Wind Power Forecasting Using Advanced Neural Networks Models

Georges Kariniotakis, Georges Stavrakakis, Eric Nogaret

Paper submitted to IEEE Transactions on Energy Conversion, 1996.

WIND POWER FORECASTING USING ADVANCED NEURAL NETWORKS MODELS.

G. N. Kariniotakis
Ecole des Mines de Paris,
Centre d'Energétique,
06904 Sophia-Antipolis, France.

G. S. Stavrakakis
Technical University of Crete,
Dept. of Electronic & Computer Eng.,
73100 Chania, Crete, Greece.

E. F. Nogaret
Ecole des Mines de Paris,
Centre d'Energétique,
06904 Sophia-Antipolis, France.

***Abstract* - In this paper, an advanced model, based on recurrent high order neural networks, is developed for the prediction of the power output profile of a wind park. This model outperforms simple methods like persistence, as well as classical methods in the literature. The architecture of a forecasting model is optimised automatically by a new algorithm, that substitutes the usually applied trial-and-error method. Finally, the on-line implementation of the developed model into an advanced control system for the optimal operation and management of a real autonomous wind-diesel power system, is presented.**

Keywords: *Short-term wind power forecasting, recurrent neural networks, wind-diesel power systems.*

I. INTRODUCTION

Wind Energy Conversion Systems (WECS) appear as an attractive alternative for electricity generation, especially when integrated to isolated power systems, like the systems of islands or of rural areas. Although the integration of WECS results in important environmental and economic benefits [2, 4, 14], the stochastic nature of the wind, imposes considerable difficulties on the optimal management of these power systems [7, 13].

The integration of efficient wind power forecasts in the power system control and management functions may result in a reduction of operating costs and in an improvement of the quality of service [1, 3, 4, 7, 14].

In the present paper, a recurrent high-order neural network model is developed for the short-term prediction of wind power. A main advantage of this type of neural networks over conventional ones, is that fast learning algorithms can be derived for the weights estimation, enabling it to be appropriate for on-line applications [9].

The developed model can be used for the prediction of wind speed or power in time-scales that can vary between some seconds to some hours. However, this paper is mainly concerned with forecasts useful for the short-term scheduling of an autonomous power system that is, with forecasts of the WECS power output profile for the next 2 or 3 hours with a time step in the order of 10 minutes.

The paper is structured as following : initially, the state-of-the-art on the short-term wind forecasting problem is given. Then, a wind power forecasting model, is developed and evaluated using data from the wind-diesel power system of the Greek island of Lemnos. An algorithm for the optimisation of the architecture of a forecasting model is then presented. This algorithm optimises all the parameters that are critical for the *generalisation* capability of the model that is, its ability to predict data other than those on which it has been trained.

Finally, the implementation of the developed model into an advanced control system for the optimal operation and management of the wind-diesel power system of the island of Lemnos, is presented.

II. THE WIND POWER FORECASTING PROBLEM

The main differences between the various wind forecasting models found in the literature are in :

(i) *the time-scale*. It can be in the order of a few seconds when forecasts of wind speed, rather than power, are used for wind turbines control [2, 3, 5]. Alternatively, time-scale can be in the order of several minutes or even hours, when the objective is economic dispatch and power system planning. Such functions necessitate forecasts of the WECS power output profile [1, 4, 7, 13, 14].

(ii) *the methodology*. In smaller time-scales, the time-series methodology is applied, according to which, past values of a process are used to predict future values [2-7]. In higher time-scales of some hours or more, models based on meteorological information are usually used. These models produce forecasts a limited number of times per day and necessitate more parameters and considerable computer time compared to timeseries models [1].

Since in this paper, short-term wind forecasting is considered, the timeseries approach will be followed. Results from the quite limited existing literature on this approach are presented below, indicating the level of wind predictability. The evaluation of the methods is based on comparison with the persistent method described in Section III.A.

In [2], a Kalman filter technique is used for wind speed prediction in several time-scales. The improvement with respect to the persistent, on 1-minute data, is in the order of 4 to 10 % relative to the root mean square (RMS) criterion of errors. The performance of the method becomes very small or negative when 10-minute averaged data are used.

