A new Integrated Approach for Very Short Term Wind Speed Prediction Using Wavelet Networks and PSO

E. Safavieh, A. Jahanbani Ardakani, A. Kashefi Kaviani, S. A. Pourmousavi, S. H. Hosseinian, M. Abedi

Abstract-- **Very short term wind speed forecasting is necessary for wind turbine control system. In this study, a new integrated approach using Wavelet-based networks and PSO is proposed for very short term wind speed forecasting. PSO algorithm is used for training a Wavelet networks and the whole integrated approach is applied for wind speed prediction. As a case study, the wind speed data from a site in Denmark with 2.5 s measured resolution is used for training and test of the network. Proposed approach is compared to multi layer perceptron networks with Back Propagation training algorithm. Results show that the new approach improve Mean Absolute Percentage Error (MAPE) and Maximum error of prediction.**

Index Terms— **Mean absolute percentage error, Particle swarm optimization (PSO), Very short term, Turbine application, Wavenet.**

I. Introduction

VARIATION of wind speed are in the range of seconds, minutes, hours, days, weeks, seasons and years. In long-term planning for wind turbine installations, the wind speed prediction should be for years ahead [1]. However, control system of a wind turbine, requires prediction times in the range of seconds ahead of our interest, because the major problem with the control of wind turbines is the delays associated with the wind turbine system. These delays affect the response of the system in respect to controller action [1]. Very short term wind speed prediction can be used for dynamic control of a wind turbine, due to importance of short-term decisions. The short term decisions could be classified as connection of a load, changing the pitch of the blades and/or any other control action which involves delays. Lots of works have been done for wind speed prediction that most of them are about prediction in bigger ranges. Alexiadis et al. [2] propose an artificial neural network model for forecasting average values of the following 10 min or 1 hour wind speed and the related electrical power of wind turbines. This work was focused on the operation of power systems with integrated wind parks. Hoppmann et al. [3] develop a system for short-term prediction of the maximum wind speed on the basis of continuous wind measurements at distinct locations of a high-speed railway line. Kariniotakis et al. [4] introduce a new methodology for assessing on-line the prediction risk of short-term wind power forecasts. This methodology is characterized by two different concepts, the uncertainty and prediction risk estimation. Potter et al. [5] describe a novel approach for very short-term wind speed prediction using adaptive neuro-fuzzy inference system. Their approach verified with a case study from Tasmania, Australia. Negnevitsky et al. [6] use an adaptive neuro-fuzzy inference system (ANFIS) to forecasting wind time series. Oztopal [7] develop an Artificial Neural Network, tuning weighting factors of surrounding stations necessary for the prediction of a pivot station. Geometrical weighting functions are necessary for regional estimation of the regional variable at a location with no measurement, which is referred to as the pivot station from the measurements of a set of surrounding stations. Cadenas et al. [8] compare two techniques for wind speed forecasting in the South Coast of the state of Oaxaca, Mexico. These two methods were Autoregressive Integrated Moving Average (ARIMA) and the Artificial Neural Networks (ANN). Sahin et al. [9] generate hourly wind speed data in northwestern region of using transition matrix approach of the first-order Markov chain method. For this purpose, the wind speed time series is divided into various states depending on the arithmetic average and the standard deviation. Shamshad et al. [10] use the transition matrix approach of the Markov chain process to generate Hourly wind speed time series data of two meteorological stations in Malaysia for stochastic generation of wind speed data. In [10] The transition probability matrices have been formed using two different approaches: the first approach involves the use of the first order transition probability matrix of a Markov chain, and the second involves the use of a second order transition probability matrix that uses the current and preceding values to describe the next wind speed value. Riahy

This work was supported in part by Center of Excellence for Power Eng, Amirkabir university of Technology, Tehran, 15914, Iran.

E. Safavieh is an BSc student in Computer Science Department. AmirKabir University of Technology. Tehran, Iran (e.safavieh@srrf.net)

Ali Jahanbani Ardakani is an Msc Student in Electrical Engineering Department. AmirKabir University of Technology. Tehran, Iran.(Ali_jahanbani@aut.ac.ir)

Ali Kashefi Kavyani is an Msc Student in Electrical Engineering Department. AmirKabir University of Technology. Tehran, Iran.(Ali_kashefi@aut.ac.ir)

S. A. Pourmousavi is an Msc Student in Electrical Engineering Department. AmirKabir University of Technology. Tehran, Iran.(Pourmousavi@aut.ac.ir)

S. H. Hosseinian is with the Department of Electrical Engineering, AmirKabir University of Technology. Tehran, Iran. (Hosseinian@aut.ac.ir.)

