Control and Cybernetics

vol. **31** (1999) No. 2

Evolutionary neural-networks based optimisation for short-term load forecasting

by

Maciej Grzenda and Bohdan Macukow

Warsaw University of Technology
Faculty of Mathematics and Information Science
Plac Politechniki 1, 00-661 Warszawa, Poland
e-mail: {grzendam,macukow}@mini.pw.edu.pl

Abstract: The purpose of short-term load forecasting is to optimise the power supply volume in short time horizon. There is no straightforward mapping rule between the type of time period and the resulting power consumption. Still, it is inevitable for the overall efficiency of the power system to rely on a good prediction model.

Our paper illustrates a novel approach based on evolutionary programming. Feedforward networks are being evolved by the *ECoMLP* method in order to properly solve the optimisation problem, defined as minimisation of the prediction error. All the results have been obtained using the data from the Polish Power System. The data used for the training and tests has been chosen so as to reflect both short-time and long-time dependencies between time period category and load of the system.

The primary feature of the described method is a novel self-adaptive procedure that is a part of a sophisticated design algorithm serving to select both network architecture and weight connections. Due to the application of this procedure, no time consuming tests are required to train and retrain neural prediction models. Therefore, the method makes it possible to construct and maintain prediction models for load forecasting without expert knowledge about neural networks.

Keywords: optimisation, neural networks, evolutionary programming

1. Introduction

The paper deals with the construction of prediction models for power consumption (see Loi, 1998, Osowski and Siwek, 1998). The way the prediction models are deployed has to reflect optimisation criterion. In the problem discussed in the actual system load as possible. In other words both insufficient volume of power supply and the power supply exceeding actual demand should be avoided. Formal description of the optimisation criterion is presented in the next section.

Load forecasting models for power systems can be based on several factors. The most important of them is power consumption on previous days and hours. In our work, the prediction model has been prepared in accordance with the following requirements:

- power demand for every hour $t \mid t = 0, \ldots, 23$ of the next day has to be predicted,
- real power consumption measured during each hour of preceding days is used as input data for the prediction model,
- type of the day has to be encoded and used as a part of input vector as well,
- for both training and testing of the prediction models the real data describing actual power demand on the territory of Poland is used.

The optimisation criterion for the prediction models built in accordance with the assumptions formulated above is described in Section 2.

Several factors affect power consumption. It is vital for the overall quality of the prediction model to determine the most important of them. Data analysis has been based on the real data describing the demand for electric power in Poland and has been used to derive general characteristics of the data. The results of this pre-processing stage have been presented in Section 3.

The problem of power demand prediction has been analysed for over 40 years. Different methods based on pattern recognition, knowledge-based systems, statistic methods and artificial neural networks have been proposed and applied during that period (see Srinivasan, 1995, for references). Some of the best results have been obtained using the latter technique (see Doveh, 1999). An important factor affecting power consumption is air temperature. However, it can be used as input data only under strict conditions. First of all, it should be uniform over the region considered or some additional information on the temperature distribution should be provided. Unfortunately, in Polish data set no temperature information can be provided. This is due to significant differences in air temperatures between different areas of the country and large disproportion in power consumption between these areas. As a consequence, the results of the simulations performed on the Polish data set can not be compared to similar results on the data sets including temperature, like the Singapore data set (see Srinivasan, 1995). In addition, data sets from real power systems are confidential, thus limiting the possibility of methodological comparisons. Therefore, we can compare our results only with these reported for the Polish data set by S. Osowski (see Osowski and Siwek, 1998). Still, we have been able to use only a part of this set.

The prediction models we have applied were based on multilayer perceptrons construction of multilayer perceptrons *ECoMLP*. The method was used to select the network architecture and weights.