In [3], ARMA models achieve an improvement w.r.t. the persistent in the order of 5-12 % on 2-second data (horizon up to 20 seconds), and 9-14 % on 1-minute data (horizons up to 10 minutes). The performance of multilayer feed-forward neural networks for the prediction of wind power is found to be very close to the persistent when 10-second data are used [5], while the improvement for one-step prediction of 1-minute data is 11% and for 10-min data is 8% [6]. A similar performance on 1- and 10-minute data is also obtained by Radial Basis Functions [6].

In [7], various models like ARMA and bilinear ones are developed for wind power forecasting. They achieve an improvement of 7 to 12 % with respect to the persistent, for a forecast horizon of 2 hours and time-steps of 30 minutes.

Finally, statistically evaluated results on multi-step ahead forecasting of timeseries with a resolution in the order of ten minutes are not found in the literature.

III. WIND POWER FORECASTING METHODOLOGY

Two main approaches may be followed in order to generate wind power forecasts :

(i) to develop an explicit prediction model for wind power, in which it will be possible to consider wind speed, wind direction etc. as explanatory variables.

(ii) to develop a prediction model for wind speed and a second model for the transformation of wind speed to power. As transformation model can be used the manufacturer's characteristic curve of the wind turbine (WT) power output as a function of wind speed. If however, the point of measurement of wind speed is different than the hub height of a wind turbine, then the transformation model should account for the spatial variations of wind.

In the case-study of Section IV, where the wind park power output profile is predicted and the wind speed measurements are made by a unique anemometer - as is usually the case in applications, the first approach gave better results. The consideration of the WT characteristic curve as a transformation model gave poor results, while an advanced transformation model did not give any advantages over the first approach. Similar results are also reported in [7].

A. The Naive Predictors

When an advanced forecasting model is not available, forecasts may be obtained with a minimal effort and data manipulation and can be based solely on the most recent information available. Such forecasts are referred to as *Naive* forecasts. One such method (*Persistent* or *Naive*_1 method), is to use the most recent datum available $P(t)$, as forecast $\hat{P}(t+k/t)$ for each one of the future time-steps that is : $\hat{P}(t+k/t) \hat{=} P(t)$ for k=1,...,n. A slightly more sophisticated method would be to use the average of m past values (*Naive_m*) as forecast :

$$\hat{P}_m(t+k/t) = \frac{1}{m}\sum_{i=0}^{m-1} P(t-i)\,,\; k=1,...,n. \tag{1}$$

The forecasting errors over a set of data, are used to calculate various performance criteria like the Root Mean Square (RMS) of errors both for Naive and advanced methods. The benefit gained by the use of an advanced method is typically measured as a *percentage improvement* on a certain criterion (e.g. RMS) [2-7].

B. The Advanced Neural Network Model

Recently, increasing interest has been shown for the use of Recurrent Neural Networks, for modelling and identification of dynamic systems. These networks dispose dynamic elements in the form of feedback connections. This distinguishes them from feedforward neural networks, where the output of one neuron is connected only to neurons in the next layer. In the simple case, the state history of each neuron is determined by a difference equation of the form :

$$x_i(t+1) = a_i x_i(t) + b_i \sum_j w_{ij}(t) y_j(t) \tag{2}$$

where x_i is the state of the *i*-th neuron, a_i, b_i are constants, w_{ij} is the synaptic weight connecting the *j*-th input to the *i*-th neuron, and y_j is the *j*-th input to the above neuron. Each y_j is either an external input or the state of a neuron passed through a sigmoidal function, i.e., $y_j = S(x_j)$ where $S(.)$ is a sigmoidal nonlinearity. The dynamic behaviour and stability properties of models described by (2) have been studied by various researchers like Grossberg [8].

High-order networks are expansions of the first-order Hopfield and Cohen-Grossberg [8] models that allow higher-order interactions between neurons. In a recurrent *second-order* neural network the total input to the neuron is not only a linear combination of the components y_j, but also of their products $y_j y_k$. Moreover, one can pursue along this line and include higher-order interactions represented by triplets $y_j y_k y_l$, quadruplets, etc. This class of neural networks form a *recurrent higher-order neural network* (RHONN).

