Neuro-fuzzy Prediction Systems in Energetics

Tomasz Peszek

Jagiellonian University, Institute of Computer Science
Nawojki 11, 30-072 Kraków, Poland
e-mail: tpeszek@wp.pl

Abstract. The paper concentrates on problems of short term electrical load forecasts. It focuses on application of hybrid systems based on neuro-fuzzy components for making such predictions. Subsequently basic concepts concerning modelling of such systems and examples of working solutions are introduced. Finally the method and the system created by the author is presented and discussed.

Keywords: artificial intelligence, hybrid systems, energy forecasts, energetics.

1. AI techniques for short term energy consumption forecasts

The prediction systems based on artificial intelligence are significant part of working systems which forecast electrical load in short term. Among them the neural, the fuzzy and the hybrid systems with the neuro-fuzzy techniques play and important role. The mentioned AI-system types are complementary, i.e. both have certain advantages and disadvantages which determine the decision on usage of a certain technique in a particular context. Hybrid systems should concentrate on extracting the strengths of components and diminishing their weaknesses. Below several examples of systems based on neural and fuzzy techniques working in reality are introduced and analyzed.

Among described examples the focus is on systems supporting creation of short term forecasts.[1]

1.1. Neural networks in forecasting

Classical concepts on neural networks offer at least two solutions supporting the process of electrical load forecasting:

1. The layered neural networks (perceptrons) are capable of modelling the functions which are not known *a priori*. The process of modelling is based on the set of the function known values. The difficulty sticks in the fact, that the shape of the function is not known. Even the set of the arguments implying its fluctuations can remain unrecognized. In addition the data representing a modelled function can contain a certain rate of noises. Perceptron networks cope quite well with problems of this type.

2. The Kohonen neural networks are applied as far as the pattern recognition problems are concerned. In the set of historical data there are several patterns extracted. The neural network can determine, using the algorithms for vector similarity, whether presented pattern belongs to the set of already recognized patterns.

The most important advantages of neural networks are:

1. The computational cost of creating a compact solution is very low, even if the knowledge accumulated in the network is detailed and vast. The cost of neural network work is surprisingly low especially in comparison to the fuzzy systems with a similarly extensive knowledge base.

2. Another strong point of neural networks is reliability (because of the distributed representation of knowledge) and greater generalization abilities of the possessed knowledge.

3. Neural techniques are successfully exploited for grouping historical data and finding similarities among them. It is not required to set *a priori* the criterions, on base of which the data should be grouped. As far as the mathematics is concerned, the neural networks try to divide the given set of patterns using the scalar product. As a result of application of simple tools, the majority of executed operations have simple

---

[1]Such forecasts should not reach beyond the 24 hours from the moment of creation.

nature. In addition there is quite easy to manipulate with the degree of knowledge minuteness.

4. Simple architecture of solutions built on Kohonen-type networks makes a hardware implementation of such mechanisms possible. This can positively influence performance and efficiency parameters of the system.

The specification of neural networks should be finished by mentioning the weak points of that technology:

1. The implicit knowledge representation. The information contained in the weights vectors of the neurons decodes whole knowledge of the network. The problem of its extraction is difficult and computationally expensive.

2. Despite that the information is distributed in the network, the loss of a particular neuron causes the loss of the knowledge contained in it. If we consider network neurons as some terms, we can say that from the moment of the neuron loss, the network fails to operate on that term.

The problems of neural network application are analyzed and discussed in [1, 2, 4, 6].

1.2. Fuzzy prediction systems

Fuzzy systems are systems working on the grounds of a fuzzy-sets-language. The terms describing the system environment are modelled by means of linguistic variables and fuzzy sets representing their values. The language of fuzzy sets is quite similar to the natural language, which makes the term translation between languages easier. The natural way to create a fuzzy system is to describe the problem with the common language and then to translate, to formalize the description by the model consisting of fuzzy sets, linguistic variables, fuzzy rules. On the base of knowledge collected in fuzzy sets and rules the system conducts the process of conclusion and generalization.

The main advantages of fuzzy systems are:

1. The possibility of formulating the solution as well as the problem description by means of rules of type $IF-THEN$, which are built of the natural language terms. The set of rules collected by the system can be partially mutually contradicting. It is the consequence of the heuristic description of the problem.

2. The knowledge of the system is easily accessible for the user and advanced tools are not necessary to extract it. The knowledge implementation makes its analysis and management easier.

3. The prognoses and conclusions are easy to reason. A simple algorithm is able to confirm the predictions value. The most weighting rules can be mentioned with only less effort. These factors makes the predictions more trustworthy.

4. The process of acquiring the knowledge does not require expensive algorithms and allow to make use of the newly integrated information immediately. The adaptation abilities and knowledge accessibility make the fuzzy system a source of the new domain knowledge.

