NEW ANN MODELS FOR SHORT TERM FORECASTING OF ELECTRICITY LOADS

Jelena Milojković[1], Vančo Litovski[2]

[1]University of Niš, Faculty of Electronic Engineering, Aleksandra Medvedeva 14, 1800 Niš, Serbia
[2]University of Niš, Faculty of Electronic Engineering, Aleksandra Medvedeva 14, 1800 Niš, Serbia

vanco@elfak.ni.ac.rs (Vančo Litovski)

Abstract

Short term prediction of electricity load is discussed. It will be shown here first that for the subject of short term prediction of electricity load, even though a large amount of data may be available, only the most recent of it may be of importance. That gives rise to prediction based on limited amount of data. We here propose implementation of feed-forward artificial neural network models for a potentially systematic solution of that problem as opposed to heuristics that are in use. Examples will be given related to short-term (hourly) one- and two-steps-ahead forecasting of the electricity load at suburban level.

Keywords – modelling, forecast, power consumption, artificial neural networks

Presenting Author's biography

Vančo B. Litovski was born i Rakita, South Macedonia, Greece. He received all his degrees from the Faculty of Electronic Engineering in Niš, Serbia. His research interest spans from electronic filters, over electronic circuits and systems modeling and simulation, to integrated circuits design, testing, and diagnosis. He is author of several hundreds of scientific papers and several tenths of scholar books. He received The Tesla and The Savastano awards. Prof. Litovski is the establisher and President of the Yugoslav Simulation Society. He is chairing the Organising and the Program Committees of the Small Systems Simulation Symposium.

1. Introduction

In an inspired paper [1] Prof. Mendel' clames: "Prediction of short time series is a topical problem. Cases where the sample length N is too small for generating statistically reliable variants of prediction are encountered every so often. This form is characteristic of many applied problems of prediction in marketing, politology, investment planning, and other fields." Further he claims: "Statistical analysis suggests that in order to take carefully into account all components the prediction base period should contain several hundreds of units. For periods of several tens of units, satisfactory predictions can be constructed only for the time series representable as the sum of the trend, seasonal, and random components. What is more, these models must have a very limited number of parameters. Series made up by the sum of the trend and the random component sometimes may be predicted for even a smaller base period. Finally, for a prediction base period smaller than some calculated value $N_{\min}$, a more or less satisfactory prediction on the basis of observations is impossible at all, and additional data are required".

Among the fields not mentioned in [1], dealing with really small set of data or "prediction base period", we will discuss here hourly short-term prediction of electricity loads at suburban level or on the level of a low voltage transformer station. In fact, as can be seen from Fig. 1. the amount of data available in this case is large enough to apply any other forecasting method [2,3,4] but looking to the load diagram i.e. hourly load-value curves, we easily recognize that past values of the consumption are not very helpful when prediction is considered. That stands even for data from the previous day and for data from the same day in the previous week.

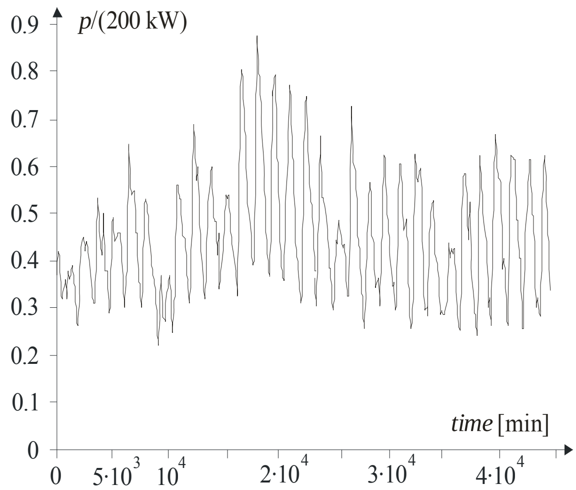

Figure 1. Average values of hourly consumption at the observed site in January 2009

As an illustration of the claim in Fig. 2 we give three load diagrams representing one day consumption of one load on a) Friday January 31, 2009, b) Thursday January 30, 2009, and c) Friday January 24, 2009. The numerical values are shown in Table 1. The power is normalized by a factor of 200 being the turn ratio of the appropriate current transformer in the transformer station. One may notice the similarity of the general shape and the difference in main details confirming the paramount importance of the most recent data for prediction.