Consider now a discreet time RHONN consisting of n neurons and m inputs. The state x_i of the *i*-th neuron is governed by a difference equation of the form [9] :

$$x_i(t+1) = a_i x_i(t) + b_i \sum_{k=1}^{L} w_{ik}(t) z_k(t) \tag{3}$$

where z_k is defined as : $z_k \hat{=} \prod_{j \in I_k} y_j^{d_j(k)}$, $\{I_1, I_2, \ldots, I_L\}$ is a collection of L not-ordered subsets of $\{1,2,\ldots,m+n\}$, and $d_j(k)$ are non-negative integers. The vector $\boldsymbol{y}$ with the inputs to each neuron is defined by :

$$\begin{aligned} \boldsymbol{y} = \left[y_1, \ldots, y_n, y_{n+1}, \ldots, y_{m+n}\right]^T \hat{=} \\ \hat{=} \left[S(x_1), \ldots, S(x_n), S(u_1), \ldots, S(u_m)\right]^T \end{aligned} \tag{4}$$

where $\boldsymbol{u}=\left[u_1,u_2,\ldots,u_m\right]^T$ is the external input vector to the network and ^{T} denotes the transpose operator. The function $S(.)$ is a monotone increasing, differentiable sigmoidal function of the form :

$$S(x)=\alpha\frac{1}{1+e^{-\beta x}}-\gamma \tag{5}$$

where α, β are positive real numbers and γ is a real number.

The stochastic gradient method is used to derive the following learning rules for the weights estimation :

$$w_{ik}(t+1)=w_{ik}(t)+\eta_i z_k(t)e_i(t) \tag{6}$$

where $e_i(t)\hat{=}x_i(t)-\chi_i(t)$ denotes the prediction error, $\chi_i(t)$ is the measured value of the process and η_i is a small positive parameter denoting the *learning rate*. The value of the learning rate is reduced during the learning process in order to avoid residual fluctuations or instability. The reduction is made according to a *learning rate schedule function* of the "*search-then-converge*" [10] type :

$$\eta(t)=\frac{\eta_0}{(1+\rho)^{k-1}} \tag{7}$$

where η_0 is the initial value of the learning rate, which is reduced by the small positive quantity ρ at each learning epoch *k*. The total number of learning epochs is denoted as *K*. Since the RHONN model is linear in the parameters (weights) the learning procedure leads to globally optimal values of the weights.

Consider now that there is a dynamic process, whose input/output relation is given by a general difference equation of the form :

$$\chi(t+1)=f\left(\chi(t),\boldsymbol{u}(t)\right) \tag{8}$$

where $\chi\in\Re^n$ is the scalar process output, $\boldsymbol{u}\in\Re^m$ is the scalar input of the process and $f(.)$ is an unknown function. It has been shown in [9] that if (i) the function $f(.)$ satisfies some very mild continuity assumptions, (ii) the vector $\boldsymbol{u}(t)$ is available for measurable, (iii) the number L of high-order connections is sufficiently large, and (iv) an appropriate adaptive (learning) algorithm for adjusting the weights of the RHONN is available, then the RHONN model is possible to approximate the dynamic system to any degree of accuracy.

Considering the problem of the wind power prediction, it is assumed that the dynamic process χ, governed by (8), is the *future* wind power profile. The objective is then, to find a neural network (NN) function $f_M(.)$ that can approximate $f(.)$. There are 3 main ways to represent this future profile or, in other words, to consider multi-step predictions [11] :

(i) The first one is to consider that the process can be approximated by a unique NN with *n* outputs, each one giving a prediction for a different time-step. Then, the process vector $\chi(t+1)$ and the RHONN state vector $\boldsymbol{x}(t+1)$ are considered to be :

$$\chi(t+1)=\left[P(t+1),P(t+2),\ldots P(t+n)\right]^T$$

$$\boldsymbol{x}(t+1)=\left[x_1,x_2,\ldots x_n\right]^T\hat{=}\left[\hat{P}(t+1/t),\hat{P}(t+2/t),\ldots\hat{P}(t+n/t)\right]^T$$

(ii) Alternatively, it is possible to consider one NN with one output only for the first time-step. In this case the predictions are fed-back as input to the same NN model to give the output for the following time-steps.

(iii) Finally, one can consider that the real process can be approximated by *n* different NNs, each one with one output corresponding to a different time-step.