ECoMLP is based on our previous experience with evolutionary design of feedforward (Grzenda and Macukow, 2000) and evolutionary neural networks (see Macukow and Grzenda, 2001 for an overview). *ECoMLP* has been based on evolutionary programming (EP) (see Bäck and Schwefel, 1993; Fogel, 1999) and allowed for parallel implementation. The design and optimisation of neural networks has been performed on SUN HPC E10000 parallel server. In order to properly exploit the opportunities offered by the parallel architecture a group of 24 processes has been deployed.

The most important feature of the method is a new self-adaptive procedure serving for the adjustment of algorithm parameters. Not only did it help to reduce the computation time, but also it allowed for automatic selection of the problem and search-related mutation rates and types. By selecting both architecture and weight connections the algorithm can be applied to create final solutions in one run of the algorithm without user interaction. Unlike most evolutionary algorithms it uses self-adaptive procedure to avoid manual selection of algorithm parameters. These features should be emphasised as they make it possible to apply the algorithm in a software package that does not require expert knowledge on neural networks and training algorithms.

The method itself and the most important implementation details are described in Sections 4 and 5.

In order to evaluate the efficiency of the training method, several tests have been made. The results of training performed by *ECoMLP* have been described in Section 6. Finally, conclusions and the main topics of future research are outlined in the last section.

2. Optimisation

In the problem discussed the primary goal is to make the power system provide suitable volume of power supply. More precisely, it is to make the power volume equal to the actual demand in a given time period. With A being the set of time periods, $\widehat{P(a)}$ – the real load volume for the time period a, and $P(a)$ – predicted load volume, we can define the optimisation task as the minimisation of the following function (see Grzenda, 2001):

$$MAPE(s) = \frac{1}{\mathrm{card}(A)} \sum_{a \in A} \frac{|P(a) - \widehat{P(a)}|}{\widehat{P(a)}}. \tag{1}$$

Thus, the lower $MAPE(s)$ is, the closer the power profile produced by the power system s to the demand observed in the set of time periods A. In our paper, the set A will be composed of a number of one hour long periods corresponding to each hour of the day. Not only does it allow to take into account the hour separately. In addition, deeper insight into the complexity of prediction process is provided. Once again, proper minimisation of the criterion (1) can be obtained only when the proper prediction model producing adequate $P(a)$ values is provided.

The purpose of the next section is to present the most important tendencies in load volume profiles. It should also help verify the validity of one hour long time periods approach.

3. Data profiles

The primary purpose of this pre-processing stage is to determine the most important factors, crucial for satisfactory prediction. In order to evaluate the way different types of day categories affect the demand for power, a set of statistics has been prepared. All of them have been based on real data collected during three years on the territory of Poland. The most important of them are:

- the impact of the type of the day on power consumption,
- the impact of the day of the week on power consumption.

The most important features of the so-prepared demand profiles are:

- virtually identical power consumption on Sundays and bank holidays,
- close overall characteristics of power consumption between Monday and Friday,
- considerable differences between Sundays, Saturdays and other days.

The differences related to the day category e.g. the difference between the load of the power system observed on Fridays and bank holidays can be as large as 20–25%. Moreover, similarly large differences can be observed within the same day, in particular between mid-night hours (the so called *valleys*) and late evening (*peak* consumption). Therefore, the power system should be capable of following the day profile and changing the power supply volume continuously.

4. Prediction models

In accordance with the optimisation criterion (1), the power system s should be made capable of producing power demand equal to the actual load in a set of time periods. In order to let the power system acquire the expected behaviour, we have to be able to predict the actual load of the system for each of the time periods $a \in A$.

In our approach the multilayer perceptrons MLPs (see Haykin, 1999) have been used as a prediction model. The input of each network is composed of data from the preceding days and hours - both day categories and actual load volume. The output of the network is the forecasted load for the given hour of the next day. All the networks are being constructed under the assumptions listed in the introduction. These assumptions stem from the availability of the real data and the capabilities of the power system. The next section deals with

5. Algorithm

The optimisation algorithm has been based on the novel *ECoMLP* method. The purpose of the algorithm is to minimise the prediction error measured on a learning set and obtain networks with generalisation abilities, thus making it possible to use the final networks for the purpose of load prediction. Therefore, the final networks should enable management of power supply volume assuring optimisation in terms of (1).