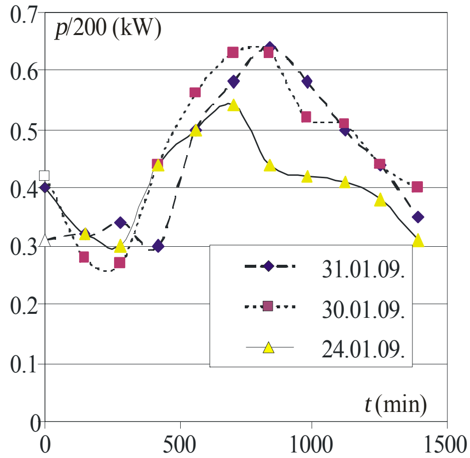

Figure 1. Average values of hourly consumption (kW) on three days (Table 1. visualized)

Accordingly, we propose the problem of prediction of the load value in the next hours (two or four) to be performed as a deterministic prediction based on very short – one day – time series. To help the prediction, however, in an appropriate way, we introduce past values e.g. loads for the same day but in previous weeks. That is in accordance with existing experience claiming that every day in the week has its own general consumption profile [2]. The fundamental idea is best understood using Fig. 3 where the dots represent load values of the same time in the past weeks while the bolded part of the line at the end of the curve covers the actual load values used for control of the forcast.

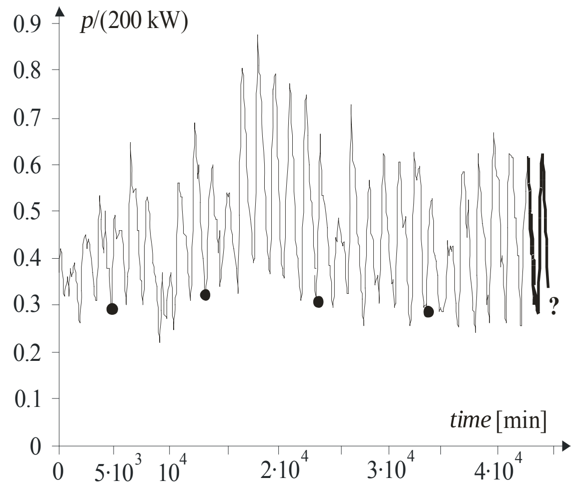

Figure 3. Data used for short term forecast

Having all that in mind we undertook a project of developing an artificial neural network (ANN) based model that will be convenient for systematic implementation in stationary time series prediction with reduced set of data. Our first results were applied to prediction of environmental as well as technological data and published in [5, 6] while general analysis as to why neural networks are implemented for prediction may be also found in [5]. The main idea exploited was the following. If one wants to create neural network that may be used for forecasting one should have such a topology and data structure to accommodate to the prediction problem observed. Following these considerations new forecasting structure was developed. Later on, in order to accommodate for implementation in the field of short term electricity load forecasting it was upgraded [7].

Table 1. Average values of hourly consumption (kW) on three days

No.	t (min)	p/200 31.01.09	p/200 30.01.09	p/200 24.01.09
1	0.06	0.40	0.42	0.31
2	146	0.32	0.28	0.32
3	284	0.34	0.27	0.30
4	422	0.30	0.44	0.44
5	561	0.50	0.56	0.50
6	700	0.58	0.63	0.54
7	837	0.64	0.63	0.44
8	977	0.58	0.52	0.42
9	1115	0.50	0.51	0.41
10	1255	0.44	0.44	0.38
11	1393	0.35	0.40	0.31

The goal of this paper is to describe our model`s implementation for short term electricity load forecasting and, for the first time to produce results of implementation to two step ahead prediction.

The structure of the paper is as follows. After general definitions and statement of the problem we will give a short background related to ANNs application to forecasting. Then we will describe the solution for possible applications of ANNs aimed to short term forecasting of electricity loads. Finally short discussion of the results and consideration related to future work will be given.

2. Problem formulation and one-step-ahead solutions

A time series is a number of observations that are taken consecutively in time. A time series that can be predicted precisely is called deterministic, while a time series that has future elements which can be partly determined using previous values, while the exact values cannot be predicted, is said to be stochastic. We are here addressing only deterministic type of time series.

Consider a scalar time series denoted by y_i, i=1,2, … m. It represents a set of observables of an unknown function $\hat{y} = \hat{f}(t)$, taken at equidistant time instants separated by the interval Δt i.e. $t_{i+1} = t_i + \Delta t$. One step ahead forecasting means to find such a function f that will perform the mapping

$$y_{m+1} = f(t_{m+1}) = \hat{y}_{m+1} + \varepsilon, \qquad (1)$$

where $\hat{y}_{m+1}$ is the desired response, with an acceptable error ε.