The process input $\boldsymbol{u}(t)$ at time *t* is determined as :

$$\boldsymbol{u}(t)\hat{=}[U_1\left(t-p_{1,1}\right),\ldots,U_1\left(t-p_{1,r_1}\right),U_2\left(t-p_{2,1}\right),\ldots,U_2\left(t-p_{2,r_2}\right),\ldots,$$
$$\ldots,U_s\left(t-p_{s,1}\right),\ldots,U_s\left(t-p_{s,r_s}\right)]^T \tag{9}$$

where $p_{i,r_i}\in J_i$, $J_i\subseteq\aleph$ with $i=1,2,\ldots,s$ and *s* is the number of different kinds of input data (e.g. wind power, speed, direction etc.). The total number of input variables in the model is $m=\sum_{i=1}^{s}r_i$ An example of $\boldsymbol{u}(t)$ can be :

$$\boldsymbol{u}(t)=\left[U_1(t),U_1(t-1),U_1(t-3),U_2(t),U_2(t-4),U_2(t-9)\right]^T$$

with U_1 representing wind power, and U_2 wind speed.

C. Optimisation of the forecasting model architecture

A major problem in time-series forecasting is the determination of the optimal architecture of a model. Usually, the *trial-and-error* method is applied to test various alternative model architectures and choose the one with the optimal generalisation capability.

As *generalisation,* is defined the capability of a forecasting model to predict data other than those on which it has been trained. A model with too many free parameters will fit the training data arbitrarily closely, but will not necessarily lead to optimal generalisation.

Two classes of generalisation criteria are usually used for model architecture selection [10, 15]. The first class contains criteria formed by a first term measuring the goodness of fit and a second term penalising the number of parameters in the model (e.g. Akaike Information Criterion [15]). The second class of criteria is based on the principle of cross-validation, according to which, the decisions on the model structure are made on a sample of data different than the sample used to estimate the parameters of the model.

In this Section, an algorithm for the optimisation of the architecture of a RHONN forecasting model, is presented. The optimisation criterion is based on the cross-validation approach, which has been extended to consider that the generalisation capability of a model can be ameliorated if training is stopped, before the model starts fitting the noise in the data [10]. The parameters that are optimised by the algorithm are the parameters with an influence on the generalisation capability of a model, that is :

(i) The *number of past values* of each kind of data that will be used as input in the model. A preliminary analysis of the available wind data showed that not important seasonallities are present. Hence, the set J_i of orders for the *i*-th type of data, is considered to contain consecutive values, that is $J_i = \left\{0,1,2,\ldots,p_{i,r_i}\right\}$. The algorithm optimises the number r_i of elements in J_i which is : $r_i = p_{i,r_i} + 1$. This optimisation leads automatically to a selection of data since in the case that $r_i^* = 0$ then, the *i*-th type of data will not be considered as input to the model.

(ii) The parameters η_0, r, K of the *learning rate schedule* given by (7). The optimisation of the number of training epochs K leads to an optimal terminating point for the learning process.

(iii) Finally, the parameter β, that determines the shape of the sigmoidal function (5) used in the neural network, is optionally optimised for each kind of input. It is noted that the parameters α, γ in (5) are fixed to some constant values according to the range of possible values of the input *x* of the sigmoidal function.

The vector of the candidate model architecture parameters is defined as :

$$\boldsymbol{q} \mathrel{\hat{=}} \left[r_1, r_2, \ldots, r_s, \eta_0, \rho, K, \beta_1, \ldots, \beta_s\right]^T , \tag{10}$$

where the indices associated to β indicate that each kind of data is passed through a different sigmoidal function.

According to the cross-validation approach, the available data set $S \mathrel{\hat{=}} \left\{P(1),\ldots,P(T)\right\}$ is randomly partitioned into three subsets :

- The *learning set* $S_L \mathrel{\hat{=}} \left\{P(1),\ldots,P(T_L)\right\}$, which is used for the estimation of the weights using (6).
- The *validation set* $S_V \mathrel{\hat{=}} \left\{P(T_L+1),\ldots,P(T_V)\right\}$, which is used to take decisions on the architecture parameters.
- Finally, *the test set* $S_T \mathrel{\hat{=}} \left\{P(T_V+1),\ldots,P(T)\right\}$, which is neither used in the weights estimation, nor in the architecture optimisation, but only for the ultimate evaluation of the model. The performance on this set of data is compared with the performance of the Naive methods, or other advanced methods in the literature.