Evolutionary construction of multilayer perceptrons *ECoMLP*

- $t := 0$,
- create initial population of MLPs $P(0) = \{s_i \in I \mid i = 1, \ldots, N\}$ satisfying the following assumptions:
  - fixed number of input layer nodes $\mathrm{card}(L_0)$ and output layer neurons $\mathrm{card}(L_M)$ defined by the problem,
  - random number of hidden neurons,
  - randomly chosen connection weights,
- in a sequence of generations construct consecutive populations $P(t)$ by:
  - applying standard EP selection and promoting the best $\frac{N}{2}$ networks to the next population, duplicating them and producing child sub-population thereafter; the networks are evaluated in terms of their compliance with (1), i.e. the closer the predicted load of the system to the actual one, the better the network,
  - affecting offspring networks with the mutation operator composed of:
    * hidden node mutation (add or delete a neuron with probability p_a and p_d, respectively),
    * weight mutation in accordance with the learning rule of each individual (see below),
    * mutation of learning rule with probability of p_{lr},
- $t := t + 1$.

As far as the number of hidden neurones is concerned, it is randomly chosen from a predefined range. Then it can be changed by the mutation operator within the same range.

The weight mutation has been strictly based on the form of the *learning rule* controlling the behaviour of the mutation operator.

Learning rule and weight mutation

- let us define for each individual in the population $a_i \in P(t)$ the learning rule (Met_i, Par_i) where:
  - $Met_i \in \{0, 1\}$ stands for mutation type, discrete or real,
  - Par_i is defined by $Par_i = (p_m)$ for the discrete type and $Par_i =$

- mutate the weights of the network in accordance with the learning rule of the individual, using uniform distribution over a specified range of weights, in case of discrete mutation, or Gaussian distribution, otherwise; thus Par is used to store the distribution settings,
- mutate learning rules as well.

Every network can be mutated using a different learning rule. As a consequence, numerous search strategies are applied and verified in each generation. The way learning rule affects the weight mutation is described by the following operator:

Weight mutation operator

For each offspring network s_i:

- if $Met_i = 0$, each network weight w is replaced with probability p_m with another valid discrete weight $w' \in \Theta$; the range of weights Θ should be chosen in accordance with the neuron activation function so as to ensure that virtually all function values are feasible;
- if $Met_i = 1$, each network weight w is replaced with probability p_m with a disturbed weight value $w' := w + N(0, \mu)$.

The learning rules are also mutated with probability p_{lr}. As a consequence, mutation mode can be switched between discrete and real. In addition p_m and μ can be randomly changed as well, with the same probability p_{lr}.

As a consequence of learning rule mutation and standard selection mechanism, appropriate mutation types and attributes are promoted in the population. Still, the changes in attribute settings are possible, so as to reflect the different nature of initial search and final adjustment of weights. Having applied this method, we can easily obtain the solutions close to the optimum in one run of the algorithm. In other words, learning rules help overcome one of primary deficiencies of the evolutionary algorithm – strong impact of the algorithmic parameters on the search efficiency and lack of straightforward methods for managing them. In the proposed algorithm, standard selection promotes the best weight mutation attributes. What is important, optimal mutation settings depend on the search stage. In the early stages large changes to the genotypes may provide the best results. These are unlikely to improve the performance of the networks in the final stages of the search. On the contrary, only slight modifications should be used. Learning rules allow to reflect these guidelines in a straightforward method.

The solution belongs to the class of self-adaptive control methods and allows for serious reduction in computations. For an overview of different control methods see Bäck (1998).

