optimum schedule of generating units based on load forecasting data, the improvement of load forecasting is first step to enhance the UC solution [24].

In this paper, we propose security constrained unit commitment (SCUC) method to reduce the production cost by combining load and price forecasting with UC problem. First, short-term loads and price are forecasted using EOLSVR, local SVR and local RBF models. Then UC problem is solved using the dynamic programming method.

We have chosen the historical data for the South Australia electricity market, which includes the power demand and price for the period of 2003-2005. Historical weather data was collected from Macquarie University Web Site. Then the forecasted loads and price are fed into IEEE 30 bus test system for unit commitment to show the reactions of unit commitment to forecasting errors.

The paper is organized as follows: Section 2 review the time series reconstruction method. Section 3 gives a brief discribtion about GA. A review of the SCUC problem and its formulation are presented briefly in Section 4. The proposed method is presented in details in Section 5. Applications and simulations for load and price forecasting and UC problem are given in Sections 6. Finally, Section 7 concludes the work.

2 Time Series Reconstruction

Nonlinear time series analysis and prediction have become a reliable tool for the study of complex time series and dynamical systems. A commonly used tool is the phase space reconstruction technique which stems from the embedding theorem developed by Takens and Sauer [22], [25]. It illustrates clearly the phase space trajectory of a time series in the embedded space instead of the trajectory in the time domain. The theorem regards an 1-dimensional time series *x(t)* for *t = 1, 2, ...,N* as compressed higher dimensional information and, thus, its features can be extracted by extending *x(t)* to a vector *X(t)* in a d-dimensional space as follows:

$$X(t) = \left[x(t), x(t-m), x(t-2m), ..., x(t-(d-1)m)\right] \quad (1)$$

where d denotes the embedding dimension of the system and m is the delay constant. Based on Takens' theorem [22], to obtain a faithful reconstruction of the dynamic system, the embedding dimension must satisfy $d_2 \geq D_a + 1$, where D_a is the dimension of the attractor. In order to obtain an appropriate model reconstruction, it is necessary to estimate d and m.

The correlation dimension method [26] is the most popular method for determining d because of its computational simplicity. The mutual information method proposed in [27] usually provides a good criterion for the selection of m. In general, the proper value of m corresponds to the first local minimum of mutual information. In this paper, the correlation dimension method [26] and the mutual information method [27] are used to calculate d and m respectively. The details of how to choose the proper values of d and m using these two methods have been reported in [19].

3 Genetic Algorithm

The GA is a search algorithm for optimization, based on the mechanics of natural selection and genetics [28], [29]. The GA is able to search very large solution spaces efficiently by providing a lower computational cost, since they use probabilistic transition rules instead of deterministic ones. GA has a number of components or operators that must be specified in order to define a particular GA. The most important components are representation, fitness function, selection method, crossover, mutation and termination.

The GA starts with an initial population of individuals (generation) which are generated randomly. Every individual (chromosome) encodes a single possible solution to the problem under consideration. The fittest individuals are selected by ranking them according to a pre-defined fitness function, which is evaluated for each member of this population. The individuals with high fitness values therefore represent better solution to the problem than individuals with lower fitness values.

There are many different selection operators presented by some researchers such as stochastic sampling with replacement"roulette wheel selection" and tournament selection [30]. Following this initial process, the crossover and mutation operations are used where the individuals in the current population produce the children (offspring). The idea behind the crossover operator is to combine useful segments of different parents to form an offspring that benefits from advantageous bit combinations of both parents [31]. While, by mutation, individuals are randomly altered. These variations (mutation steps) are mostly small [31]. Normally, offspring are mutated after being created by crossover. It is intended to prevent premature convergence and loss of genetic diversity. A new population of individuals (generation) is then formed from the individuals in current population and the children. This new population becomes the current population and the iterative cycle is repeated until a termination condition is reached [28].

4 Security Constraint Unit Commitment (SCUC)

The objective of security-constrained unit commitment (SCUC) discussed in this work is to obtain a unit commitment schedule at minimum production cost without compromising the system reliability. The reliability of the system is interpreted as satisfying two functions: adequacy and security. In several power markets, the independent system operator ISO plans the day-ahead schedule using (SCUC)[32], [33].

The traditional unit commitment algorithm determines the unit schedules to minimize the operating costs and satisfy the prevailing constraints such as load balance, system spinning reserve, ramp rate limits, fuel constraints, multiple emission requirements and minimum up and down time limits over a set of time periods. The scheduled units supply the load demands and possibly maintain transmission flows and bus voltages within their permissible limits [34]. However, in circumstances where most of the committed units are located in one region of the system, it becomes more difficult to satisfy the network constraints throughout the system.