The mentioned advantages of fuzzy prediction systems, in particular the explicit knowledge representation, are paid for by several weak points. We have to mention the following items:

1. The great computational expense that is to be taken in order to make use of the collected information. This expense grows dramatically together with the growth of number of rules contained in the system.

2. An important problem to be solved before the start of such system is to secure the system before the situation in which the system cannot recognize the presented pattern on the base of the possessed knowledge. In this situation other mechanisms should be able to generate the prediction. After analysis of such case its exhausting description should be added to the knowledge base.

3. Another problem concerning the fuzzy-rules-systems working on tremendous knowledge bases containing thousands of facts and rules is maintaining their consistency and keeping them up to date. The facts gathered in such extensive bases are very precious data source for various trends analysis.

The readers willing to inquire into the fuzzy systems application should necessarily consider reading [4].

1.3. Forecasts from hybrid systems

The conception of the system introduced in the following chapter makes it necessary to focus on another type of predicting systems, i.e. the hybrids. The systems containing components of different architectures may be very

efficient predicators. The application of different types of hybrid systems are analyzed in [4].

The system designed by the author of this paper is based on neural and fuzzy components managed by a supervisor equipped with some set of tools, mainly statistical, to evaluate the effects of component work. The positive features of all systems, conception of which were used to build the mentioned system, have been strengthened whereas the weaknesses have been diminished.

Main advantages of the solution are:

1. A partially explicit knowledge representation. It makes possible to verify the reasons on the base of which particular forecast is created. Such system can generate new domain knowledge on the modelled phenomenon.

2. Easy accessibility of the knowledge and the possibility of its immediate application.

3. The reliability implied by the nature of a hybrid system, in particular by the distribution of forecasts sources. The probability of occurring of a pattern that cannot be recognized by the system decreases significantly. In such case the hybrid generates its prediction alternatively, for example by means of statistics.

4. Significant difference in exactness of generated forecasts. This progress is achieved by use of a separate algorithm for control, estimate and synthesis of the partial predictions to the one consistent forecast of the system. This mechanism is mainly supported by simple statistical tools.

It is well known that the hybrid system is as weak as its weakest component. This principle is fully confirmed by the case of the described hybrid. Its disadvantages are to be conducted from the weak points of its components. Here the following points must be stressed:

1. Computational expensiveness on generating a synthetical forecast consumed mainly by the fuzzy component of the hybrid.

2. The heterogenity of a knowledge representation at different components, which can negatively influence the possibility of its management, analysis and comparison. Algorithms supporting one of the mentioned tasks could occur expensive.

1.4. Statistical methods for verification

Apart from classical concepts based on artificial intelligence effecting in many working prognostic solutions we should mention statistics as a domain offering a wide spectrum of tools that allow to verify the generated outcomes *ex ante*. A convincing example is the system introduced in the following paragraph and the analysis of the growth of exactness as a result of use statistic tools to verification and aggregating partial forecasts.

2. A neuro-fuzzy hybrid to forecast the energy consumption

The current paragraph aims to present the concept of a hybrid system predicting short term energy consumption. From the side of architecture the system remains a hybrid built from two parallel working modules. Above them there is a supervisor module controlling the work of the components and executing the procedures for verification and synthesis of partial forecasts. In case of predictions exceeding the statistically probable range supervisor decides to reject the forecasts and generate them in an alternative way.

The particular model for the entire prediction process, such as the set of explaining variables necessary for the description of the unknown dependence, was set on the base of solutions working successfully. The main factors determining the fluctuations of the energy consumption have been chosen as a result of the process of advanced statistical analysis of dependencies present in the set of historical data. Already simple analysis of data on energy consumption and the changes of minimal and maximal air temperature changes discovers strong negative linear correlation between them. Table 1 contains the registered values of McPhearson linear correlation coefficients of energy consumption vs minimal and maximal air temperatures.[2]

A neural component is a one layered Kohonen-type network with 35-dimensional input vectors. Every input vector consists of the following parts:

1. $ZD(k)$ daily electrical load demand registered in day k.

2. $ZG1(k-1), ..., ZG24(k-1)$ hourly electrical load demand registered in day $k-1$.

---

[2]The correlation coefficients have been computed on the data that was used for the hybrid implementation and tests. The set of data will be precisely described in the following paragraph.

Tab. 1. Correlation between minimal, maximal temperature and the energy consumption registered in the test data set

Year	McPhearson corr. coefficient max temperature vs energy cons.	McPhearson corr. coefficient min temperature vs energy cons.
2002	−0.86	−0.84
2003	−0.86	−0.83

3. $TMin(k)$, $TMax(k)$, $TMin(k-1)$, $TMax(k-1)$ denote the minimal and maximal temperature in the day k as well as the forecast for minimal and maximal temperature for the day k.

4. $D1(k), ..., D6(k)$ the set of variables decoding a particular weekday. For example if tuesday is to be decoded then all variables apart from $D2$ are zero. The value of $D2$ is one.

