Miloš STOJANOVIĆ[1], Miloš BOŽIĆ[2], Zoran STAJIĆ[1], Marko MILOŠEVIĆ[3]

School of Higher Technical Professional Education Niš (1), Faculty of Electronic Engineering, University of Niš (2), R&D Center Alfatec (3)

LS-SVM model for electrical load prediction based on incremental training set update

***Abstract**. In this paper a forecasting model based on an incremental update scheme is proposed for the hourly load demand of the next day, using least square support vector machines (LS-SVM). The model is based on historical daily load demands in combination with calendar and climate features. The presented model was tested on real-life load data and the results show that the proposed approach can, by catching the evolving nature of the load pattern and dynamically updating the training set with new instances, lead to significant improvements in the accuracy of load forecasts.*

***Streszczenie.** W artykule opisano opracowany model przyrostowy do przewidywania godzinowego zapotrzebowania na energię elektryczną na dzień następny, w którym wykorzystano maszynę wektorów pomocniczych LS-SVM. Proponowany model bazuje na wcześniejszych danych, dotyczących zapotrzebowania dziennego w połączeniu z analizą kalendarza i warunków klimatycznych. Badania eksperymentalne na rzeczywistych danych pozwala na skuteczne przewidywanie obciążenia energetycznego. (**Model LS-SVM w predykcji obciążenia elektrycznego – przyrostowa metoda uczenia**).*

Keywords: load forecasting, least squares support vector machines, incremental model.
Słowa kluczowe: przewidywanie obciążenia, LS-SVM, model przyrostowy.

Introduction

Nowadays, with the privatization and deregulation of electricity networks, accurate electric load forecasting has an even more important role in the planning, operation and control of electric power systems. Usually, the short-term load forecast (STLF) is related to the hourly prediction of electricity load demand for a time period from one hour to a few days in advance. Many operating decisions rely on accurate STLF, such as generation capacity scheduling, scheduling of fuel and coal purchases, system security analysis, energy transaction planning, etc. It also plays a significant role in the coordination of hydro-thermal systems, generator maintenance scheduling, load flow analysis, etc. Therefore, improving STLF accuracy is crucial for increasing the efficiency of energy systems and reducing operational costs.

However, STLF is a complex problem because of load nonlinear relationships with other factors such as weather conditions, social activities, seasonal factors, past usage patterns and calendar features. Each of these factors has a significant impact on future load.

In recent decades, many STLF methods have been developed. These methods can generally be classified in one of three categories: conventional methods, artificial intelligence techniques and hybrid methods. The most frequently used conventional techniques are linear regression methods [1], exponential smoothing [2], the Box-Jenkins ARIMA approach [3] and the Kalman filter [4]. Artificial intelligence based techniques include neural network models [5], expert system models [6], fuzzy inference [7], support vector machines [8], least squares support vector machines [9] and relevance vector machines [10]. Hybrid models have been presented for STLF in [11-15].

LS-SVMs, proposed in [16], as reformulations of standard SVMs, instead of solving the quadric programming (QP) problem, which is complex to compute, obtain a solution from a set of linear equations. Therefore, LS-SVMs have a significantly shorter computing time and they are easier to optimize.

In this paper a forecasting model based on the incremental update of a training set with new instances is proposed for the hourly load demand of the next day, using LS-SVMs. Most machine learning based models, employed for the STLF, use a fixed size training set, i.e. forecasts for several days or even weeks are being made by a training model with the same training set. But in many forecasting problems, such as STLF, where new data is constantly arriving, a dynamic update of the model is crucial for improving and preserving its performance. Accordingly, after the hourly forecasting of the load for the next day has been completed, the initial training set is updated by adding hourly data from the previous day, which is in that moment known. Then, for next day prediction, the model is retrained with an extended training set. As the experimental results show, in this way improvements to the accuracy of forecasting results can be achieved.

The rest of this paper is organized as follows: Section 2 presents the used methodology. Section 3 presents the data set analysis, section 4 describes the proposed forecasting model. Section 5 gives the experimental results. Finally, Section 6 provides conclusions.