Mathematically, the objective function, or the total operating cost of the system can be written as follows [32], [33]:

$$\min_{P_i^t u_{it}} \left(\sum_{t=1}^{T} \sum_{i=1}^{N} u_i^t \left[F_i\left(P_i^t\right) + S_i^t \left(1 - u_i^{t-1}\right) \right] \right) \tag{2}$$

where P_i^t is the output power of unit i at period t, u_i^t is the commitment state of unit i at period t, $F_i\left(P_i^t\right)$ is the fuel cost of unit i at output power P_i^t, S_i^t is the start up price of unit i at period t, N is the number of generating unit and T is the total number of scheduling periods.

The constraints are as follows:

Power balance:

$$\sum_i^N u_i^t P_i^t = D^t \tag{3}$$

where D_t is the customers' demand in time interval t.

Generating limits: These limits define the region within which a unit must be dispatched.

$$u_i^t P_i^{\min} \le P_i^t \le u_i^t P_i^{\max} \tag{4}$$

Minimum up time: Once the unit is committed, it must be kept running for certain number of hours, called the minimum up time, before allowing turning it off. This can be formulated as follows:

$$\left(X_{on,i}^{t-1} - T_i^{up}\right)\left(u_i^{t-1} - u_i^t\right) \ge 0$$
$$X_{on,i}^{t} = \left(X_{on,i}^{t-1} + 1\right) u_i^t \tag{5}$$

where, $X_{on,i}^t$ is the number of hours the unit has been on line and T_i^{up} is the minimum up time.

Minimum down time: Once the unit is turned off, it is not allowed to be brought online again before spending certain number of hours called minimum down time. This can be formulated as follows:

$$\left(X_{off,i}^{t-1} - T_i^{down}\right)\left(u_i^t - u_i^{t-1}\right) \ge 0$$

$$X_{off,i}^{t} = \left(X_{off,i}^{t-1} + 1\right)\left(1 - u_i^t\right) \tag{6}$$

where $X_{off,i}^t$ is the number of hours the unit has been off line and T_i^{down} is the minimum down time.

Spinning reserve: It can be modeled as follows:

$$\sum_i^N u_i^t P_i^{\max} \ge D^t + R^t \tag{7}$$

where R^t is the spinning reserve requirements.

Transmission flow limit from bus k to bus m:

$$P_{km}^{\max} \le P_{km}(t) = f\left(P(t), \varphi(t)\right) \le p_{km}^{\max} \tag{8}$$

where $P(t)$ is the real power generation vector and $\varphi(t)$ is the phase shifter control vector at time T.

The Start up cost which can be modeled by the following form:

$$S_i^t = \begin{cases} HS^i, & \text{if } X_{off,i}^t \le T_i^{down} + CH^i \\ CS^i, & \text{if } X_{off,i}^t > T_i^{down} + CH^i \end{cases} \tag{9}$$

where, HS^i, CS^i is the unit's hot/cold start up cost and CH^i is the cold start hour.

Fuel cost functions $F_i\left(P_i^t\right)$ is frequently represented by the following polynomial function:

$$F_i\left(P_i^t\right) = a_i + b_i P_i^t + c_i \left(P_i^t\right)^2 \tag{10}$$

where a_i, b_i, c_i are the coefficients for the quadratic cost curve of generating unit i.

5 Evolutionary Optimized Local Support Vector Regression (EOLSVR)

5.1 Support Vector Regression (SVR)

The basic idea of SVR is to map the data x into a high dimensional feature space via a nonlinear mapping, and perform a linear regression in that feature space [17] as:

$$f(x) = \langle w, x \rangle + b \tag{11}$$

Where ⟨., .⟩ denotes the dot product, w contains the coefficients that have to be estimated from the data and b is a real constant. Using Vapink's ε–insensitive loss function [16], the overall optimization is formulated as:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2} w^T w + C\sum_{i=1}^{N}\left(\xi_i + \xi_i^*\right)$$
$$\text{subject to} \begin{cases} y_i - \left(w^T \phi(x_i) + b\right) \le \varepsilon + \xi_i^* \\ \left(w^T \phi(x_i) + b\right) - y_i \le \varepsilon + \xi_i \\ \xi_i, \xi_i^* \ge 0, \qquad i = 1,\ldots, N \end{cases} \tag{12}$$

where, x_i is mapped to higher dimensional space by the function φ, ε is a real constant, ξ_i and ξ_i^* are slack variables subject to ε-insensitive zone and the constant C determines the trade-off between the flatness of f and training errors.