6. Prediction models

A set of $T = 24$ networks has been evolved using the $ECoMLP$ method. They day. Two sets of experiments have been prepared. The results of the series A have been obtained for the training and validation patterns collected from the whole available period and are listed in Table 1. The second series has been prepared on the data set with discarded input patterns corresponding to the prediction for bank holidays, the days that precede and follow the bank holidays. The results of the computations on the so-prepared data set are summarised in Table 2. In both cases the error rates for the best network in the population $P(20000)$ are provided. Thus the search complexity for different hours of the day can be compared.

In all the algorithm runs the following settings have been applied:

- the range of the number of hidden neurons: $1, \ldots, 3 \times \mathrm{card}(L_0)$; this range has been used for the initialisation of the networks,
- $p_m \in (0, 0.3]$ for the discrete mutation mode $Met = 0$ and $\Theta = [-3, 3]$,
- $p_m \in (0, 0.7]$ and $\mu \in (0, 0.2]$ for the real mutation mode,
- probability of learning rule mutation p_{lr}=0.05,
- probability of adding/deleting a single neuron $p_d = p_a = 0.1$.

In both cases MAPE and MAE errors have been computed both on the learning and validating set. The error measures have been computed using the following formulas:

$$MAPEL(s^t) = \frac{1}{\mathrm{card}(A_L^t)} \sum_{i=1}^{\mathrm{card}(A_L^t)} \frac{|P(i) - \widehat{P(i)}|}{\widehat{P(i)}} \cdot 100\% \tag{2}$$

where A_L^t is a learning set for time t, $\widehat{P(i)}$ stands for real value, $P(i)$ denotes predicted value, and

$$MAEL(s^t) = \frac{1}{\mathrm{card}(A_L^t)} \sum_{i=1}^{\mathrm{card}(A_L^t)} |P(i) - \widehat{P(i)}| \cdot 100\%. \tag{3}$$

MAEV and MAPEV are computed against a separate set A_V consisting of the data patterns from another year and not used for network evaluation, thus assuring unbiased evaluation of the final networks. It is crucial for the overall assessment of the networks to evaluate their generalisation abilities (see Haykin, 1999). In other words, the prediction error has to be computed on a separate set of validation data patterns and compared to the error obtained for the learning set. The learning and validation set errors are referred to as MAPEL and MAPEV in our work, respectively. If neural network is able to represent the general rules between input and output data, it should predict the load demand for different input patterns with a similar quality. However, if network does not gain the understanding of these rules and just reproduces the expected output for known patterns, it can not properly react to new data patterns. The MAPEV error rate is greater than MAPEL error rate, then. In the latter case, the neural network provides little or no practical value, as it can not be used the demand forecast for the next day. In order to evaluate the generalisation abilities, we have used the difference MAPEV-MAPEL as a general measure. The closer to zero it is, the better generalisation has been obtained by the network.

Table 1. Series A - results for all the day categories

Hour	MAPEL	MAPEV	MAPEV-MAPEL	MAEV
0	2.24	2.21	-0.03	1.33
1	2.54	2.60	0.06	1.55
2	2.99	3.31	0.02	1.95
3	1.94	1.94	0.00	1.14
4	2.88	3.00	0.12	1.76
5	3.29	3.69	0.40	2.16
6	3.06	3.23	0.17	2.00
7	3.85	3.60	-0.15	2.25
8	3.84	3.52	-0.32	2.25
9	3.76	3.53	-0.23	2.51
10	3.39	3.43	0.04	2.28
11	3.61	3.60	-0.01	2.35
12	3.06	3.10	0.04	2.05
13	3.71	3.79	0.08	2.34
14	3.58	3.47	-0.11	2.25
15	4.13	4.11	-0.02	2.61
16	3.72	3.89	0.17	2.52
17	3.23	3.26	0.03	2.14
18	3.19	3.15	-0.04	2.70
19	3.34	3.36	0.02	2.30
20	3.42	3.26	-0.16	2.20
21	2.52	2.52	0.00	2.52
22	2.48	2.63	0.15	1.67
23	3.27	3.19	-0.08	1.92
avg	3.21	3.22	0.01	2.01
min	1.94	1.94	-0.32	1.14
max	4.13	4.11	0.40	2.80