M. Abedi is with the Department of Electrical Engineering, AmirKabir University of Technology. Tehran, Iran. (abedi@aut.ac.ir.)

et al. [1] utilize the linear prediction method in conjunction with filtering of the wind speed waveform as a new method for short-term wind speed forecasting.

In this study, an integration of wavelet-based networks and PSO are applied. Wavelet-based networks are proposed to model the highly nonlinear, dynamic behavior of the wind speed and to improve the performance of traditional ANNs. PSO is used for learning the wavelet-based networks as an optimizing algorithm, because the main object of training of artificial neural networks is founding weights and biases so that minimize training error, Hence we can approach problem of wavelet-based networks training as an optimization problem. The one-layer networks of the wavelet, the weighting, and the summing nodes are built by PSO algorithm.

The rest of this paper is organized as follows. Section II is devoted to exposition of wavelet-based networks. Then in section III PSO algorithm is explained. Section IV presents adaptation of PSO algorithm for wavelet-based networks training. Finally, in section V, results of simulation are given.

II. WAVELET NETWORKS

For the multidimensional function $f(x_1, x_2, \ldots, x_n)$ the wavelet function can be obtained by the scalar product of ψ :

$$\psi(x) = \psi(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} \psi(x_i) \tag{1}$$

The corresponding wavelet function family is defined as:

$$\Phi = \left\{ \psi_{ab}(x) = \det(A^{-1/2})\psi\left[Ax - b\right] \right. \\ \left. : b \in R^n, A = diag(a), a \in R_+^n \right\} \tag{2}$$

For more details about wavelet transform, see Daubechies [11].

Most Feed-forward Neural Networks (FNN) are the expansion models of the basis function (such as sigmoid basis, radial basis, spline basis, etc.), and which have good performance in function approximation. The network can be viewed as a complex nonlinear mapping model:

$$y = f(x) = \sum_{j=1}^{L} W_{j.N} . S\left(W_{M.j} . x - \theta_j\right) \tag{3}$$

where M and N are the node numbers of the input and output layers, respectively, L is the node number of the hidden layer, *S(.)* is the activating function of the node in the hidden layer, $W_{M,j}$ and $W_{j,N}$ are the connection weights from the input layer to the hidden layer and from the hidden layer to the output layer, respectively, and θ_j is the threshold of the hidden node j. Wavelet networks are just feed-forward neural networks with only one hidden layer consisting of the wavelet function from the wavelet family [12]. If the wavelet family (2) constitutes an orthonormal basis of $L^2(R^n)$ then selecting elements of this family for the construct wavelet network of the form in Eq. (4) will obviously be able to approximate any function of $L^2(R^n)$ [11]. More generally, family (2) may constitute a frame of $L^2(R^n)$ instead of a basis [13]. In this case, family (2) is redundant to span the $L^2(R^n)$ space. It is more convenient in practice to use a redundant wavelet family than an orthonormal wavelet basis for constructing the wavelet network, because admitting redundancy allows us to construct wavelet functions with a simple analytical form and good spatial-spectral localization properties. In this paper, we use such a non-orthogonal wavelet family to construct the wavelet network. The structure of the wavelet network is described in Fig. 1, and the output of the network is of the form Eq. (4) which is presented in [14].

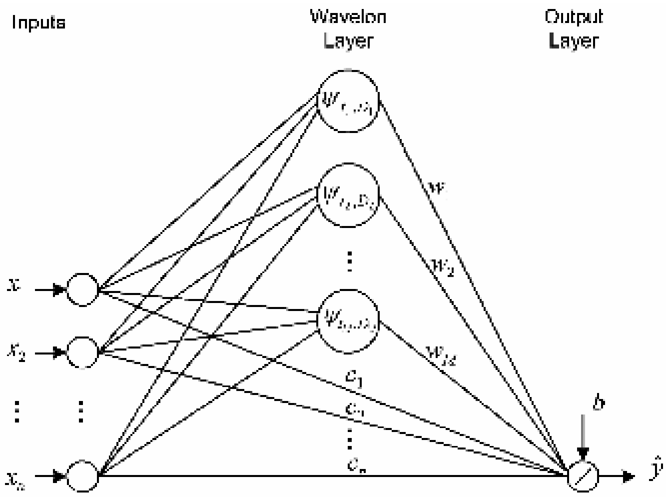

Fig. 1. Structure of a wavelet network.

$$\hat{y} = \sum_{i=1}^{M} w_i \psi_{a_i,b_i}(\mathbf{x}) + \mathbf{c}.\mathbf{x} + b \tag{4}$$

where w_i is the connection weight from the hidden layer to the output layer, $\boldsymbol{c}$ is the vector for linear combination and b is the parameter for zeroing. Introducing the parameter b into the network can make the network able to approximate the function with a nonzero mean. The network is trained to determine the parameters $\{w_i, a_i, b_i, c, b, i = 1, 2, \ldots, M\}$ by minimizing the mean square function of the training error, which is described as follows:

$$E = \frac{1}{2}\sum_{j=1}^{P}(y_j - \hat{y}_j)^2 \tag{5}$$

where P is the number of training samples, and y_j is the desired output of the jth sample.