The form of data accepted by the fuzzy system looks different. Generally speaking they have to be aggregated before they are passed to the rule – processing – engine.

1. Variable $ZD(k)$ denotes the daily electrical load demand in the day k as well as the forecast of this consumption.

2. $TMin(k)$, $TMax(k)$, $TMin(k-1)$, $TMax(k-1)$ denote the minimal and maximal temperature in the day $k-1$ as well as the forecast for minimal and maximal temperature for the day k.

3. $DT(k)$ variable describing the weekday, taking values from the set $\{1, ..., 7\}$, for the present day of week.

A fuzzy rule engine receives the input vectors in the following form

$$(ZD(k-1), \, TMin(k-1), \, TMax(k-1), \, TMin(k), \, TMax(k), \, DT(k))$$

and tries by means of collected knowledge to match them with to one of already known patterns.

The work of a hybrid system starts with the phase, in which the neural network learns to recognize the patterns of the form: 34 explaining variables defined above, one variable explained denoting the typical daily electrical load for the defined context. Here we have to stress the following point that distinguishes the process of learning from the real-time work. In the learning phase we know exactly the whole pattern, including even its last component (forecasted daily electrical load demand). In the reality we deal with missing patterns, because we only know the context and we have to guess the forecast.

That is the reason, why we have to set an initial forecast in an alternative way. At this point we are supported by statistics.

In the proces of learning the network is built step by step, starting from a single neuron and adding the new ones when necessary. Let us explain when creating a new neuron is reasoned. It seems that a new neuron should be added if none of the existing ones has reacted to the presented pattern strong enough. In order to investigate the case of adding a new neuron, let us make a short digression.

We want the network to classify the set of data. The output of such process should contain the certain number of clusters, i.e. groups of similar patterns. Clusters can be represented as centroids of such groups. From the mathematical point of view we can consider clusters as some subsets of the unit sphere in a fixed n-dimensional linear subspace of patterns. We must now understood what significance the metric size of the cluster has. The bigger the maximal allowable diameter of these subsets is, the less clusters are likely to be created and the more heterogenous their contents becomes. As a result the analysis of particular classes is more difficult. So we should choose the diameter of the clusters in such a way, that their predictable contents becomes as homogenous and easily characterizable as possible. The maximal diameter of the cluster should be the parameter of the learning process.

Returning to the necessity of adding a new pattern, it is the case when the presented pattern, considered as a vector from the n-dimensional unit sphere, lays in a bigger distance from the centroids of all groups than the maximal allowed cluster diameter. Then it becomes impossible to assign any cluster without breaking the constraint of the maximal cluster diameter. The only way out of the situation is to create a new class corresponding to the current pattern. The outcome of the learning process is a network containing knowledge about all situations occurring in the test data set encoded in recognized clusters. The number of the clusters is obviously the function of the particular test data set and the maximal allowable cluster diameter.

After the network is learnt, we try to translate this knowledge into the langauge of fuzzy terms and rules. Before translation the linguistic variables and fuzzy sets should be defined. We accept the following model:

1. Linguistic variable *weekday* taking values of the set:

   {monday, tuesday, wednesday, thursday, friday, freeday}.

2. Linguistic variables encoding climatic factors *temperature_today_min/max* and *temperature_yesterday_min/max* taking values of the set:

   {strong_negative, average_negative, weak_negative,

zero, weak_positive, average_positive, strong_positive}.

Particular fuzzy sets are defined on the closed interval [–35,35] by means of a Gauss-type function of partity with optimized parameters. The unit of temperature is one Celsius degree.

3. Linguistic variables describing the electrical load demand *electrical_load_today* and *electrical_load_yesterday* taking values:

{very_small, small, average_small, normal, average_big, big, very_big}.

Again particular fuzzy sets are defined on the closed interval [110,250] by means of a Gauss-type function of partity with optimized parameters. The unit of energy consumptions is 100 MWh (Megawatthours).

In the described fuzzy system there is a classical algorithm of fuzzy reasoning implemented. It is based on the sequential processing of all fuzzy rules and aggregating outcomes.

For the input vector containing the real numbers and every fuzzy rule the firing power of the ancestor is computed. For this purpose we use the correlation formula based on the minimum function. The other method to compute the firing power of the ancestors is to take the product of the firing power of all parts of the ancestor. However it leads to too big defuzzyfication of the rules, which may negatively influence the reliability of the fuzzy system.[3] For every fuzzy set corresponding to the value of the linguistic variable *electrical_load*, we cut it to the level of value of the minimal firing power among the rule ancestor terms. Subsequently we join the cut fuzzy sets to the one by taking the maximum operator and then the outcome is underlaid to the process of defuzzyfication by use of a *center of area* formula, i.e.:

$$\bar{x} = \frac{\sum\limits_{i=110}^{250} i\mu(i)}{\sum\limits_{i=110}^{250} \mu(i)},$$

where μ is the function of partity of the defuzzyficated fuzzy set and i its domain.