The results show that:
- an appropriate prediction model has been obtained,
- strong relation between time of the day and error rate can be noticed, the latter tends to be higher in the mid-day hours,
- considerable impact of bank holidays and the neighbouring days on the overall quality of the prediction can be observed; the error rates are significantly lower when the input patterns corresponding to these days are

- good generalisation ability of the thus obtained networks has been achieved, the average difference between the error rate measured on training and validation set is close to zero.

Table 2. Series B - results for all the day categories but bank holidays and the neighbouring days 2

Hour	MAPEL	MAPEV	MAPEV-MAPEL	MAEV
0	2.14	2.46	0.32	1.45
1	2.44	2.63	0.19	1.54
2	2.02	2.35	0.32	1.41
3	3.01	3.01	0.00	1.77
4	2.88	3.00	0.12	1.82
5	2.97	2.68	-0.29	1.58
6	2.94	3.14	0.20	1.95
7	2.94	2.84	-0.10	1.81
8	2.98	2.68	-0.30	1.79
9	2.83	2.81	-0.02	1.94
10	2.35	2.39	0.04	1.58
11	2.55	2.58	0.03	1.78
12	2.87	2.48	-0.39	2.05
13	2.62	2.70	0.08	1.78
14	2.71	2.67	-0.04	1.77
15	3.37	3.19	-0.18	2.19
16	3.89	3.78	-0.12	2.56
17	3.05	2.72	-0.33	1.82
18	3.97	4.13	0.16	2.71
19	2.55	2.48	-0.07	1.66
20	2.08	2.19	0.11	1.49
21	2.48	2.59	0.11	1.71
22	2.29	2.32	0.03	1.51
23	3.01	3.01	0.00	1.79
avg	2.67	2.79	0.12	1.81
min	2.02	2.19	-0.30	1.41
max	3.97	4.13	0.32	2.71

To sum up, the results of the computations show that $ECoMLP$ is capable of providing fine-tuned networks by selecting both their architecture and connection weights. Preliminary results suggest changes in the input attributes as the most promising way of further improvements. The stability of results has been also checked. For selected time unit 30 runs have been performed, resulting in average MAPEL error 3.19 and standard deviation 0.201. Average difference in error rate compared to the validation set has been lower than 0.05%. In final *ECoMLP* networks have been obtained without exhaustive search for network architecture and do not require the OBD procedure (see Haykin, 1999) to ensure proper generalisation. Even though only a part of the whole set has been available for our experiments, better generalisation and error rates have been achieved.

The structures of the prediction models based on the final neural networks vary strongly in size and complexity. For instance, the sample structure of the network for the 3:00 a.m. hour prediction task is 29-2-1, which stands for 29 input nodes (24 hour loads + type of the day + time of the year), 2 neurons in the first hidden layer, 1 hidden neuron in the second layer. The sample structure for 3:00 p.m. is 29-9-1. In general, the number of input nodes and output neurons is the same for all the networks. The algorithm prefers simple network structures based on limited number of neurons to more sophisticated ones. This is due to the fact that appropriate weight settings are easier to find when lower number of neurons and possible connections exists. Simulation results show that additional neurons are added only when necessary. It is also reflected by proper generalisation of the networks. If the network structure was inadequately large, the performance of the network on the validation set A_v expressed by MAPEV would worsen (see Haykin, 1999).