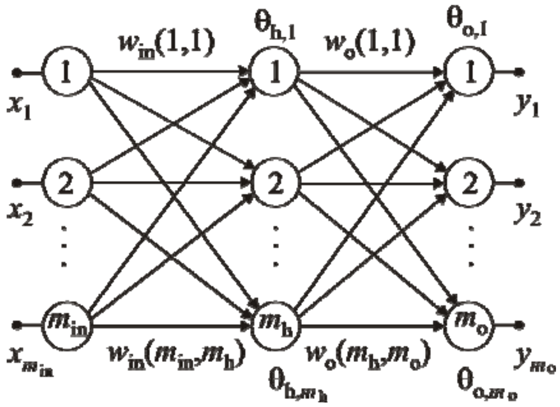

Figure 4. Fully connected feed-forward neural network with one hidden layer and multiple outputs

The prediction of a time series is synonymous with modeling of the underlying physical or social process responsible for its generation. This is the reason of the difficulty of the task. There have been many attempts to find solution to the problem. Among the classical deterministic methods we may mention the k-nearest-neighbor [8], in which the data series is searched for situations similar to the current one each time a forecast needs to be made. This method asks for periodicity to be exploited that, as already discussed, here is not of much help.

In the past decades ANNs have emerged as a technology with a great promise for identifying and modeling data patterns that are not easily discernible by traditional methods. A comprehensive review of ANN use in forecasting may be found in [9]. Among the many successful implementations we may mention [10]. A common feature, however, of the existing application is that they ask for a relatively long time series to become effective. Typically it should be not shorter then 50 data points [9]. This is due to the fact that they all look for periodicity within the data. Very short time series were treated [10]. Here additional "nonsample information" was added to the time series in order to get statistical estimation from deterministic data.

That is why we went for a search for topological structures of ANN that promise prediction based on short time series. In the next, we will first briefly introduce the feed-forward neural networks that will be used as a basic structure for prediction throughout this paper.

The network is depicted in Fig. 4. It has only one hidden layer, which has been proven sufficient for this kind of problem [11]. Indices: "in", "h", and "o", in this figure, stand for input, hidden, and output, respectively. For the set of weights, $w(k, l)$, connecting the input and the hidden layer we have: k=1,2,..., m_{in}, l=1,2,..., m_{h}, while for the set connecting the hidden and output layer we have: k=1,2, ...m_{h}, l=1,2,..., m_{o}. The thresholds are here denoted as $\theta_{x,r}$, r=1,2, …, m_{h} or m_{o}, with x standing for "h" or "o", depending on the layer. The neurons in the input layer are simply distributing the signals, while those in the hidden layer are activated by a sigmoidal (logistic) function. Finally, the neurons in the output layer are activated by a linear function. The learning algorithm used for training is a version of the steepest-descent minimization algorithm [12]. The number of hidden neurons, m_{h}, is of main concern. To get it we applied a procedure that is based on proceedings given in [13].

In prediction of time series, in our case, a set of observables (samples) is given (approximately every two hours) meaning that only one input signal is available being the discretized time. According to (1) we are predicting one quantity at a time meaning one output is needed, too. The values of the output are numbers (average power for a period of approximately two hours). To make the forecasting problem numerically feasible we performed transformation in both the time variable and the response. The time was reduced by t_0 so that

$$t=t^*-t_0. \qquad (2)$$

Having in mind that t^* stands for the time (in minutes) during one day, this reduction gives the value of 0 to the time (t_0) related to the first sample. The samples are normalized in the following way

$$y=y^* / M \qquad (3)$$

where y^* stands for the current value of the target function, M is a constant (for example M=200, being the turn ratio of the current transformer in the station).

If the architecture depicted in Fig. 1 was to be implemented (with one input and one output terminal) the following series would be learned: (t_i, $f(t_i)$), i=1,..., m.

Starting with the basic architecture of Fig. 4, in [6] possible solutions were investigated and a new structure was suggested to be the most convenient for the solution of the forecasting problem based on short prediction base period. Here, however, having in mind the availability of data related to previous weeks, that arhitecture will be properly extended.