In this way the system generates its forecast on the base of the input data and the knowledge accumulated in fuzzy rules. Both components work parallel generating the pair of forecasts. On the base of this pair and the history

---

[3]The more defuzzyficated the rules are, the bigger is the number of cases, which the system does not recognize (even on generalization of the collected knowledge).

of predicting the supervising module generates the single value containing the final forecast. Before the algorithm of synthesis is precisely described we have to stress the following fact. Analyzing the series of the daily electrical load demand values we can easily see that the serie is cyclic, changing in a weekly period. The important property implied by that fact is also that the variable containing the change on energy consumption between adjacent days is cyclic. Let us denote the difference $\Delta_k = E_k - E_{k-1}$, where E_k is the daily electrical load demand in the day k. In this notation we can easily express the obvious dependance

$$E_t \approx E_{t-1} + \Delta_{t-7}. \tag{1}$$

The partition of the variable Δ carries a piece of important information. In particular by means of formula 1 we can estimate the interval which the expected energy consumption value should match. We will consider the value computed on the base of formula 1 as the *initial forecast* of the system and denote it by $\widetilde{E}_t$. As δ we will denote the maximal, assumed in advance, expected difference between the real value of energy consumption and the initial forecast. The forecasted value returned by one of the subsystems will be called *allowable* if it is contained in the interval $(\widetilde{E}_t - \delta, \widetilde{E}_t + \delta)$. Forecasts which occur not allowable will be rejected by the supervising module.

At this point we are ready to precise the algorithm for constructing the final forecast of the system:

1. As an initial forecast we take the value defined by use of formula 1.

2. Receiving the pair of forecasts (the neural and fuzzy one), the supervising module checks if they are allowable and then takes an appropriate action.

   (a) If both values occur not allowable then the initial forecast becomes the final one.

   (b) If the only one forecast is allowable then it should become final.

   (c) If both predictions are allowable the system should prefer the forecast from the system which predicted better yesterday.

3. System's outcomes

In the current paragraph we analyse the outcomes of the system descried in the previous section. We also discuss the influence of the particular parameters on the system and the choice of data for the learning phase. The

data used for tests of the implemented solution concern the electrical load demand and air temperature changes for the area of Kraków in years 2002 and 2003.

3.1. Discussion of parameters for reliability, efficiency and performance

We start the analysis of the system work with investigating the volume of knowledge stored in the hybrid knowledge base. The neural component stores its knowledge in neurons, to be more precise in their weight vectors. In case of the fuzzy system the information is kept in form of fuzzy sets, their partity functions and fuzzy rules. The ancestors of these rules describe the context of a particular situation and the consequents suggest the probable electrical load demand.

At the beginning of the system work the neural network is started. On the grounds of its outcomes the knowledge of the fuzzy system is established. There is the function which is capable of translating a weight vector of a neuron to a fuzzy rule describing this situation. The function scales the particular weight vector of the neuron,[4] distinguishes its parts and extracts them. They should contain:

1. $ZD(k)$ the likely electrical load demand in the day of forecast.

2. $ZD(k-1)$ the electrical load demand denoted in the previous day.

$$ZD(k-1) = \sum_{i=1}^{24} ZG_i(k-1).$$

3. The set of minimal and maximal air temperatures registered at the previous day and the forecasted temperatures on the current day.

4. $DT(k) \in \{1, 2, 3, 4, 5, 6, 7\}$ the variable encoding the particular weekday.

$$DT(k) = T(D1(k), D2(k), D3(k), D4(k), D5(k), D6(k)),$$

where T is a function transforming the set of variables into a one dimensional variable.

---

[4]The weight vectors from the network of the Kohonoen type – network have the length 1.

In this way the information interchange between fuzzy and neural systems works.

The neural network operates on patterns considered as the vectors taken of the unit sphere in a fixed n-dimensional linear subspace. This implies the fact, that we have a good measure of the distance between them, i.e. the angle between them. To be more precise it is enough to control the cosinus of this angle and further the scalar product of these vectors. As a result of the fact that both the weight vector and the input vector are taken of the unit sphere, the scalar product of the vectors is identical with the cosinus of the angle between them. We will impose the constraint of the maximal cluster diameter by determining the value of the angle between the n-dimensional vectors. The particular constraint should have the form

$$< v, w > = \cos(\angle(v, w)) > \alpha,$$

where v, w are considered vectors, and $\alpha \in [-1, 1]$ is a set real number.

Before we present the table displaying the influence of the maximal cluster diameter of a cluster on the system work, we have to explain the procedure of choice of the elements for the test set.

Each of the input vectors containing data from the year 2002 was presented to the network from one to five times. Such construction of the test data set was reasoned by experiments in which the patterns were presented to the network once. Subsequently the errors made by the network were measured. The highest mistakes were made in the period of late spring, summer, early autumn. The system made the best forecasts in late autumn, winter and early spring. That is the reason why in the test set several copies of the same pattern occurred. The bigger mistake was made by the network while analyzing a particular pattern the more times it was repeated in the test data set. Each pattern was repeated in the test data set up to five times.