Least squares support vector machines

Consider a known training set $\{\mathbf{x}_k, y_k\}$, $k=1,...,N$ with input vectors $\mathbf{x}_k \in R^n$ and output scalars $y_k \in R$. The following regression model is built by using the non-linear mapping function $\varphi(\cdot): R^n \rightarrow R^{n_h}$ which maps the input space into a high-dimensional feature space and constructs a linear regression in it, expressed in:

$$y(\mathbf{x}) = \boldsymbol{\omega}^T \varphi(\mathbf{x}) + b, \tag{1}$$

where $\boldsymbol{\omega}$ represents the weight vector and b is a bias term. The optimization problem is formulated in primal space:

$$\min_{\boldsymbol{\omega},b,\mathbf{e}} J_p(\boldsymbol{\omega},\mathbf{e}) = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + \frac{1}{2}\gamma\sum_{k=1}^{N} e_k^2 , \tag{2}$$

subject to linear equality constrains expressed by:

$$y_k = \boldsymbol{\omega}^T \varphi(\mathbf{x}_k) + b + e_k, k = 1,...,N , \tag{3}$$

where e_k represents error variables and γ is a regularization parameter which must be determined by the user.

In order to solve the optimization problem defined with (2) and (3), it is necessary to construct a dual problem using the Lagrange function. The solution to this problem is presented as:

$$\begin{bmatrix} 0 & \mathbf{1}_v^T \\ \mathbf{1}_v & \boldsymbol{\Omega} + \mathbf{I}\gamma^{-1} \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix} . \tag{4}$$

Example (4), $\mathbf{y}=[y_1,...,y_N]^T$, $\mathbf{1_v}=[1,...,1]^T$ and $\boldsymbol{\alpha}=[\alpha_1,...,\alpha_N]^T$ represents the column vectors of dimensions $N\ x\ 1$, where α_k, $k=1,...,N$ are Lagrange multipliers, $\mathbf{I}$ is the identity matrix and $\mathbf{\Omega}_{kl}=\varphi(\mathbf{x}_k)^T\varphi(\mathbf{x}_l)=K(\mathbf{x}_k,\mathbf{x}_l), k,l=1,...,N$ denotes the kernel matrix, both of them of dimensions $N\ x\ N$. The linear system defined in (4) is of the order $(N+1)\ x\ (N+1)$. It is important to notice that in (4) vector $\mathbf{y}$ is formed form the training set outputs y_k, $k=1,...,N$.

The resulting LS-SVM model for function estimation in dual form is represented in (5), where α_k and b are solutions of linear system defined by (4):

$$(5)\qquad y(\mathbf{x})=\sum_{k=1}^{N}\alpha_k K(\mathbf{x},\mathbf{x}_k)+b\ .$$

The dot product $K(\mathbf{x},\mathbf{x}_k)=\varphi(\mathbf{x})^T\varphi(\mathbf{x}_k)$ represents a kernel function. Kernel functions enable the computation of the dot product in a high-dimensional feature space by using data inputs from the original space, without explicitly computing $\varphi(\mathbf{x})$. A commonly used kernel function in non-linear regression problems, one that is employed in this study, is a radial basis function represented as:

$$(6)\qquad K(\mathbf{x},\mathbf{x}_k)=e^{-\frac{\|\mathbf{x}-\mathbf{x}_k\|^2}{\sigma^2}}\ ,$$

where σ represents a kernel parameter which should be determined by the user. When choosing an RBF kernel function with LS-SVM, the optimal parameter combination (γ, σ) should be established. It can be noticed that only two additional parameters (γ, σ) need to be optimized, instead of three (γ, σ, ε) as in SVM but at the cost of a lack of a sparse solution, which follows from (5).

Parameter selection is the most important part in the formation of the LS-SVM model, because it significantly affects performance. Accordingly, a grid search algorithm in combination with a *k*-fold cross-validation is employed in this paper.

Data analysis

The available data used for model formation consist of weather, calendar and history load demands for the territory of the city of Niš and its surroundings for the period from January 2008 to March 2010. The data were obtained from electric distribution utility "ED Jugoistok".

The average daily temperature is the most important weather variable that affects the electrical load pattern. Fig. 1 shows the relation between the load and average daily temperature for the period of one winter month. The temperature and load have a negative correlation coefficient for the winter months which is close to one. This is explained by that the load demand increase with a temperature decrease and vice versa. Temperature shapes the trend of the load demand curve. Temperature also influences load variance, but that influence is smaller than influence on the trend.

The hour of the day and day of the week are two major calendar variables that have an influence on the load curve shape. Load changes during the day from one hour to another. Fig. 2 shows hourly load during the day for each day of the week in February.