Introducing Lagrange multipliers α_i and α_i^* with $\alpha_i\, \alpha_i^*=0$ and α_i, $\alpha_i^*=0$ for $i=1,\ldots,N$, and according to the Karush-Kuhn-Tucker optimality conditions [17], the SVR training procedure amounts to solving the convex quadratic problem:

$$\min_{\alpha,\alpha^*} \frac{1}{2}\sum_{i,j=1}^{N}\left(\alpha_i - \alpha_i^*\right)\left(\alpha_j - \alpha_j^*\right)Q\left(x_i, x_j\right) + \varepsilon\sum_{i=1}^{N}\left(\alpha_i + \alpha_i^*\right) - \sum_{i=1}^{N} y_i\left(\alpha_i - \alpha_i^*\right)$$
$$\text{subject to} \begin{cases} 0 \le \alpha_i, \alpha_i^* \le C \\ \sum_{i=1}^{N}\left(\alpha_i - \alpha_i^*\right) = 0, i = 1,\ldots, N \end{cases} \tag{13}$$

The output is a unique global optimized result that has the form:

$$\hat{f}(x) = \sum_{i=1}^{N}\left(\alpha_i - \alpha_i^*\right)Q(x_i, x_j) + b \tag{14}$$

where, Q(xi,x)= φ(xi). φ (x). Using kernels, all necessary computations can be calculated directly in the input space, without computing the explicit map φ(x). Various kernel types exist such as linear, hyperbolic tangent, Gaussian, polynomial, etc. [17]. Here, we employ the commonly used Gaussian kernel which can be as defined as following:

$$Q(x_i, x) = \exp\left(-\frac{\|x_i - x\|}{2\sigma^2}\right) \tag{15}$$

5.2 Local predictors

Local prediction is concerned with predicting the future based only on a set of K nearest neighbors in the reconstructed embedded space without considering the historical instances which are distant and less relevant. Local prediction constructs the true function by subdivision of the function domain into many subsets (neighborhoods). Therefore, the dynamics of time series can be captured step by step locally in the phase space and the drawbacks of global methods can be overcome.

The local SVR (LSVR) and local RBF (LRBF) methods can be summarized as follows [19]:

First, reconstruct the time series as described in the previous section. For, each query vector q, the K nearest neighbors $\{z_q^1, z_q^2, \ldots, z_q^K\}$ among the training inputs is choosing using the Euclidian distance as the distance metric between the q and each z in the reconstructed time series. Using these Knearest neighbors, train the SVR (or RBF) to obtainsupport vectors and corresponding coefficients. Finally, the output of SVR (or RBF) can be computed.

5.3 EOLSVR

There are some key parameters for SVR, which are C, ε and σ in the Gaussian kernel function. The selection of these parameters is important to the generalization of the prediction. Inappropriate parameters in SVR lead to overfitting or under-fitting [35]. Therefore, these parameters must be chosen carefully. However choosing the optimal parameters is a very important step in SVR, there are no general guidelines available to select these parameters till now. The problem of optimal parameters selection is very complicated because the complexity of SVR (and hence its generalization performance) depends on all three parameters together. Thus, a separate selection of each parameter is not adequate to get an optimal regression model.

There are many trials to choose the SVR's parameters. Various authors have selected these parameters by experience [16], [36] but this method is not suitable for nonexpert users. The grid search optimization method has been proposed by Scholkopf and Smola [37] to get the optimal parameters of SVR. However, this method is time consuming. The cross validation method has been also used to select the SVR's parameters [36]. This method is very computationally intensive and data-intensive. Pai and Hong proposed a GASVR model [38] to optimize the SVR's parameters in which the parameters are encoded as a binary code. This method suffers from some problems. The first one is that encoding the parameters as binary code will lead to integer valued solutions and may suffer from the lack of accuracy [29]. In addition, if the length of the string is not long enough, it might be possible for the GA to get near to the region of the global optimum but never will arrive at it.

As evident from above, there is a lacking of the structural methods for confirming the selection of SVR's parameters efficiently. Therefore, a real value GA is proposed in this work to select the SVR's parameters of local SVR method which simultaneously optimizes all SVR's parameters from the training data. The steps for load and price forecasting based on the proposed method can be summarized as following:

- Step 1: Reconstruct the time series: Load the multivar ate time series dataset X = (x_1(t), x_2(t), ..., x_M(t)), (t =1, 2, ...,N). Using the correlation dimension method and the mutual information method, calculate the embedding dimension d and the time delay constant m for each time series data set. Then, reconstruct the multivariate time series using these values.
- Step 2: Form a training and validation data: The input dataset after reconstruction $\tilde{X}$ is divided into two parts, that is a training $\tilde{X}_{tr}$ dataset and validation $\tilde{X}_{va}$ dataset. The

size of the training dataset is N_{tr} while the size of the validation dataset is N_{va}.