The structure was named *feed forward accommodated for prediction* (FFAP) and depicted in Fig. 5. Our idea was here to force the neural network to learn the same mapping several times simultaneously but shifted in time. In that way, we suppose, the previous responses of the function will have larger influence on the $f(t)$ mapping. In this architecture there is one input terminal that, in our case, is t_i.

The *Output*$_3$ terminal, or the *future* terminal, is to be forced to approximate y_{i+1}. In cases where multiple-step prediction is planned *Output*$_3$ may be seen as a vector. Alternatively (what is used in the next) in multistep ahead prediction the *Output*$_3$ terminal is seen as a scalar and should be forced to approximate y_{i+k}, where k is the number of steps in future related to the last sample known in advance i.e. the present information. *Output*$_2$ should represents the *present* value i.e. y_i.

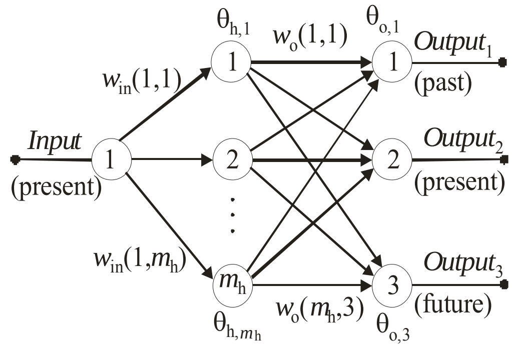

Figure 5. FFAP. Feed forward ANN structure accommodated for prediction

Finally, *Output*$_1$ should learn the *past* value i.e. y_{i-1}. Again, if one wants to control the mapping by a *set* of past values, *Output*$_1$ may be seen as a vector.

We may express the functionality of the network as

$$\{y_{i+k}, y_i, y_{i-1}, ..., y_{i-q}\} = \mathbf{f}(t_i), \; i=q+1,..,m, \qquad (5)$$

where *Output*$_1$=$\{ y_{i-1}, ..., y_{i-q} \}$, meaning: one future ($k$-steps ahead), one present, and q previous responses are to be learned.

In the case of prediction of power consumption we extended the FFAP to introduce the influence of the response values from the previous weeks at the given time of the day. In that way for the extrapolation function we may write the following

$$\{p_{n,i+k}, p_{n,i}, p_{n,i-1}, p_{n,i-2}, ..., p_{n,i-q}\} =$$
$$= \mathbf{f}(t_i, p_{n-1,i+k}, p_{n-2,i+k}, p_{n-3,i+k}, p_{n-4,i+k})$$
$$, \; i=q, ...,m. \qquad (6)$$

The new network is learning the future (unknown) value $p_{n,i+k}$, based on the actual time t_i, the actual consumption $p_{n,i}$, the past consumption values for the given day in n–th (actual) week ($p_{n,i-j}$, j=1,2,…,q), and the past consumption values for the same day at the actual time of the previous weeks ($p_{n-j,i+k}$, j= 1,2,3,4). Namely, only four past weeks are observed what was considered sufficient. The new structure is referred to as extended feed forward accommodated for prediction (EFFAP). It is depicted in Fig. 6.

The error function that was minimized during the training process, for the network structure depicted in

Fig. 6, was as follows:

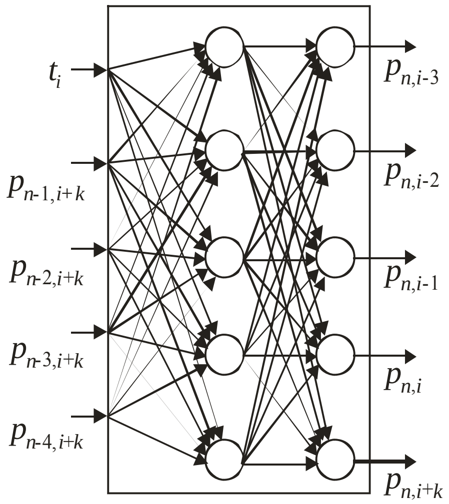

Figure 6. EFFAP. Extended feed forward accommodated for prediction ANN (q=3)

$$\delta = \frac{1}{2}\left\{\sum_{r=-q}^{0}\left[\sum_{i=q+1}^{m}\left(p_{n,i+r}(t_i,\mathbf{w},\boldsymbol{\theta}) - \hat{p}_{n,i+r}(t_i)\right)^2\right] + \right.$$
$$\left. + \left[\sum_{i=4}^{m}\left(p_{n,i+k}(t_i,\mathbf{w},\boldsymbol{\theta}) - \hat{p}_{n,i+k}(t_i)\right)^2\right]\right\} \qquad (7)$$

where $\hat{p}_{n,i+k}(t_i)$ stands for the known values, $p_{n,i+k}(t_i,\mathbf{w},\boldsymbol{\theta})$ is the actual response at the ANNs output, and $\mathbf{w}$ and $\boldsymbol{\theta}$ are vectors of unknown weights and threshold, respectively.