From Fig. 2 it can be seen that load curve is slightly different from one week day to another but its shape is the same. It can be noticed that load demand during the weekends is less than on the week days, and not only for the winter season but also for all seasons.

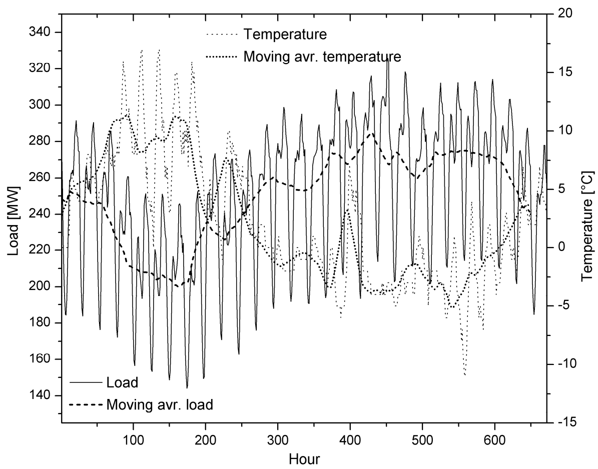

Fig. 1. Average daily temperature and load for one winter month

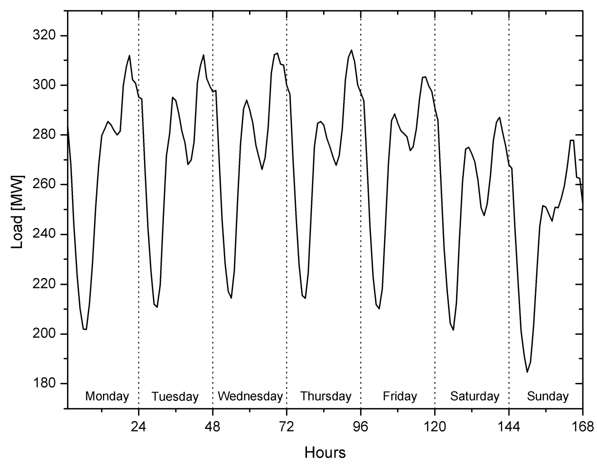

Fig. 2. Hourly load during the week

Fig. 3 shows that the shape of the load curves for each day of the week are very similar, which indicates that the usage of history load per hour can supply the model with additional information about the expected load. Load peak occurs twice a day, one is from about 9:00 AM to 10:00 AM and another is from about 7:00 PM to 9:00 PM. When history load is used as a model feature, the question is which size of regressor to use, i.e. how many past load values to take as model features.

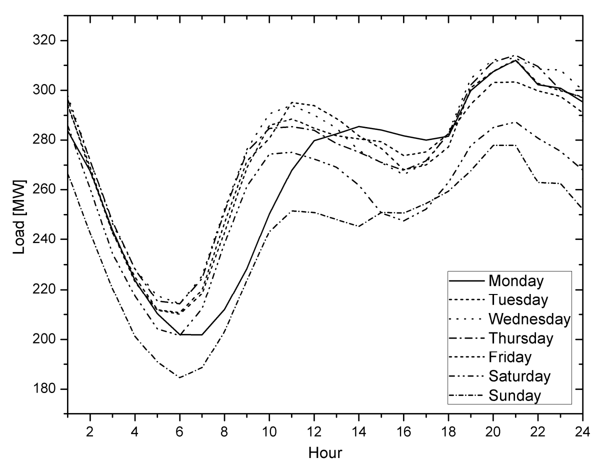

Fig. 3. Hourly load during the day

Model formation

Different factors that have an influence on electric load were analyzed, and accordingly appropriate features were chosen for the model. The past load time horizon used in this paper is m=24, i.e. the model used the last 24 hour loads from the prediction moment. Fig. 4 shows the structure of the input vectors. Input vectors consist in total of $m+s$ features, where m is the past load time-series features P_k, $k=1,..,24$ and s=3 non-time series features: the average daily temperature T_k, the hour of the day H_k, $H_k \in \{1, 2, ..., 24\}$ and the day of the week D_k, $D_k \in \{1, 2, ..., 7\}$ where 1 corresponds to Monday, 2 to Tuesday and so on. It is important to emphasize that D_k and H_k are known values in both the training and test set, but the temperature T_k is only known in the training set, while in the test set the predicted temperature for the next day is used instead of the true one which is unavailable at the moment of prediction.