Having prepared a proper test data set and determined the maximal allowable cluster diameter, we start the network to classify the data with a certain degree of minuteness. Table 2 presents the outcomes of the learning process on the test data set containing approximately 1000 patterns (not necessarily mutually different).

The number of rules created is obviously the function of the neuron number after learning phase and, in consequence, also of the maximal allowable cluster diameter. The more fuzzy rules are contained in the fuzzy system, the more detailed is its knowledge about the forecasted phenomena. Such implemented fuzzy knowledge base can be the source of the new domain facts, because the facts are represented by the terms of a natural language. The use of knowledge in such representation does not require from the user any special tools and almost any effort to make use of. The extraction of

Tab. 2. Maximal allowable cluster diameter, the minuteness of collected knowledge and the duration of the learning phase

Max. allowable cluster diameter (rad)	The number of neurons after learning phase	The duration of the learning phase (ms)
$\pi/6$	2	7.187
$\pi/8$	6	10.312
$\pi/10$	35	19.219
$\pi/12$	56	24.625
$\pi/14$	67	26.375
$\pi/16$	94	34.891
$\pi/18$	99	37.672
$\pi/20$	115	42.625

Tab. 3. Costs of the fuzzy component of the hybrid

Average number of fuzzy rules	Average reaction time (ms)	Average number of operations
10	3638	2626
21	5635	4591
35	7712	7092
50	12444	9772
65	17017	12452
81	20346	15310
101	22732	18883

the knowledge from the learnt neural network is a quite difficult problem requiring computationally expensive algorithms.

Returning to the knowledge contained in the fuzzy system, there is a price to be paid for the minuteness. This price is an extreme growth on the number of operations necessary for the outcome receival. At this point the following fact should be stressed. The computations done by the neural network involve much fever operations than the ones done by an even small rule system. Concluding, the growth on computational expenses at the hybrid system is merely due to the growth of these expenses at the fuzzy system's side. This is the cost of the explicit knowledge representation.

The efficiency of the fuzzy system is obviously dependent on the volume of the system's knowledge base. We can measure it in two ways. The first method is to register the average reaction time on the presented patterns. The time is counted in miliseconds (ms). Another method for analyzing the fuzzy system efficiency is to measure the average number of operations that are to be taken to reach the final result. In Table 3 we present the registered values concerning the mentioned criterions.

Tab. 4. Costs of the neural component of the hybrid

Average number of neurons	Average reaction time (ms)	Average number of operations
4	7.46	310
12	10.06	970
42	17.08	3350
70	23.20	5460
86	25.48	6700
116	36.09	9070
129	36.51	10100
146	43.57	11420

To compare the registered values with the analogous values from the neural networks we have to translate the number of the fuzzy rules on the appropriate number of neurons. We suggest setting the quotient on 1:1. The outcomes of the neural network are contained in Table 4.

Exact tests on existence of the dependance between the maximal cluster diameter and the measured average number of operations necessary for reaching the final result have revealed the strong linear dependance.

3.2. The exactness of the forecast

In the current paragraph we try to analyse the system's work in terms of exactness of generated forecasts. To be more precise we will focus on the errors made by the systems while predicting the electrical load demand. The interesting point is the problem of application of various methods for the synthesis of the partial forecasts. Afterwards we check how the parameters of the learning process influence the system's work in the real-phase. We have to mention such parameters as:

1. The diameter of the maximal cluster that can be created by the neural network. The parameter is connected with the process of learning and extracting the knowledge from the learnt neural component.

2. The diameter of the interval in which the variations of the forecasted electrical load demand are considered allowable. The parameter is responsible for the evaluation of the generated forecast. Speaking more precisely, before creating the forecast the supervising module determines the range in which the expected value of STLF should fit. The interval is found by use of statistical tools. Then if the generated fore-

cast lays in the interval, the systems decides on the construction of the final forecast. The mentioned parameter is a maximal distance which can exists between the initial, statistically produced forecast, and the final one.

Before presenting the outcomes of the system we will explain the methods for the synthesis of the partial predictions to one final forecast. Let us mention the following strategies:

1. *Strategy 1.* "Neural network." In this strategy we ignore the results of the fuzzy systems preferring the forecasts of the neural network independently of other circumstances.

2. *Strategy 2.* "Fuzzy system." In this strategy we ignore the results of the neural network preferring the forecasts of the fuzzy component independently of other circumstances.

3. *Strategy 3.* "Better yesterday, better today." The strategy prefers the component that turned out to guess better yesterday.