Following the above considerations, the architecture optimisation problem is formulated as :

$$\boldsymbol{q}^*: \quad \min_{\boldsymbol{q}} \; J_V(\boldsymbol{q}) = \frac{1}{2}\sum_{k=1}^{n} \sum_{\substack{t+k: \\ P(t+k)\in S_V}} \left(P(t+k) - \hat{P}(t+k/t)\right)^2$$

under the constraints :

1) $\boldsymbol{q}_{min} \le \boldsymbol{q} \le \boldsymbol{q}_{max}$

2) $\hat{P} = f_M\left(\hat{P}, \boldsymbol{u}; \boldsymbol{q}, \boldsymbol{w}^*\right)$

3) $\boldsymbol{w}^*: \quad \min_{\boldsymbol{w}} \; J_L(\boldsymbol{w}/\boldsymbol{q}) = \frac{1}{2}\left(P(t+k) - \hat{P}(t+k/t)\right)^2, \quad \forall\, P(t+k) \in S_L, \;\; k = 1,\ldots n$

where $J_V(\boldsymbol{q})$ is the sum of square prediction errors on the validation set S_V. Its minimisation gives the optimal value of the model architecture parameters $\boldsymbol{q}^*$.

The 1st constraint stands for the range of acceptable values for the architecture parameters.

The 2nd constraint denotes that the predictions in the validation set are made using the neural network model described by (3), with an architecture determined by $\boldsymbol{q}$ and with optimal values of weights w^*.

Finally, the 3rd constraint denotes that the optimal values of the weights of a candidate model, are obtained by training the model on the data of the learning set S_L by using the learning rules (6) and according to the schedule defined by the parameters η_0, ρ, K inside $\boldsymbol{q}$.

The above optimisation problem is solved using the non-linear Simplex method of Box [12]. This method is based on the evaluation of the surface of the function $J_V(\boldsymbol{q})$. The method has been adapted to optimise both discrete (e.g. number of past data, number of epochs, etc.) and continuous variables (e.g. learning rate).

IV. Experimental Results & Discussion

The case-study of the wind-diesel power system of the island of Lemnos is considered. The installed diesel capacity is 13.25 MW, while 2 wind parks of a total capacity of 1.14 MW are installed [8 WTs of 55 kW, 7 WTs of 100 kW]. The short-time scheduling of the above power system is performed by an advanced control system described below, and necessitates forecasts of the wind power profile of each wind park for an *horizon of 2 hours* with a *time-step of 10 minutes* [13].

The wind park power profile is defined as the *average wind turbines power* of the park that is, as the ratio of the total wind park power at time *t* divided by the number of WTs in operation at *t*. Forecasts of the total wind park power would not be easily usable by the scheduling function, since this quantity does not contain any information on the number of WTs in operation. In addition, in the timeseries of the total wind park power, discontinuities due to switching operations of WTs, provoked by external, and hence unpredictable events (e.g. dispatching actions), are present.

The measurements presented here have been obtained by the data acquisition system of the island and cover a period of five days with a time-step of 1 minute. These are measurements of the wind speed at each park (anemometer location), as well as of the power output of each WT. The 1-minute wind speed data and the values of the average WTs power were averaged in 10-minutes intervals giving the timeseries of Fig. 1. These timeseries have been divided into a learning set containing the initial 360 data, a validation set with 100 data, and a test set with the last 190 data. Below, the results concerning only the second wind park (7x100kW), are presented.

Three basic artificial neural network (ANN) configurations have been tested : (i) a network with 12 outputs, one for each time-step, denoted as ANN-a. (ii) A network with one output, which is used iteratively to give forecasts for all the 12 time-steps, denoted as ANN-b. Its