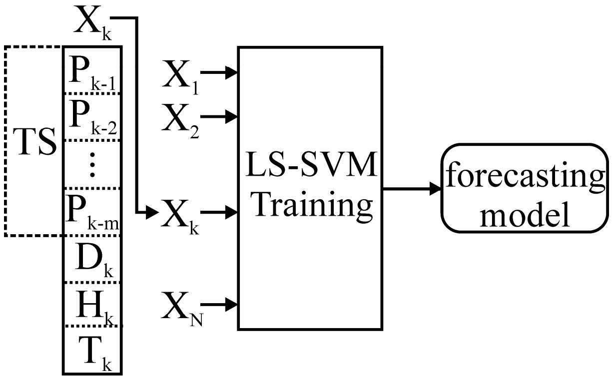

Fig. 4. Input vector structure

In order to have an optimal training of the model, the data set has to be normalized before training. This prevents the dominance of any features in the output value and provides faster convergence and better accuracy of the learning process. Accordingly, all features and their output values are normalized within the range [0, 1].

For the initial training segment choice the "similarity" principle is used, modeled based on the k-nearest-neighbor (KNN) algorithm. The idea behind the algorithm is that similar inputs also have similar mapping relationships with their outputs. The training segment is chosen to have a similar calendar category as the forecasting period (Feb 2010), in order to compose an LS-SVM training set with similar weather features. Therefore, the training segment consists of hourly features for three winter months (Dec. 2008, Jan. 2009, Feb. 2009). This also reduces the size of the training set and the constructing time of the LS-SVM.

The model training procedure and the hourly forecasting of the load demand for one day in advance is presented in Algorithm 1, where $\{\mathbf{X}, \mathbf{Y}\} = \{\mathbf{x}_k, y_k\}, k = 1, ..., N$ denotes the initial training set with input vectors $\mathbf{x}_k \in R^{m+s}$ and outputs $y_k \in R$.

Algorithm 1. LS-SVM model training and forecasting

```
for j = 1 ... number of days
    xt= Xt(j)
    (γ,σ)=tunesvm(X, Y, grid-search, cross-validation)
    model = trainsvm(X, Y, γ , σ)
    for i = 1 ... 24
        P(i) = forecast(model, xt)
        Update xt:
        xt(1) = hi+1
        xt = shift_ left_time-series(xt, 1)
        xt(m+s)=P(i)
    endfor
    Results(j)=P
    Update(X, Y)
endfor
Output : Results
```

The first step in the algorithm is the selection of a new instance $\mathbf{x}_t$ from the test set $\mathbf{X}_t$. After that, the optimal (γ, σ) pair is determined on $(\mathbf{X}, \mathbf{Y})$ using a grid search with k-fold cross validations, as mentioned in section 2. The training set is randomly subdivided into k disjoint subsets of approximately equal size and the local LS-SVM model is built k times with the current pair (γ, σ). Each time, one of the k subsets is used as the test set and the other k - 1 subsets are put together to form a training set. After k iterations, the average model error is calculated for the current pair (γ, σ). The entire process is repeated with an update of the parameters (γ, σ) until the given stopping criterion (e.g. Mean Squared Error) is reached. The parameters (γ, σ) are updated exponentially in the given range using predefined equidistant steps, according to the grid-search procedure. After obtaining the optimal (γ, σ) combination, an LS-SVM forecasting model is formed according to (5) and (6).

The model is then employed for the prediction of load demand for one step ahead, i.e. for the next hour, and the result is placed in vector P. After that, it is necessary to update the $\mathbf{x}_t$ vector for the next prediction step. The update is needed because the true values of the load for the past 24 hours are available only for the first prediction step. After that, for the next predictions, the predicted values from the previous steps are used instead of the true ones, which are unknown at the moment. Accordingly, the hour feature h_{i+1} is updated (day and temperature features remain the same for the current day), load time-series are shifted left for one place in order to remove load from the earliest hour, and the prediction from the previous step is placed in the final position of the time-series instead of the true load from the last hour. The whole process is repeated 24 times and at the end, hourly predictions for the following day will be obtained.