4. *Strategy 4.* "Weights sum" is based on the synthesis of the partial forecasts by use of the linear function. It is easy to determine the statistically optimal weights. Starting the system with one set of test data we receive three series $(x_n)_{n \in I}$, $(y_n)_{n \in I}$, $(z_n)_{n \in I}$, containing the sequences of neural forecasts, of fuzzy forecasts and of real values for some indexing set I. Now we classically define the function of the average square error by means of the formula:

$$Err(a) := \sum_{i \in I}(z_i - ax_i - (1-a)y_i)^2, \ \ a \in [0,1].$$

The problem of finding the best possible method for weighting the partial forecasts is equivalent to finding the minimum of Err function in its domain. It is easy to verify that function Err is a polynomial of the degree two, i.e. it is of the form:

$$Err(a) = Aa^2 + Ba + C,$$

where

$$A = \sum_{i \in I}(x_i^2 - x_iy_i + y_i^2),$$

$$B = \sum_{i \in I}(2z_iy_i - 2z_ix_i - 2y_i^2 + x_iy_i),$$

$$C = \sum_{i \in I}(z_i - y_i)^2.$$

Tab. 5. The exactness of a hybrid forecast with different strategies. Error statistics. The common daily electrical load demand fluctuates between 14000 and 23000 MWh. Variable AE denotes the sum of absolute values of differences between the real and forecasted demand value, APE contains the relative values of these differences

Measure / Strategy	MAE (100 MWh)	D(AE) (100 MWh)	Max(AE) (100 MWh)	MAPE	D(APE)	Max(APE)
Strat. 1.	12.7	12.3	147.4	7.44%	7.11%	66.24%
Strat. 2.	12.71	15.17	230.2	7.31%	7.88%	100%
Strat. 3.	11.68	9.90	53.17	6.91%	6.35%	37.55%
Strat. 4.	11.85	11.17	97.60	7.03%	6.97%	55.57%
Strat. 5.	6.03	6.33	38.99	3,46%	3,82%	25,71%

Since we have $A > 0$, then function Err takes its global minimum at the point given by the formula:

$$a_{\min} = -\frac{B}{2A} = -\frac{\sum_{i\in I}(2z_iy_i - 2z_ix_i - 2y_i^2 + x_iy_i)}{2\sum_{i\in I}(x_i^2 - x_iy_i + y_i^2)}.$$

If this value does not lay in the interval $[-1, 1]$, then this means that the function reaches its minimal value on the border of the interval. Conducting tests with the data concerning the year 2002 the minimal value of Err function was taken for $a = 0.501$.

5. *Strategy 5.* "Strategy of allowable intervals." It is the algorithm used by the constructed hybrid by default. It was precisely described at the beginning of the paragraph.

For all defined strategies we make comparisons of efficiency and exactness of the hybrid outcomes. We make use of the classical measures offered by statistics.

For the estimation's of particular strategy effectiveness sake we start the hybrid with parameters set to:

1. The maximal allowable cluster diameter measured by the bound on angle ($\frac{\pi}{12}$ rad).

2. The width of the interval of allowable variations of the variable containing the energy consumption is equal to 800 MWh.

The system outcomes grouped by various strategies are contained in Table 5.

In the subsequent columns there are values of several statistics characterizing the relative and absolute error. For each type of error measure there is an average, standard deviation and the maximal value presented. The crucial

Tab. 6. Exactness of the forecasts dependant on different strategies

Strategy	The number of the best answers	Correlation coefficient systems forecast vs real value
Strategy 1.	104/365	0.80
Strategy 2.	105/365	0.74
Strategy 3.	114/365	0.85
Strategy 4.	41/365	0.83
Strategy 5.	247/365	0.95

values for the description of the error function are the ones from the first two columns.

The best results are achieved by using the strategy number five. It is the default strategy used by the system. There is a great difference between the default strategy and the other ones. This fact reveals the dominance of hybrid systems based on neural, fuzzy and statistical components over homogenous models. The algorithm of partial forecast aggregation applied in the fifth strategy is more sophisticated than the ones used in strategies 3 and 4. It effects in the growth of efficiency and exactness of the system work.

Another fact deserves mentioning. In case of the forecast generated with the strategy number two (the forecast created by the fuzzy system), in the column denoting the relative prediction error occurred value 100%. This means that the fuzzy submodule did not recognize the presented pattern at all. The lack of appropriate rules in the knowledge database could be the possible source of the problem. It is obviously the situation in case of which the system should be specially secured. It must be able then to generate the forecast in an alternative way. The hybrid architecture is already self secured in this case, because the sources of forecasts are distributed. Despite this the likelihood of occurring of the nonrecognizible pattern can remain nonzero. For that reason it is suggested to use statistics to back up the emergency cases.

As it was said in the previous paragraph the energy consumption is a cyclic function changing with a weekly period. This allows to suspect that also the function containing its changes should be cyclic with the same period. This remark makes it possible to construct the initial forecast, then to verify *a priori* the generated forecasts and reject them if they are less likely. As the example of the fifth strategy shows, the application of statistics based security mechanisms effected in great improvement of forecast's exactness.