When compared with load profiles described in Section 3, the difference in complexity of the final networks seems to be quite natural. During the night the load profiles tend to be close to each other, no matter which type of the day is analysed. This is due to the static part of the demand resulting from by the continuous work of industry. On the other hand, the load for the mid-day hours is much more related to the type of the day. Thus, the network corresponding to one of the afternoon hours has to be able to properly reproduce some very different load profiles. The latter results in a much more complex structure of neural connections and increase in their number, as well.

7. Conclusions

The parallel implementation of the *ECoMLP* method has provided a set of prediction models for 24 time periods of the day. Depending on the time of the day the mean average error MAPEV ranged from approximately 1.9% to 4.11%. As a consequence, the neural-networks based optimisation of power supply volume has been obtained. As the power system enriched with the neural prediction model can properly predict the load volume, it can also adapt to the changing power demand and provide power volume close to the actual demand.

Due to the different power consumption profiles for different day categories, the quality of prediction models strongly depends on the time of the day. As a consequence, further development of a new heterogeneous prediction system can be considered. For the night periods one neural network-based model can set of models can be constructed. Each of these models can be made responsible for the power consumption prediction for different categories of days, e.g. bank holidays and other days. In other words, the complexity of the solution of the optimisation problem can reflect the complexity of the prediction problem for each hour separately.

What is important, all the results have been obtained in one run of the algorithm and did not involve time-consuming computations. Still, the stability of results has been checked as well.

It should be emphasised that the proposed method, unlike most other neural network methods, selects the network architecture and weights at the same time. In addition, the self-adaptive procedure makes it possible to avoid manual setting of algorithm attributes. Therefore, parallel implementation has been made available. Traditional methods frequently use trial-and-error method to select architecture or the parameters of the evolutionary algorithm.

To sum up, the *ECoMLP* method has successfully created prediction models. These models can be used to optimise the power supply volume by reducing the difference between the power supply volume and the actual demand for the system.

8. Acknowledgements

The computations for this work have been performed using the support from the Centre for Computer Science of the Warsaw University of Technology and the account on SUN HPC E10000.

References

BÄCK, T. and SCHWEFEL, H.P. (1993) An Overview of Evolutionary Algorithms for Parameter Optimization, *Evolutionary Computation*, **1**, 1, 1–23.

BÄCK, T. (1998) An Overview of Parameter Control Methods by Self-Adaptation in Evolutionary Algorithms. *Fundamenta Informaticae*, **35**, 51–66.

DOVEH, E. et al. (1999) Experience with FNN Models for Medium Term Power Demand Predictions. *IEEE Transactions on Power Systems*, 538–546.

FOGEL, D.B. (1999) An Overview of Evolutionary Programming. *The IMA Volumes in Mathematics and its Applications*, **111**, 89–109.

GRZENDA, M. (2001) *Application of evolutionary programming in the design of multilayer networks.* Ph.D. thesis (in Polish), Warsaw University of Technology.

GRZENDA, M. and MACUKOW, B. (2000) The Role of Weight Domain in Evolutionary Design of Multilayer Perceptrons. *Proc. of the IEEE-INNS-ENNS International Joint Conference on Neural Networks IJCNN 2000.*

HAYKIN, S. (1999) *Neural Networks: a Comprehensive Foundation.* Prentice-

LOI, L.L. (1998) *Intelligent System Applications in Power Engineering. Evolutionary Programming and Neural Networks.* John Wiley and Sons.

MACUKOW, B. and GRZENDA, M. (2001) Towards the Evolution of Neural Networks. *Opto-Electronics Review*, **9** (3), 316–319.

OSOWSKI, S. and SIWEK, K. (1998) Selforganizing Neural Networks for Short Term Load Forecasting in Power System. *Engineering Applications of Neural Networks (EANN)*, 235–268.

SRINIVASAN, D., CHANG C.S. and LIEW, A.C. (1995) Demand Forecasting Using Fuzzy Neural Computation, With Special Emphasis on Weekend and Public Holiday Forecasting. *IEEE Transactions on Power Systems*, **10**, 4, 1897–1903.