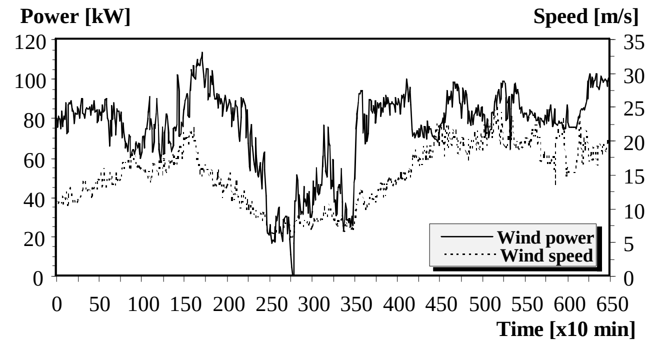

Fig. 1 : Average wind turbines power and wind speed data from the island of Lemnos (Febr. 10-14, 1994).

architecture has been optimised by considering in $J_V(\boldsymbol{q})$ the forecast errors of all the time-steps. (iii) A network similar to the previous one, denoted as ANN-c, which has been optimised by considering in $J_V(\boldsymbol{q})$ only the errors of the first time-step.

High-order terms up to two were considered in all cases, while the neurons are fully interconnected via recurrent links. The vectors of the optimal architecture parameters were found to be :

	r_1	r_2	η_0	ρ	K	β_1	β_2
ANN-a	2	1	0.02	0.11	61	3.28	2.72
ANN-b	8	3	0.09	0.17	9	3.99	0.77
ANN-c	6	5	0.01	0.05	23	3.66	0.76

where r_1 are the number of past wind power and r_2 the number of past wind speed values.

The improvement on the RMS criterion gained by the ANN models over Persistent is given in Fig. 2. From the use of the ANNs there is a clear gain concerning multi-step forecasts. ANN-a and ANN-b have a better performance than ANN-c on forecasts of longer time-steps. This is because in the architecture optimisation of these networks their performance in the whole horizon has been considered. Concerning one step-ahead forecasts, the best performance is achieved by ANN-c, which has been exclusively built for such forecasts. Although better results can be expected if a different model is built for each time-step, this approach is undesirable from the on-line implementation point of view.

Finally, it is remarked that the length of the considered

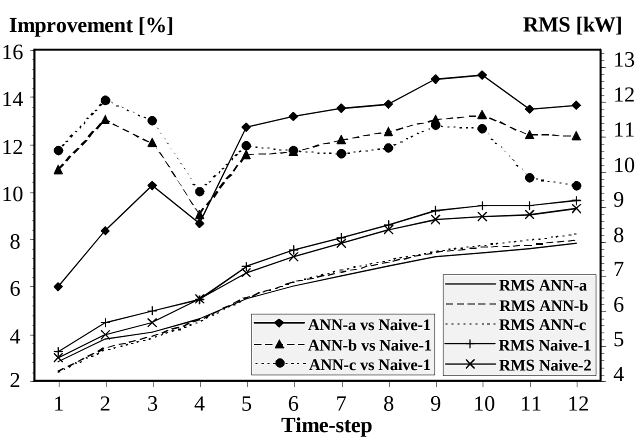

Fig. 2 : RMS error of various methods and improvement of ANN method over simple methods.

timeseries is certainly small for a complete statistical evaluation of the developed model. However, this case-study, has been selected for presentation here due to the on-line implementation of the developed model as described in the next Paragraph. In addition, from tests with longer timeseries from other sites, it was concluded that the above results can be considered as representative for the level of performance that can be obtained by the proposed method.

Implementation in the existing Control System

An advanced control system (CS) for the optimal operation and management of wind-diesel power systems has been developed in the frame of the EU project JOU2-CT92-0053 [13]. The control system has been installed and is under evaluation in the wind-diesel power system of the island of Lemnos.

The CS is aimed to assist the power system operators by proposing them optimal scenarios for the power system operation, so that maximum fuel saving is achieved, without deterioration of the quality of service to the consumers. The scenarios are generated by an economic dispatch module by taking into account load and wind power forecasts - see Fig. 3. The security of the power system is guaranteed by a security assessment module, which supervises the generation of the operation scenarios and rejects those that might lead to dynamically unsafe situations for the power system.