In Table 6 there are system's outcomes presented. They are grouped by use of different strategies. We analyse the number of the best answers, i.e. the answers that turned out to be the closest to the real value of electrical

Tab. 7. The exactness of the error function. The error partition

Strategy \ the number of an absolute errors in interval (100 MWh)	[0,10)	[10,20)	[20,30)	over 30
Strategy 1.	189	104	49	22
Strategy 2.	187	102	53	22
Strategy 3.	197	99	48	20
Strategy 4.	202	89	59	14
Strategy 5.	308	43	7	6

load demand. Such statistics can be made only *ex post*. Table 6 contain also the values of the correlation coefficient of the real energy consumption value vs the system's forecasts created by use of a particular strategy. The values of these coefficients explain how the changes of the real energy consumption value influence the forecasts generated by means of a particular strategy.

As far as the number of best answers is concerned we can distinguish three groups. The first group which consist of the fourth strategy (optimal weights) achieved the worst outcome, i.e. under 20%. It shows that the application of the linear functions to the synthesis of the forecast, despite simplicity, is not always the best idea. The second group must contain the first (only a neural network), second (only a fuzzy system) and third strategy (better yesterday, better today) which exceed the level of 30%. The last group achieving the best results (the rate exceeding 65%) consists only of the default strategy.

The correlation coefficients are generally high and in all cases significantly exceed the value 0.75. It proves the existence of strong linear dependance between the real value of energy consumption and the system's forecasts independently of the used aggregating algorithm. As it was easy to foresee the best result is scored by the fifth strategy (0.95). It suggest almost ideal linear dependence between the generated predictions and the reality.

At the end we should consider the partition of the error function understood as an absolute value of the difference between the system suggestion and the real value. It is the statistic AE which denotes this type of error.

In the case of the absolute prediction error the appropriate values are contained in Table 7. Subsequent columns denote the number of days on which the absolute forecast error belonged to one of the intervals $[0, 10)$, $[10, 20)$, $[20, 30)$, $[30, +\infty)$. Again the best result is the one of the default strategy. Dramatic differences are remarkable in each of the distinguished intervals.

For all strategies we can easily observe that the number of errors denoted in subsequent intervals is smaller. The character of this change is exponential. Parameters describing the number of observations divide the set of strategies

Tab. 8. The outcomes achieved by means of different strategies

Day	Real value	Strategy 1	Strategy 2	Strategy 3	Strategy 4	Strategy 5
..	..	..	..	..	..	..
29	212	210.11	208.38	208.38	209.24	208.38
30	215	206.79	202.79	206.79	204.78	219.01
31	209	210.94	209.78	210.94	210.36	210.94
32	199	201.45	192.44	192.44	196.94	192.44
33	222	201.25	184.58	201.25	192.90	218.42
34	223	186.69	75.16	186.69	130.81	222.82
35	220	211.23	225.19	211.23	218.22	225.19
36	222	210.14	218.20	218.20	214.18	218.20
37	226	211.28	216.98	216.98	214.13	216.98
38	214	210.33	221.15	221.15	215.75	221.15
39	197	201.46	197.67	201.46	199.56	201.46
40	217	201.18	188.09	188.09	194.62	220.09
41	220	213.98	210.47	213.98	212.22	213.98
42	228	222.73	223.96	222.73	223.35	222.73
43	234	218.81	236.88	236.88	227.87	236.88
44	237	209.50	261.30	261.30	235.45	239.13
45	223	207.30	253.08	253.08	230.24	224.57
46	203	201.39	223.43	201.39	212.43	201.39
47	226	201.36	202.86	201.36	201.86	223.74
48	236	183.58	207.16	207.16	195.40	228.80
49	234	197.54	232.61	232.61	215.11	243.54
50	228	220.12	223.72	223.72	221.92	240.25
51	224	210.47	215.44	215.44	212.96	230.61
52	209	209.93	211.13	211.13	210.53	211.13
53	196	201.39	203.03	201.39	202.21	189.10
54	217	201.30	188.76	201.30	195.01	219.37
55	221	197.70	172.19	197.70	184.92	226.84
56	218	191.41	203.82	191.41	197.63	218.94
57	215	207.74	207.52	207.52	207.63	207.52
58	215	210.74	210.69	210.74	210.72	210.74
59	209	209.44	212.70	209.44	211.07	199.78
..	..	..	..	..	..	..

into two groups. The first group which consists strategies from one to four has approximately the form $a(\frac{1}{2})^{(n-1)}$, where a is an initial number and n the number of the subsequent interval. In the second group the dependance has a similar form with the only difference that it is equal to $\frac{1}{7}$. This is clearly a significant difference and points at the great quality gap between the default strategy and the other ones.

Finally let us introduce Table 8 presenting a sample of the system work groupped by usage of different strategies. The simulation of the real forecasting was conducted on the data concerning the year 2003.