- Step 3: For each query point xq, choosing the K nearest neighbors of this query point using the Euclidian distance between x_q and each point in X_{tr} (1 < K<<< $\widetilde{X}_{tr}$).
- Step 4: Representation and generation of initial population: In real value GA the real value parameters can be used directly to form the chromosome. This means that the chromosome representation in real value GA is straightforward. In this case, the three parameters C, ε and σ are directly coded to generate the chromosome $CH = \{C, \varepsilon, \sigma\}$. These chromosomes are all randomly initialized.
- Step 5: Evaluation: each chromosome is evaluated using the fitness function which measures the performance of the model. It is quite important for evolving systems to find a good fitness measurement. The fitness (F) of each chromosome evaluated using mean absolute percentage error (MAPE) defined as:

$$\text{MAPE} = \frac{1}{N_{va}} \sum_{i=1}^{N_{va}} \frac{|A_i - F_i|}{A_i} \times 100 \tag{16}$$

where A_i and F_i are the actual value and the forecasted value, respectively, N_{va} is the validation dataset size, and $_i$ denotes the test instance index.

- Step 6: Selection: A standard roulette wheel selection method is employed to select the fittest chromosomes from the current population.
- Step 7: Crossover: The operator of crossover can now be implemented to produce two offspring from two parents which are chosen using the roulette wheel selection method. In this work, the line arithmetical crossover is used [28].
- Step 8: Mutation: Similarly, the mutation operation can contribute effectively to the diversity of the population. In this work, the Gaussian mutation [28] is used.
- Step 9: Elitist strategy: The chromosome that has the worst fitness value in the current generation is replaced by the chromosome that has the best fitness value in the old generation
- Step 10: Check the stopping criterion: The modelling can be terminated when the stopping criterion is reached. In this work, we use a predetermined maximum number of generations as a termination condition. If the stopping criterion is not satisfied, the model has to be expanded, the steps 5 to 9 can be repeated until the stopping criterion is satisfied.
- Step 11: After the termination condition is satisfied, the chromosome which gives the best performance in the last generation is selected as the optimal values of SVR's parameters.
- Step 12: Train SVR: The K nearest neighbors of the query point and the optimized parameters are used to train the SVR algorithm.
- Step 13: Calculate the prediction value of the current query point using equation (14).
- Step 14: Then, the steps 3 to 13 can be repeated until the future values of different query points are all acquired.

6 Exprement Results

In this paper, the performance of the EOLSVR is tested and compared with local SVR and local RBF using hourly load price and temperature data in South Australia. The load data used includes hourly load and price for the period of 2003-2005 for the South Australia electricity market. While the hourly temperature for the same period is collected from Macquarie university web site.

6.1 Parameters

To implement a good model, there are some important parameters to choose. Choosing the proper values of d and m is a critical step in the algorithm. The correlation dimension method and the mutual information method are used to select d and m, respectively, and the optimal values of these parameters are shown in Table 1. Using the obtained values of d and m, the multivariate time series can be reconstructed as described in Section 2.

TABLE 1
Phae Reconstruction Parameters

Dataset	Load data		Temperature data		Price data	
	d	m	d	m	d	m
South Australia electricity market	4	3	4	2	5	3

Choosing K is very important step in order to establish the local prediction model. There are some methods used in literatures to find this parameter. In this paper K is calculated using a systematic method which is proposed by us in [20]:

$$K = \text{round}\left(\frac{\alpha}{N \times k_{\max} \times D_{\max}} \sum_{i=1}^{N} \sum_{k=1}^{k_{\max}} D_k(x_i)\right) \tag{17}$$

where, N is the number of training points, k_{max} is the maximum number of nearest neighbors, $D_k(xi)$ is the distance between each training point x and its nearest neighbors while D_{max} is the maximum distance, $\frac{1}{N \times k_{\max} \times D_{\max}} \sum_{i=1}^{N} \sum_{k=1}^{k_{\max}} D_k(x_i)$ is the average distance around the points which is inversely proportional to the local densities and α is a constant. The two constants k_{max} and α are very low sensitivity parameters. k_{max} can be chosen as a percentage of the number of training points (N) for efficiency while α can be chosen as a percentage. In this paper, k_{max} and α are always fixed for all test cases at 70% of N and 95, respectively.

6.2 Forcasting Accuracy Evaluation

For all performed experiments, we quantified the prediction performance with the Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE). They can be defined as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |A(i) - F(i)| \tag{18}$$

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \frac{|A(i) - F(i)|}{A(i)} \times 100$$

(19)

where, A and F are the actual and the forecasted loads, respectively, n is the testing dataset size, and i denotes the test instance index.

6.3 Results and Discussion

The performance of the evolutionary optimized local SVR (EOLSVR) is tested and compared with local SVR and local RBF using hourly load, price and temperature data in South Australia.

To make results comparable, the same experimental setup is used for the three predictors. That is the week of February 15-21, 2005 has been used as a testing week. The available hourly load and temperature data (for the period of 2003-2005) are used to forecast the load of testing week. Also, the available hourly price and temperature data (for the period of 2003-2005) are used to forecast the price of testing week.