Wind power forecasts are provided by a wind power forecasting (WPF) module, whose detailed operation within the control system is shown in Fig. 4. Various Naive predictors, as well as advanced models, have been integrated in the WPF module. The ANN-a RHONN model was chosen to be implemented due to its globally better performance compared to ANN-b and ANN-c. The role of Naive predictors is to replace advanced methods each time that the CS starts to operate. Then, a transient period is necessary for the auto-adaptation of the parameters of the

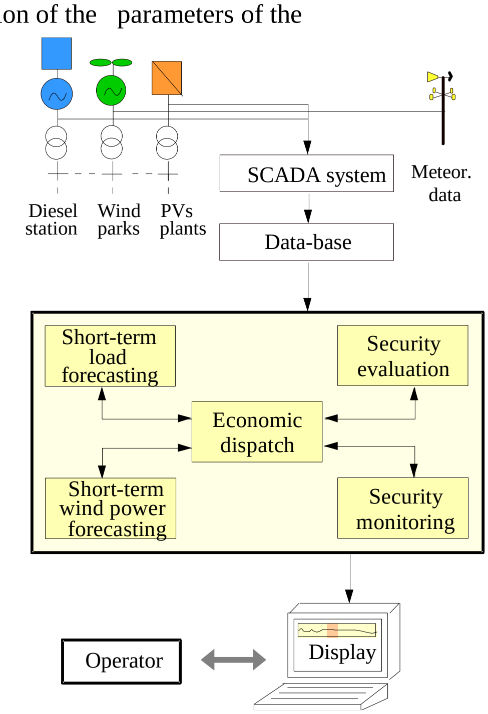

Fig. 3 : The architecture of the pilot control system

advanced models to the new conditions. An advanced model starts to provide forecasts, when its adaptation leads to a performance superior to that of simple methods.

The on-line operation of the power system, during the period of five days considered here, is analysed in [14]. The utilisation of advanced forecasts by the control system, during this period, results (i) in a decrease of the total number of start/stops of the diesel units, which may arise up to 30 %; (ii) in a decrease of the loss of load events; (iii) in an increase of fuel savings restricted to 1-2 % for this limited period; (iv) and finally in a higher utilisation of the available wind energy.

V. CONCLUSIONS

In the present paper, an advanced neural network based model for wind power timeseries forecasting has been developed. The performance of the model on forecasts appropriate for the short-term scheduling functions of an autonomous power system was examined. The model outperforms Naive methods, while the obtained results are superior to similar results of the known alternatives. This is due to advantages coming both from the adaptive neural network approach, as well as from the algorithm proposed here to optimise the architecture of the forecasting model. This algorithm, which replaces the trial-and-error method, is aimed to maximise the generalisation capability of a forecasting model. It is of general value, since it is applicable to any timeseries analysis non-linear problem.

Finally, the developed model has been implemented for on-line use in the pilot control system for the wind-diesel power system of the Greek island of Lemnos.

VI. ACKNOWLEDGEMENT

Authors would like to thank the EU for the funding of the Joule-II project : JOU2-CT92-0053 part of which is the present work.

VII. REFERENCES

[1] L. Landberg et al, "Short-term prediction of local wind predictions", *Procd. of the 4th European Wind Energy Conference*, EWEC'93, Lübeck-Travemünde, Germany, 1993, pp. 677-680.

[2] E.A. Bossanyi, "Short-term wind prediction using Kalman filters", *Wind Engineering*, vol. 9, No. 1, 1985.

[3] E.A. Bossanyi, "Stochastic wind prediction for wind turbine system control", *Procd. of the 7th BWEA Wind Energy Conference*, 27-29 March 1985, pp. 219-226.

[4] G.C. Contaxis, J. Kabouris, "Short-term scheduling in a wind/diesel autonomous system", *IEEE Trans. on Power Systems*, 91 WM 188-3 PWRS.

[5] J.O.G. Tande, L. Landberg, "A 10 sec. forecast of wind turbine output with neural networks", *Procd. of the 4th European Wind Energy Conf.*, EWEC'93, Lübeck-Travemünde, Germany, 1993, pp. 774-777.

[6] H.G. Beyer et al, "Short-term prediction of wind-speed and power output of a wind turbine with neural networks", *Procd. of the 5th European Wind Energy Conference*, EWEC'94, Thessaloniki, Greece, 1994, pp.349-352.

[7] ELSAM, *Wind power prediction tool in central dispatch centres*, Progress Report 3 of the EU project JOU2-CT92-0083, October 1993.

[8] M. A. Cohen, S. Grossberg, "Absolute stability of global pattern formation and parallel memory storage by competitive neural networks", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-13, pp. 815-826, 1983.