Before the selection of the next $\mathbf{x}_t$ for the next day, the initial training set is updated by adding hourly data from the previous day, which is known at that moment, and the model is retrained. The update and re-training is performed in each iteration of the outer loop (by day) until the given number of days in the test is reached. Although it is possible to update the initial training set and re-train the model in each iteration of the inner loop (by hour), in this case it is not necessary, because hourly predictions are needed for one day ahead, i.e. the prediction horizon is one day. If prediction for one hour ahead was needed, the model could be retrained in each iteration of the inner loop (by hour).

Experimental results

For methodology evaluation, the forecasting of hourly loads in February 2010 was done for each day. Two models are generated, the first with the initial training set configuration and the second with an incremental training set update scheme. These models are denoted with:

- M1 - a model trained with an initial training set that contains 2136 vectors,
- M2 - a model which is retrained with daily update of the training set.

To be clear, for the prediction of the first day in the test set (i.e. February 1st, 2010) the M2 model is trained with a training set that contains 2136 vectors, same as model M1. For the prediction of February 2nd, 24 vectors from the previous day (which are known at the moment) are added and the model is then retrained with 2160 vectors. And finally, for the prediction of February 28th all of the vectors for the previous 27 days are added into the initial training set, and the model is retrained with 2784 vectors.

The prediction quality is evaluated using Mean Absolute Percentage Error (MAPE), Maximum Error (ME) and Absolute Percent Error (APE) defined with:

$$MAPE[\%] = 100\frac{1}{n}\sum_{i=1}^{n}\left|\frac{P_i - \hat{P}_i}{P_i}\right| , \tag{7}$$

$$ME = \max_i \left|P_i - \hat{P}_i\right| , \tag{8}$$

$$APE[\%] = \left|\frac{P_i - \hat{P}_i}{P_i}\right| \cdot 100 , \tag{9}$$

where P_i and $\hat{P}_i$ are the real and the predicted value of the load demand in the i^{th} hour and $n=24$ is the number of hours.

The MAPEs obtained by committing models with the test set are shown in Fig. 5. The MEs obtained by committing models with the test set are shown in Fig. 6.

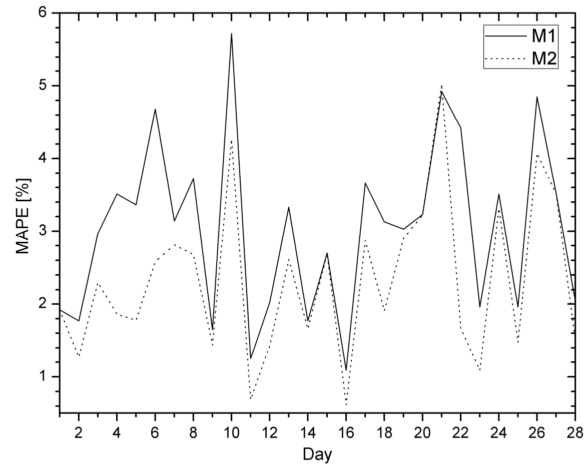

Fig. 5. Daily MAPEs for models M1 and M2

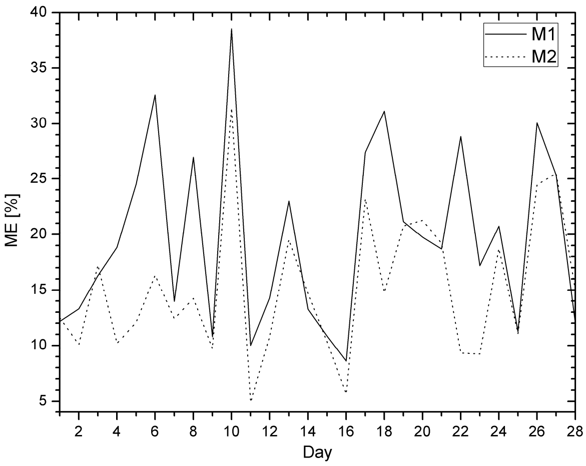

Fig. 6. Daily MEs for models M1 and M2

Table 1 shows the models average, maximum and minimum daily MAPEs for the entire test set. Similarly, Table 2 shows the average, maximum and minimum daily MEs. As can be noticed form Figs. 5 and 6 and Tables 1 and 2, by using the proposed methodology, model M2 performs better than model M1. All of the values for average, max and min MAPEs are significantly reduced, as well as the values for average, max and min MEs.