First, we calculate the MAE and MAPE of each day during the testing week. Then the average MAE and MAPE values of each method for the testing week are calculated. The results are shown in Tables 2 and 3.

These results show that, the EOLSVR method outperforms local SVR and local RBF. Table 4 shows the MAE improvements of the EOLSVR method over local SVR and local RBF. While Table 5 shows the MAPE improvements of the EOLSVR method over local SVR and local RBF. These results show the superiority of the proposed method over other methods.

TABLE 2
LOAD FORECASTING RESULTS

	Local RBF	Local SVR	EOLSVR
MAE (GW)	0.0314	0.0213	0.0132
MAPE (%)	2.3	1.55	0.94

TABLE 3
Price Forecasting Results

	Local RBF	Local SVR	EOLSVR
MAE (GW)	0.0322	0.0220	0.0140
MAPE (%)	3.79	2.96	1.90

Figures 1-3 show the actual load and forecasted load values using local RBF, local SVR and EOLSVR, respectively for the testing week.

TABLE 4
IMPROVEMENT OF THE EOLSVR OVER OTHER APPROACHES REGARDING MAE

	Load Forecasting		Price Forecasting	
	MAE	Improvement	MAE	Improvement
EOLSVR	0.0132	--	0.0140	--
Local RBF	0.0314	57.96%	0.0322	56.52%
Local SVR	0.0213	38.02%	0.0220	36.36%

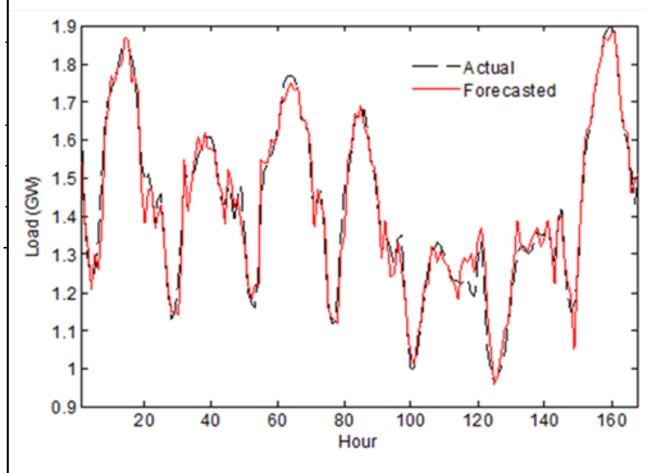


Fig. 1 Forecasted and actual hourly load from 15th to 21st February 2005 using local RBF

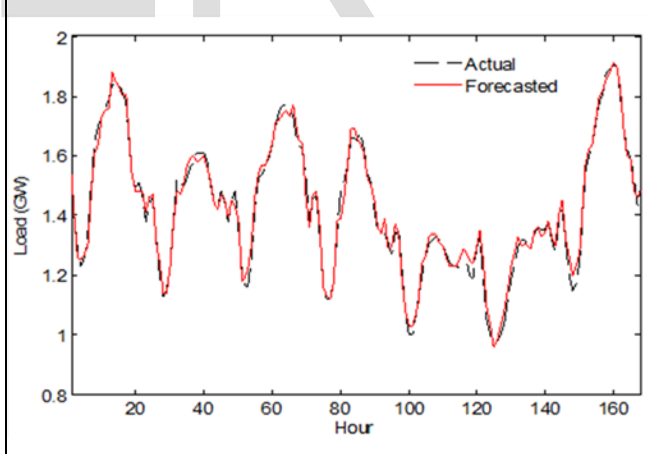
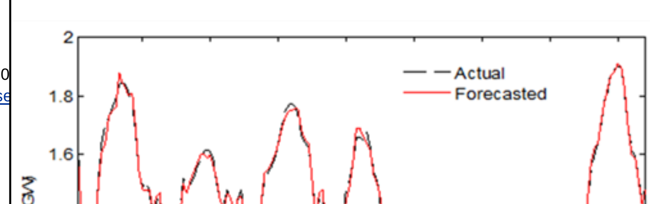


Fig. 2 Forecasted and actual hourly load from 15th to 21st February 2005 using local SVR



ISSN 2229-5518

The results of load and price forecasting are fed into the IEEE 30 bus test system. The IEEE 30 bus test system is used with a total of 6 generators and 41 lines. Table 6 shows the test system data. The spinning reserve is assumed to be 10% of the demand. The actual loads (24 hour) as well as the forecasted loads are given in Table 7.