[9] E. Kosmatopoulos et al, "Identification of nonlinear systems using new dynamic neural network structures", *Procd. of the 31st IEEE Conf. on Decision and Control*, Arizona, USA, 1992, vol. 1, pp. 20-25.

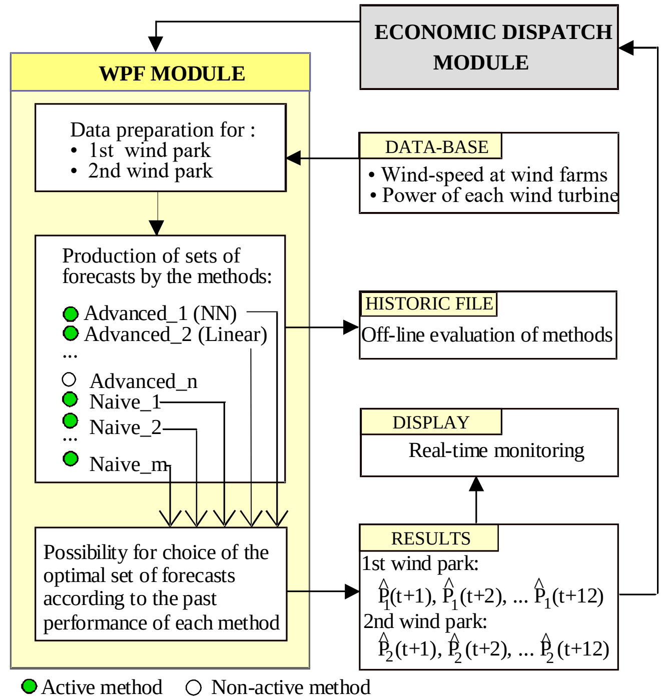

Fig. 4 : Flow-chart of the WPF module operation.

[10] S. Haykin, *Neural networks - A comprehensive foundation*, Macmillan College Publ. Company, 1994.

[11] J.L. Lin, C.W.J. Granger, Forecasting from nonlinear models in practice", *Journal of Forecasting*, vol. 13, 1994, pp. 1-9.

[12] S.S. Rao, *Optimisation theory and applications*, 2nd Edition., Willey Eastern Ltd., 1984.

[13] E. Nogaret et al, "A new expert system based control tool for power plants with large integration of PVs & Wind power plants", *Procd. of the 1st World Conference on Photovoltaic Energy Conversion*, December 5-9, 1994, Hawai, USA, pp. 1052-1055.

[14] A.I. Androutsos, M. P. Papadopoulos, A.V. Machias, "A logistic Model for on-line application in diesel-wind power systems", *Procd. of the 5th European Wind Energy Conference*, EWEC'94, Thessaloniki, Greece, 1994, vol. 2, pp. 1068-1073.

[15] J.G. Gooijer et al, "Methods for determining the order of an autoregressive-moving average process: A survey", *International Statistical Review*, (1985), **53**, 3, pp. 301-329.

VIII. BIOGRAPHIES

G. N. Kariniotakis was born in Athens, Greece, 1967. He received the Production & Management Eng. degree (1990) and the McS degree on Electronic & Computer Eng. (1992) from the Technical University of Crete, Greece. He is currently studying towards the PhD degree at the Paris School of Mines, France. His research interests include renewable energies and artificial intelligence. He is a member of the Technical Chamber of Greece and a Student Member of the IEEE Power Eng., Automatic Control and Neural Networks Societies.

G. S. Stavrakakis was born in Crete, Greece in 1956. He received his Electrical Eng. Degree from the National Technical University of Athens, Greece in 1980 and his DEA and PhD Degree on Automatic Control from the University "Paul Sabatier", Toulouse, France in 1984. He was with the Computer Science Dept. of the National Technical University of Athens from 1985 to 1989 as Research Fellow, and with the Joint Research Center of EU from 1988 to 1989. He is currently Professor at the Technical University of Crete, Greece.

E. F. Nogaret was born in Paris, France in 1958. He received the Engineering and the Doctorate degree from the Paris School of Mines, France, in 1981 and 1983 respectively. He is currently Research Manager in the Centre d'Energétique of the Paris School of