Table 1. Average, max and min daily MAPEs of the entire test set

Model/MAPE	Average	Max	Min
M1	3.02	5.72	1.09
M2	2.32	5.02	0.62

Table 2. Average, max and min daily MEs of the entire test set

Model/ME	Average	Max	Min
M1	19697	38536	8592
M2	15167	31421	4914

Figs. 7 and 8 show the hourly APEs for models M1 and M2 respectively, for all the days in the test period. From these Figs., how the APEs behave from hour to hour and from day to day can clearly be determined. The black areas correspond to the points with a high APE, e.g. with an APE of 16%. The opposite white areas correspond to points where the APE ranges from 0 to 2%. It is obvious that the black areas in Fig. 7 are reduced in Fig. 8 for the APE distribution of model M2.

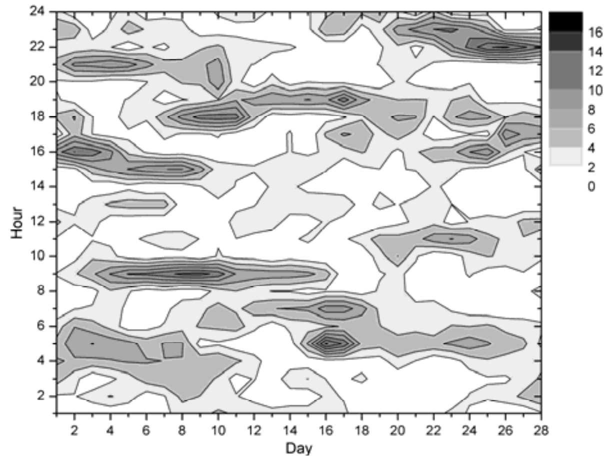

Fig. 7. Hourly APE distribution for the M1 model

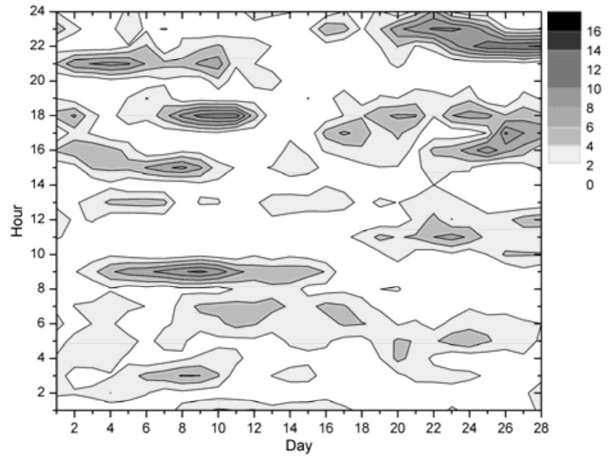

Fig. 8. Hourly APE distribution for the M2 model

In Fig. 9 real and predicted hourly load demands for February 18th with the mid-range MAPE value is given. From this figure it can be noticed that the predictions from model M2 have an improved shape and trend in comparison to model M1.

Conclusion

One method for improving short-term load forecasting is presented in this paper. The proposed approach is based on the incremental update of the initial training set by adding new instances into it as soon as they become available and then retraining the model. By this approach the evolving nature of the load pattern is followed and the model performance is preserved and improved, as the experimental results confirm, although the model trained only with an initial training set showed quite a good performance.

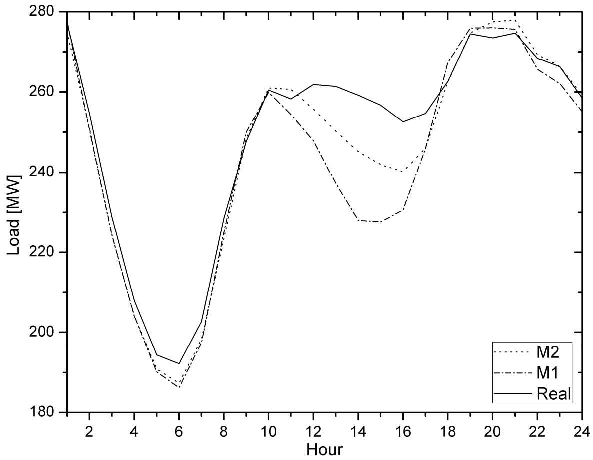

Fig. 9. Real and predicted loads of models M1 and M2 for February 18th

Different features that affect load demand are analyzed, and the appropriate ones were chosen for the structure of training vectors. The initial training set was carefully chosen to "match" with the predicted season. Model evaluation was done hourly for period of one winter month, which represents the large test segment, taking into account that the predictions are made by hours. The LS-SVMs were chosen for the non-linear model because of their good generalization performance and ability to avoid local minima.