3. Implementation examples

The training data for the EFFAP network intended to be developed for forecasting the value of the consumption at midnight Thursday, January 30, 2009. is depicted in Table 2. The time is given in minutes. Note that in the first row, enumerated 4, the value $p_{n,i\text{-}3}$ is, in fact, the first value in the time series: $p_{n,1}$. It was measured at t_1=6 min. The lowermost row, separated by bold line, is related to the time instant where prediction should be peformed. The values given in that row will be used as excitation to the ANN obtained after training.

After training the EFFAP network with these data and exciting it as described above, the predicted value was 0.581574. It is a miss of the target value by 3.85%. In addition, the EFFAP ANN performs ideally in approximation of the load curve as can be seen from Fig. 7 where the input curve and the approximation overlap in the whole approximation interval $t\in\{0, 1382\}$. This result was obtained by an ANN with five neurons in the hidden layer as depicted in Fig. 6.

Table 3 contains the weights of the hidden neurons while Table 4 depicts the values of the weights of the output neurons, for the example above. Table 6 contains the thresholds.

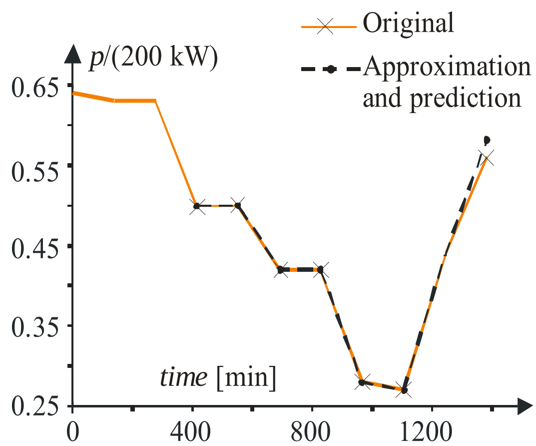

Figure 7. The actual curve (Solid line), and the approximation (Dashed line) obtained by the EFFAP network. The last segment of the dashed line finishes with the prediction

Having in mind the shape of the curve, the above prediction example may be stated as a successful one. To check the behavior of the method on a larger set of examples we repeated the above process 11 times by moving the time window by one step to generate 11 consecutive predictions. The obtained results are presented in Fig. 8. All predictions were obtained by ANNs with five neurons in the hidden layer.

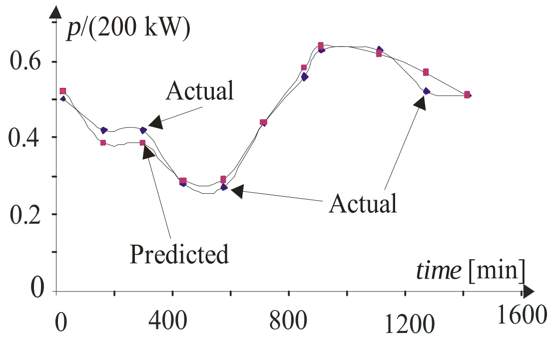

Figure 8. Predicted and actual values.

As can be seen from Fig. 8, not every prediction was as successful as the one related to the 10.16 hours in the morning. Nevertheless, most of the predictions are within 1% of the actual value and only two of them are noticeably far from the wanted one. The largest deviation is measured to be 9.38% only.

We will draw the readers attention here to the fact that the error in prediction for a given instant does not influence the next prediction i.e. there is no accumulation of the error. The reason for that is the fact that every prediction step in our method represents a separate extrapolation task.

4. Implementation to multi-step prediction

The main goal of our research was to develop a method for one-step-ahead prediction based on reduced set of data. Implementation to long term prediction was always a temptation while we are aware that it is difficult to believe that one may predict for a period in future as long as the prediction base period is. Instead, here we will give the results of an attempt to apply our method to prediction for a somewhat longer period than one-step-ahead.