If the initial commitment state of a generator is 1, it means this generator is on and zero indicates this generator is off. The IEEE 30 bus test system has 41 lines. Each line can transmit a maximum power flow in MW. Table 8 shows the maximum power flow for each line

Feasible unit combination and total cost (TC) values of the test system using dynamic programming method for load values and forecasting load values are given in Table 9. It is clear that accurate load forecasting is very important for the UC solution. The total cost of the forecasting load values for local RBF method is more than that of actual load values by $13140.6. Additionally, the total cost of the forecasting load values for local SVR is more than that of actual load values by $5016. Whereas the total cost of the forecasting load values for EOLSVR is more than that of actual load values by $3341.1.

7 Conclusion

In this paper, we have proposed EOLSVR method for electrical load and price forecasting. After that the results of load and price forecasting are used to solve the security constraint unit commitment problem.

The proposed method combines a proven powerful regression algorithm which is SVR with a local prediction framework. For data preprocessing, the embedding dimension and the time delay constant for the input data are computed firstly, and then the continuous load and price data are used for the phase space reconstruction. In addition, the neighboring points are presented by Euclidian distance. According to these neighboring points, the local model is set up. The local predictors can overcome the drawbacks of the global predictors by involving more than one model to utilize the local information. Therefore, the accuracy of the local predictor is better than the global predictor in which only one model is engaged for all available data that contains irrelevant patterns to the current prediction point. In addition, to set the SVR's parameters appropriately, a new method is proposed. This method adopts real value GA to seek the optimal SVR's parameters values and improves the prediction accuracy. Then the forecasted loads and price are fed into IEEE 30 bus test system for security constraint unit commitment to show the reactions of unit commitment to load and price forecasting errors.

Dynamic programming method is used for solving the UC problem. Total costs are calculated for load data which is taken from South Australia electricity market and forecasting load and price data computed by local RBF, local SVR and EOLSVR, separately. Comparing these total costs show that accurate load forecasting is important for UC. Over-prediction of STLF wastes resources since more reserves are available than needed and, in turn, increases the operating cost. On the other hand, under-prediction of STLF leads to a failure to provide the necessary reserves which is also related to high operating cost due to the use of expensive peaking units.

Acknowledgment

The authors gratefully acknowledge the Taif University for its support to carryout this work. It funded this project with a fund number 3267-435-1.

References

[1] A. D. Papalekopulos and T. C. Hesterberg, "A regression-based approach to short-term system load forecasting," IEEE Trans. Power Syst., 5 (4), 1535–1547, Nov. 1990.

[2] J. W. Taylor, L. M. de Menezes, and P. E. McSharry, "A comparison of univariate methods for forecasting electricity demand up to a day ahead," Int. J. Forecasting, 22 (1), 1–16, 2006.

[3] J. H. Park, Y. M. Park, and K. Y. Lee, "Composite modeling for adaptive short-term load forecasting," IEEE Trans. Power Syst., 6 (2), 450–457, May 1991.

[4] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short term load forecasting: A review and evaluation," IEEE Trans. Power Syst., 16 (1), pp. 44–55, Feb. 2001.

[5] C. J. Lin, C. H. Chen, and C. T. Lin, "A hybrid of cooperative particle swarm optimization and cultural algorithm for neural fuzzy networks and its prediction applications," IEEE Trans. Syst.,Man, Cyber. C, Appl. Rev., 39 (1), 55–68, Jan. 2009.

[6] S. Sheng and C. Wang, "Integrating radial basis function neural network with fuzzy control for load forecasting in power system," in Proc. IEEE Trans. Dist. Conf. Exhib. Asia and Pacific, 2005, pp. 1–5.

[7] C. Martos, J. Rodriguez, and M. Sanchez, 'Forecasting electricity prices by extracting dynamic common factors: application to the Iberian Market', IET Gener. Transm. Distrib., 6, (1), 11–20, 2012.

[8] L. Saini, S. Aggarwal, and A. Kumar, 'Parameter optimisation using genetic algorithm for support vector machine-based price forecasting model in National electricity market', IET Gener. Transm. Distrib., 4, (1), 36–49, 2010.

[9] T. Jakasa, I. Androces,and P. Sprcic, 'Electricity price forecasting –ARIMA model approach'. Eighth Int. Conf. Eur. Energy Market (EEM), 2011, pp. 222–225

[10] A. Conejo, M. Plazas, R. Espnola, and A. Molina, 'Day-ahead electricity price forecasting using the wavelet transform and ARIMA models', IEEE Trans.

Power Syst., 20 (2), 1035–1042, 2005.

[11] C. Garcia-Martos, C., J. Rodriguez and M. Sanchez, 'Mixed models for short-run forecasting of electricity prices: application for the Spanish market', IEEE Trans. Power Syst., 22,(2), 544–552, 2007.

[12] N. Amjady, A. Daraeepour, and F. Keynia, 'Day-ahead electricity price forecasting by modified relief algorithm and hybrid neural network', IET Gener. Transm. Distrib., 4 (3), 432–444, 2010.