Although the complexity of the calculations in the proposed algorithm is increased in regard to training only one forecasting model, it brings significant improvements to load forecasting accuracy.

Further work could consider the development of an algorithm for the determination of which vectors should be added into the initial training set, and possibly which need to be discarded.

Acknowledgements

This work was supported by the Ministry of Science and Technological Development, Republic of Serbia (Project number: III - 44006).

REFERENCES

[1] Papalexopoulos A. D., Hesterberg T. C., A regression-based approach to short-term load forecasting, *IEEE Transactions on Power System*, 5 (1990), No. 4 1535–1550.

[2] Christiaanse W. R., Short term load forecasting using general exponential smoothing, *IEEE Transactions on Power Apparatus and Systems*, 90 (1971), 900–911.

[3] Huang S. J., Shih K. R., Short-term load forecasting via ARMA model identification including non Gaussian process considerations, *IEEE Transactions on Power System*, 18 (2003), No. 2, 673–679.

[4] Irisarri G. D., Widergren S. E., Yehsakul P. D., On-line load forecasting for energy control center application, IEEE *Transactions on Power Apparatus and Systems*, 101 (1982), 71–78.

[5] Hippert H. S., Pedreira C. E., Castro R., Neural networks for short term load forecasting: A review and evaluation, *IEEE Transactions on Power System*, 16 (2001), No. 1, 44–55.

[6] Rahman S., Bhatnagar R., An expert system based algorithm for short term load forecast, *IEEE Transactions on Power System*, 3 (1998), No. 2, 392–399.

[7] Mori H., Kobayashi H., Optimal fuzzy inference for short-term load forecasting, *IEEE Transactions on Power System*, 11 (1996), 390–396.

[8] Chen B., Chang M., Lin C., Load forecasting using support vector machines: A study on EUNITE competition 2001, *IEEE Transactions on Power System*, 19 (2004), No. 4, 1821–1830.

[9] Božić M., Stojanović M., Stajić Z., Short - term electric load forecasting using Least squares support vector machines, Facta Universitatis Automatic Control and Robotics, 9 (2010), No. 1, 141-150.

[10] Wang L., Liu Z., RVM with wavelet kernel combined with PSO for short-term load forecasting in electric power systems, *Przegląd Elektrotechniczny*, 88 (2012), nr 1b, 146-149.

[11] Amjady N., Short-term bus load forecasting of power systems by a new hybrid method, *IEEE Transactions on Power System*, 22 (2007), No. 1, 333–341.

[12] Huang C. M., Huang C. J, Wang M. L., A particle swarm optimization to identifying the ARMAX model for short-term load forecasting, *IEEE Transactions on Power System*, 20 (2005), No. 2, 1126–1133.

[13] Asrari A., Javan D. S., Javidi M. H., Monfared M., Application of Gray-Fuzzy-Markov Chain Method for Day-Ahead Electric Load Forecasting, *Przegląd Elektrotechniczny*, 88 (2012), nr 3b, 228-237.

[14] Dudek G., Neuro-fuzzy approach to the next day load curve forecasting, *Przegląd Elektrotechniczny*, 87 (2011), nr 2, 61-64.

[15] Dudek G., Artificial Immune Clustering Algorithm to Forecasting Seasonal Time Series, Computational Collective Intelligence. Technologies and Applications, vol. 6922, Springer, pp 468 - 477, 2011.

[16] Suykens K., Gestel T., Brabanter J., Moor B, Vandewalle J., Least *Squares Support Vector Machines*, World Scientific, Singapore, 2002.

***Authors**: Miloš Stojanović, School of higher Technical Professional Education, Aleksandra Medvedeva 20, 18000 Niš, Serbia, E-mail: milosstojanovic10380@yahoo.com; Miloš Božić, Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia, E-mail: milos1bozic@gmail.com; Zoran Stajić, Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia, E-mail: zoran.stajic@elfak.ni.ac.rs; Marko Milošević, R&D Center Alfatec, Nikole Tesle 63/5, 18000 Niš, Serbia, E-mail: marko.milosevic@alfatec.rs.*