There are, in our opinion, two ways of how our model may be applied for longer term prediction. First, one may use the predicted results for the time instant t_{i+1}, namely y_{i+1}, and to concatenate the input set with them. Now, the prediction may start for t_{i+2} as if one has longer prediction base period. This may be repeated as long as wanted. The problem with this idea is related to the fact that the error in prediction contained in y_{i+1} will be accumulated in the next prediction, and so on. At the end, one may have no confidence in the final long term prediction. Example of implementation of this idea may be found in [14]. It was seen that the accumulation of the error became apparent.

Table 2. Training data prepared for the EFFAP method

i	t_i	$p_{n-1,i+1}$	$p_{n-2,i+1}$	$p_{n-3,i+1}$	$p_{n-4,i+1}$	$p_{n,i-3}$	$p_{n,i-2}$	$p_{n,i-1}$	$p_{n,i}$	$p_{n,i+1}$
4	415	0.48	0.60	0.53	0.37	0.48	0.64	0.63	0.63	0.50
5	553	0.50	0.48	0.46	0.39	0.64	0.63	0.63	0.50	0.50
6	692	0.40	0.41	0.44	0.36	0.63	0.63	0.50	0.50	0.42
7	828	0.34	0.40	0.38	0.34	0.63	0.50	0.50	0.42	0.42
8	966	0.27	0.31	0.36	0.29	0.50	0.50	0.42	0.42	0.28
9	1105	0.29	0.36	0.31	0.26	0.50	0.42	0.42	0.28	0.27
10	1244	0.36	0.58	0.36	0.33	0.42	0.42	0.28	0.27	0.44
11	1382	0.64	0.70	0.50	0.43					

Table 3. Weights of the hidden neurons (w_{in})

hidden→ input↓	1	2	3	4	5
1	-1.30232	1.64487	-2.0531	+0.59616	-0.58161
2	-1.40653	-10.3729	8.3361	-0.0698999	5.50658
3	-1.02379	0.116557	0.964769	2.04944	-2.36683
4	1.51116	-2.19304	-1.05227	2.92355	-0.655558
5	-0.361701	-2.77737	-2.67532	1.66006	-4.49869

Table 4. Weights of the output neurons (w_o)

output→ hidden↓	1	2	3	4	5
1	-0.177283	1.82372	0.938738	2.13305	-0.526299
2	-5.28422	0.800788	-2.59	0.100961	-1.85201
3	-0.385013	-0.531838	1.5066	0.498187	1.02514
4	-1.5363	3.269	-3.08815	-0.0663383	-1.31226
5	4.10378	1.98327	-0.049139	-0.700042	-0.689512

Table 5. Thresholds of the hidden (θ_h) and output (θ_o) neurons

Thresholds	θ_h	θ_o
1	0.559946	0.486621
2	-0.467535	-3.61724
3	-0.665698	2.19199
4	-1.49031	-0.291289
5	1.8894	1.97652

Alternatively, one may predict two (or more) steps ahead directly by skipping the intermediate intervals. In such a case, we simply put k=2, and create proper data structures for training.

Looking to them we find that for the one-step-ahead prediction (k=1) we had m samples to be used for training and b=m-1-q "training lessons". On the other side, for multistep prediction, the number of training lessons may be stated as b=m-k-q. If the number of intervals in future, k, rises, b is diminished. It is equivalent to reduction of the prediction base period what would lead to reduction of the quality of the forecast.

This model was checked by an experimental prediction of an electricity load two steps ahead. In absolute values that would be four hours since the samples are taken every two hours. The forecasting results are depicted in Fig. 9. Here for convenience the t-axis is substituted by the sample numbers. The expected value was 0.5 while the predicted one was 0.5094. That was a miss of 1.88% only which is a very encouraging result. Note that the extrapolation was obtained at the point where the derivative of the curve is changing its sign. This result was obtained, however, with additional efforts in both the data structure and the ANN structure. To get it, the number of samples from the previous values of the load was extended to 17. In addition, the number hidden neurons was n_h=7, while q=4 was used. In that way b=17-2-4=11 "lessons" were used for training of the ANN.

We consider this result as an initial success. It is our intention to find the ANN synthesis parameters (number of samples - m, q, and number of hidden neurons-m_h) that will allow for automatic (real time) generation of the solution immediately after generation of the measured data as it was done with the one-step-ahead prediction shown in Fig. 8.