3.3. The other solutions for STLF based on AI-systems

In the literature there are many examples of STLF-systems described. The main part of them are neural network and fuzzy rule based systems.

Among the neural techniques the layered networks are quite popular. Some examples are to be found in [4] and [10]. In the described systems the similar data model was exploited. The most important information to flow to a network concerned the history of electrical load demand on the previous

day, the air temperature and humidity, the type of weekday, for which the forecast was created. In course of experiments various features of the system were scaled and optimized, in particular network topology, the quantity of neurons in subsequent layers, the number of hidden layers, etc.

The system described in [10] was based on a network consisting of four layers with 8,15,10,1 neurons, respectively. The quality of prediction was estimated by means of MAPE measure. The test outcomes obtained while proceeding electrical demand data for years 1993–1994 fluctuated in the range of 1.29% and 1.32%.

An analogous solution described in [4] took advantage of the 3-layer network, containing the 34,6,1-neuron layers. The average prediction error estimated by the same measure oscillated around 2%.

Both examples scored better results than the neural component of the hybrid system. However there must be some points emphasized. First of all, both solutions described above are based on layered neural networks, whereas the hybrid system contains the one-layer(Kohonen-type) network. These two various network types work in completely different way. The Kohonen network match patterns in opposite to layered perceptrons that suit to modelling functions difficult to compute.

On the grounds of cited results we can state that layered networks cope better with STLF – problems. This fact may suggest concerning the use of a layered network as a part of the hybrid. However we must be conscious of several problems that are connected with this decision. The growth of exactness is paid for by the greater complication of network topology. As a result of that, the cost of the learning phase is greater. Finally the network knowledge extraction becomes more difficult and expensive.

To summarize, the use of perceptron networks for building the neural components of the hybrid should be the subject of another research. Such investigation should embrace not only the exactness and effectiveness of the network, but also such aspects as the cost of the learning process and the difficulty of knowledge extraction.

In [4] there are also several examples of using of fuzzy systems in application to STLF. In particular the fuzzy rule systems are analyzed. The knowledge base in the form of the set of fuzzy rules was created in the process of data classification and clusterization. (This can be done for instance by means of neural networks). For calibrating the created fuzzy systems, especially fuzzy sets, the error back-propagation method was used. The reached result of exactness is worth mentioning, i.e. MAPE = 1.99%. This fact suggest reconsidering the use of methods for calibrating the parameters of fuzzy systems. Final remark when estimating the outcomes of the fuzzy systems is that one must always consider the performance factor. As it was said previously the main part of the cost of the hybrid system's prediction is at the

fuzzy component's side. When comparing the outcomes that originated from two different fuzzy systems, one must make sure that both outcomes were achieved by means of similar computational resources.

4. References

[1] Tadeusiewicz R., Flasiński M.; *Rozpoznawanie obrazów*, PWN, Warszawa 1991.

[2] Tadeusiewicz R.; *Sieci neuronowe*, Akademicka Oficyna Wydawnicza, Warszawa 1993.

[3] Zieliński J.; *Inteligentne systemy w zarządzaniu. Teoria i praktyka*, PWN, Warszawa 2000.

[4] Zieliński J.; *Sztuczne systemy w zarządzaniu. Teoria i praktyka*, PWN, Warszawa 2002.

[5] Zieliński J.; *Survey of short-term electrical load forecasting methods*, in: *Materiały Konferencji APE'97, Aktualne Problemy w Elektroenergetyce, Gdańsk, Jurata, 11–13 czerwca 1997,* t. IV, 1997, pp. 121–129.

[6] Hippert S.H., Pedriera C.E.; *Souza RC Neural networks for short-term load forecasting: a review and evaluation*, IEEE Trans. Power Systems, 16 (1), 2001, pp. 44–55.

[7] Bartkiewicz W., Pamuła A.; *Krótkoterminowe prognozowanie obciążenia sieci elektroenergetycznej z wykorzystaniem adaptacyjnych systemów rozmytych*, in: *Materiały V Konferencji ,,Sieci i Systemy Informatyczne – Teoria, Projekty, Wdrożenia,"* Łódź 1997, pp. 63–74.

[8] Bartkiewicz W., Zieliński J.; *Zastosowanie narzędzi sztucznej inteligencji do prognozowania zapotrzebowania na energię elektryczną*, in: *Materiały Konferencji ,,Komputerowo Zintegrowane Zarządzanie,"* Zakopane 1998, pp. 27–34.

[9] Malko J.; *Wybrane zagadnienia prognozowania w elektroenergetyce*, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 1995.

[10] Osowski S.; *Sieci neuronowe w ujęciu algorytmicznym*, WNT, Warszawa 1996.

[11] Osowski S., Kądzielawa A.; *Short term load forecasting for the electric power using neural networks*, KKTPiUE, Zakopane 1995.