[13] L. Coelhoa, and A. Santosb, "A RBF neural network model with GARCH errors: application to electricity price forecasting', Electr. Power Syst. Res., 81, 74–83, 2011.

[14] F. Lira, C. Munoz, F. Nunez, and A. Cipriano, 'Short-term forecasting of electricity prices in the Colombian electricity market', IET Gener. Transm. Distrib., 3 (11), 980–986, 2009.

[15] N. Amjady, and H. Hemmati, 'Day-ahead price forecasting of electricity markets by a hybrid intelligent system', Eur. Trans. Electr. Power, 19 (1), 89–102, 2009.

[16] V. N. Vapnik, Statistical Learning Theory. New York: Wiley, 1998.

[17] A. J. Smola and B. Scholkopf, "A tutorial on support vector regression," Royal Holloway College, Univ., London, U.K., NeuroCOLT Tech. Rep. NC-TR-98-030, 1998.

[18] M. Zhang, "Short-term load forecasting based on support vector machines regression," in Proceedings of the Fourth Int. Conf. on Machine Learning and Cybernetics, China, 18–21 August 2005.

[19] E. E. El-Attar, J. Y. Goulermas, and Q. H. Wu, "Forecasting electric daily peak load based on local prediction," in IEEE Power Eng. Soc. General Meeting (PESGM09), Canada, Jul. 26–30, 2009, pp. 1–6.

[20] E. E. Elattar, J. Y. Goulermas, and Q. H. Wu, "Electric load forecasting based on locally weighted support vector regression," IEEE Trans. Syst., Man and Cyber. C, Appl. and Rev., vol. 40, no. 4, pp. 438–447, Jul. 2010.

[21] E. E. Elattar, J. Y. Goulermas, and Q. H. Wu, "Integrating KPCA and locally weighted support vector regression for short-term load forecasting," in Proc. 15th IEEE Miditerranean Electrotechnical Conf. Valletta, Malta, Apr. 25–28, 2010, pp. 1528–1533.

[22] F. Takens, "Detecting Strange Attractors in Turbulence," Lect. Notes in Mathematics (Springer Berlin), vol. 898, pp. 366–381, 1981.

[23] D. Tao and X. Hongfei, "Chaotic Time Series Prediction Based on Radial Basis Function Network," Proc. Eighth ACIS Int. Conf. on Software Engin., Artificial Intelligence, Networking, and Parallel/Distributed Computing, pp. 595–599, 2007.

[24] T. Saksornchai, W. Lee, K. Methaprayoon, J. Liao, and J. Richard, "Improve the unit commitment scheduling by using the neural- network-based short-term load forecasting", IEEE Trans. on Industry applications, vol. 41, no. 1, pp. 169–179, Jan. 2005.

[25] T. Sauer, J. A. Yorke, and M. Casdagli, "Embedology," Journal of Statistical Physics, vol. 65, pp. 579–616, May 1991.

[26] P. Grassberger and I. Procaccia, "Estimation of the kolmogorov entropy from a chaotic signal," Physics Review A, vol. 28, pp. 2591–2593, 1983.

[27] W. Liebert and H. G. Schuster, "Proper choice of the time delay for the analysis of chaotic time series," Physics Letter, vol. 142, pp. 107–111, December 1989.

[28] Z. Michalewicz, Genetic algorithms + data structures = Evolving programs. Springer-Verlag, Berlin Heidelberg, 1996.

[29] D. E. Goldberg, Genetic algorithm in search, optimization and machine learning. Addison Wesley, 1989.

[30] J. E. Baker, "Reducing bias and inefficiency in the selection algorithm," in Proceedings of Int. Conf. Genetic Algorithms, 1987, pp. 14–21.

[31] A. P. Engelbrecht, Computational intelligence: An introduction. Jhon Willey and Sons, Ltd., 2007.

[32] M. Shahidepour, H. Yamin, and Z. Li., Market operations in electric power systems. John Wiley & Sons, Inc., 2000

[33] J. Zhu, Optimization of power system operation. John Wiley & Sons, Inc., 2009.

[34] F. Benhamida, E. N. Abdallah and A. H. Rashed, "Thermal unit commitment solution using an improved Lagrangian relaxation", International Conference on Renewable Energies and Power Quality (ICREPQ), Sevilla, Spain, 2007.

[35] K. Y. Chen and C. H. Wang, "Support vector regression with genetic algorithms in forecasting tourism demand," Tourism Management, vol. 28, pp. 215–226, 2007.

[36] V. Cherkassky and F. Mulier, Learning from data: Concepts, theory and methods. New York: Wiley, 1998.

[37] B. Scholkopf and A. J. Smola, Learning with kernels. Cambridge: MIT press, 2002.

[38] P. F. Pai and W. C. Hong, "Forecasting regional electric load based on recurrent support vector machines with genetic algorithms," Elec. Power Syst. Research, vol. 74, no. 3, pp. 417–425, 2005.