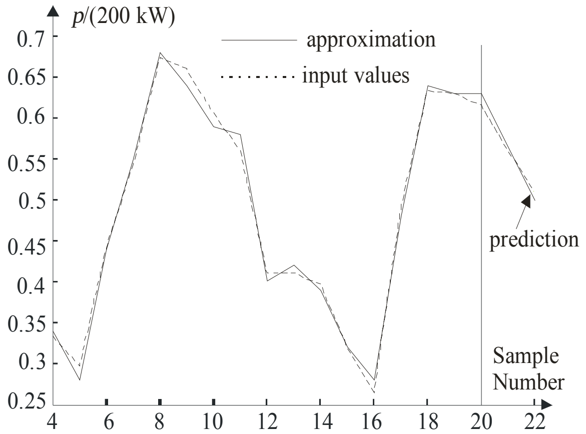

Figure 9. Approximation and two-steps-ahead extrapolation with EFFAP

4. Conclusion

The problem of short-term (hourly) forecasting of the electricity load at suburban level was considered. It is claimed first that despite "periodicity" of the phenomenon under consideration the data from previous days being from the same week or from the same day of the previous week are not convenient to be used directly. Then, an ANN structure was proposed for the solution of the problem. Both, previous data for the given day and previous data from the same day of the previous week were used in a proper manner. Encouraging results were obtained for one-step-ahead (two hours ahead) prediction were obtained. It is feasible for on line implementation. Successful experiment for two-steps-ahead prediction model was experienced. Additional work is to be done for implementation of the structure for real time (on line) run.

The solution offered performs differently as compared with the existing ones. The classical solutions are searching for periodicity which, as shown, is not a property of the modelled phenomenon. On the other side, the existing ANN based solutions ask for enormous amount of data and very complex ANNs. To our knowledge, no solution was proposed being able to perform dynamically.

5. References

[1] A. S. Mandel', "Method of Analogs in Prediction of Short Time Series: An Expert-statistical Approach", *Automation and Remote Control*, Vol. 65, No. 4, April 2004, 634-641.

[2] P. Murto, *Neural Network Models for Short-Term Load Forecasting*. MS Thesis, Helsinki University of Technology, 1998.

[3] G. Gross, F. D. Galiana, 1987, "Short-term load forecasting", *Proceedings of the IEEE*, Vol. 75, No. 12, December 1987, pp. 1558-1573.

[4] F. Cavallaro, "Electric load analysis using an artificial neural network", *Int. J. of Energy Research,* Vol. 29, 2005, pp. 377–392

[5] J. Milojković, V. B. Litovski, "New methods of prediction implemented for sustainable development", *51th Conf. ETRAN*, Herceg Novi, Monte Negro, June 2007, Paper no. EL1.8 (in Serbian).

[6] J. Milojković, V. B. Litovski, "Comparison of some ANN based forecasting methods implemented on short time series", *9th Symposium NEUREL-2008*, Belgrade, Sept. 2008, pp. 175-178.

[7] J. Milojković, V. Litovski, „*Dynamic Short-Term Forecasting Of Electricity Load Using Feed-Forward ANNs*", Int. J. of Eng. Intelligent Systems for Electrical Engineering and Communication, Vol. 17, No. 1, March 2009, pp. 38-48.

[8] E.A. Plummer, "*Time series forecasting with feed-forward neural networks: guidelines and limitations*", M.S. Thesis, University of Wyoming, Laramie, July 2000.

[9] B.G. Zhang, "Forecasting with artificial neural networks: The state of the art", *Int. J. of Forecasting*, Vol. 14, No. 1, March 1998, pp. 35-62

[10] K. Brännäs, J. Hellström, "*Forecasting based on Very Small Samples and Additional Non-Sample Information*", Umeå Economic Studies 472, Umeå University, Sweden, 1998

[11] T. Masters, "Practical Neural Network Recipes in C++", *Academic Press,* San Diego, 1993.

[12] Z. Zografski, "A novel machine learning algorithm and its use in modelling and simulation of dynamical systems", in *Proc. of 5th Annual European Computer Conference, COMPEURO '91*, Hamburg, Germany, 1991, pp. 860-864.

[13] E.B. Baum and D. Haussler, "What size net gives valid generalization", *Neural Computing*, 1989, Vol. 1, pp. 151-160.

[14] J. Milojković, V. Litovski, „*Short Term Forecasting in Electronics***",** International Journal of Electronics, ISSN 0020-7217, 2010, accepted.