TABLE 6
TEST SYSTEM DATA

	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5	Unit 6
Pmax (MW)	500	400	250	250	200	350
Pmin (MW)	200	50	50	50	25	50
a ($/h)	10	10	20	10	20	10
b ($/MWh)	200	150	180	100	180	150
c ($/MWh2)	100	120	40	60	40	100
T_i^{up} (h)	5	4	3	3	1	4
T_i^{down} (h)	3	2	2	2	1	2
Start up cost	200	100	80	80	30	95
Initial state	1	0	1	1	0	0

TABLE 7
ACTUAL LOAD OF 6 UNITS 24 HOUR TEST SYSTEM AND THE FORECASTED LOADS USING LRBF, LSVR AND EOLSVR

Hour		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Loads (GW)	Actual	1.33	1.19	1.05	1.00	0.96	0.98	1.00	1.04	1.12	1.19	1.24	1.30	1.32	1.32	1.30	1.31	1.34	1.36	1.35	1.35	1.38	1.32	1.28	1.38
	LOCAL RBF	1.37	1.28	1.14	1.05	0.96	0.98	1.04	1.10	1.12	1.23	1.28	1.39	1.32	1.31	1.31	1.34	1.36	1.37	1.32	1.34	1.39	1.33	1.22	1.40
	LOCAL SVR	1.35	1.24	1.09	1.03	0.96	0.98	1.03	1.08	1.15	1.23	1.28	1.33	1.30	1.31	1.30	1.29	1.35	1.36	1.33	1.34	1.38	1.33	1.30	1.40
	EOLSVR	1.34	1.20	1.07	1.00	0.96	0.98	1.03	1.06	1.15	1.20	1.28	1.33	1.30	1.31	1.30	1.30	1.34	1.36	1.35	1.34	1.38	1.32	1.27	1.39

TABLE 8
MAXIMUM POWER FLOW FOR EACH LINE IN THE TEST SYSTEM (MW)

L1	650	**L11**	325	**L21**	80	L31	80	**L41**	160
L2	650	**L12**	160	**L22**	80	L32	80		
L3	325	**L13**	325	**L23**	80	L33	80		
L4	650	**L14**	325	**L24**	80	L34	80		
L5	650	**L15**	325	**L25**	80	L35	80		
L6	325	**L16**	325	**L26**	160	L36	325		
L7	450	**L17**	160	**L27**	160	L37	80		
L8	350	**L18**	160	**L28**	160	L38	80		
L9	650	**L18**	160	**L29**	160	L39	80		
L10	160	**L20**	80	**L30**	80	L40	160		

ISSN 2229-5518

TABLE 9

FEASIBLE UNIT COMBINATION OF TEST SYSTEM FOR ACTUAL LOAD AND FORECASTING LOAD VALUES USING LOCAL RBF, LOCAL SVR AND EOLSVR

Hour	Feasible UC (Actual load)	Feasible UC (Local RBF)	Feasible UC (Local SVR)	Feasible UC (EOLSVR)
1	1 1 1 1 1 0 1 0 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 0 0 1 1 0	1 1 1 1 1 1 1 0 0 0
2	1 1 1 1 1 0 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 0 0 0 0 0	1 1 1 1 1 0 1 0 0 0
3	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0
4	1 1 0 0 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 0 0 1 0 0 0 0 0
5	1 1 0 0 1 0 0 0 0 0	1 1 0 0 1 0 0 0 0 0	1 1 0 0 1 0 0 0 0 0	1 1 0 0 1 0 0 0 0 0
6	1 1 0 0 1 0 0 0 0 0	1 1 0 0 1 0 0 0 0 0	1 1 0 0 1 0 0 0 0 0	1 1 0 0 1 0 0 0 0 0
7	1 1 0 0 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0
8	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0
9	1 1 0 1 1 0 0 0 0 0	1 1 0 1 1 0 0 0 0 0	1 1 1 1 1 0 0 0 0 0	1 1 1 1 1 0 0 0 0 0
10	1 1 1 1 1 0 0 0 0 0	1 1 1 1 1 0 0 0 0 0	1 1 1 1 1 0 0 0 0 0	1 1 1 1 1 0 0 0 0 0
11	1 1 1 1 1 0 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0
12	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0
13	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0
14	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0
15	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0
16	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0
17	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0
18	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 0 0 0
19	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 1 0 0
20	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0
21	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 1 0 0
22	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 1 0 0
23	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 0 0 0	1 1 1 1 1 1 0 1 0 0
24	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 1 0 0 0	1 1 1 1 1 1 0 1 0 0	1 1 1 1 1 1 0 1 0 0
TC	**$ 606165.3**	**$ 619305.9**	**$ 611181.5**	**$ 609506.4**