The wavelet function is not the global function (infinite energy function, such as the sigmoid function), but the local function. Therefore, among the whole input scope of the network, the hidden nodes with the wavelet function influence the networks output only in some local range. This can obviate the interaction between the nodes and facilitate the training process and generalization performance. The radial basis is also the local function, but it does not have the spatial-spectral zooming property of the wavelet function, and therefore cannot represent the local spatial-spectral characteristic of the function. So, the wavelet network should have a better performance for approximation and forecasting than the traditional FNN.

III. Particle Swarm Optimization Algorithm

The last three decades have witnessed the development in efficient and effective stochastic optimizations. In contrast to the traditional adaptive stochastic search algorithms, evolutionary computation (EC) techniques exploit a set of potential solutions, named a population, and detect the optimal solution through cooperation and competition among the individuals of the population. These techniques often detect optima in difficult optimization problems faster than traditional methods [17]. One of the most powerful swarm intelligence-based optimization techniques, named Particle Swarm Optimization (PSO), was introduced by Kennedy and Eberhart [15, 16]. PSO is inspired from the swarming behavior of animals, and human social behavior. During last decade many studies focused on this method and almost all of them, strongly confirmed the abilities of this newly proposed optimization technique [15-20]. Abilities such as fast convergence, finding global optimum in presence of many local optima, simple programming and adaptability with constrained problems. Beside, some papers worked on improving this method by means of imposing additional variations such as variable inertia coefficient, constriction factor [18], maximum velocity limit, parallel optimization [19], deflection, repulsion, stretching [17], mutation [20] and so on.

PSO is a population-based algorithm that exploits a population of individuals to probe promising region of the search space. In this context, the population is called *swarm* and the individuals are called *particles*. Each particle moves with an adaptable velocity within the search space and retains in its memory the best position it ever encountered. The global variant of PSO the best position ever attained by all individuals of the swarm is communicated to all the particles [17]. The general principles for the PSO algorithm are stated as follows:

Suppose that the search space is *n*-dimensional, then the i^{th} particle can be represented by a *n*-dimensional vector, and velocity $V_i = \left[v_{i1}, v_{i2}, ..., v_{in}\right]^T$, where $i = 1, 2, ..., N$ and N is the size of population.

In PSO, particle i remembers the best position it visited so far, referred to as $P_i = \left[p_{i1}, p_{i2}, ... p_{in}\right]^T$, and the best position of the best particle in the swarm is referred as $G = \left[g_1, g_2, ..., g_n\right]^T$ [21].

Each particle i adjusts its position in next iteration $t+1$ with respect to Eqs. (5) and (6) [17]:

$$\begin{aligned} V_i(t+1) = w(t)V_i(t) + c_1 r_1 \left(P_i(t) - X_i(t+1)\right) \\ + c_2 r_2 \left(G_i(t) - X_i(t+1)\right) \end{aligned} \tag{5}$$

$$X_i(t+1) = X_i(t) + cV_i(t+1) \tag{6}$$

where $w(t)$ is inertia coefficient which gradually decreases from 1 at first iteration to a small magnitude about zero on a straight line. c is constriction factor which is used to limit velocity, here $c = 0.7$. c_1 and c_2 denote the cognitive and social parameters respectively, here both of them are set to 2. r_1 and r_2 random real numbers drawn from uniformly distributed interval $[0,1]$. Number of population is set to 5 times of number of variables and for preventing explosion of swarm, maximum allowable velocity is set to 4 times of number of variables. Results show that, in this application, algorithm converges within 300 to 400 iterations. Hence, 500 will be an acceptable number for iterations.

The inertia coefficient in (5) is employed to manipulate the impact of the previous history of velocities on the current velocity. Therefore, $w(t)$ resolves the trade off between the global and local exploration ability of the swarm. A large inertia coefficient encourages global exploration while small one promotes local exploration. Experimental results suggest that it is preferable to initialize it to a large value, giving priority to global exploration of search space, and gradually decreasing as to obtain refined solution [17].

IV. Adaptation of PSO for Wavelet Networks Training

Optimization problem for one layer wavelet-based networks are involved five parameters. They are translation matrix, delay matrix, weights of wavelons, direct weights coefficient and final biases. The aim of optimization is finding these parameters such that, the mean of squared errors, defined by (4), gets to least possible value.

Considering vector X^* as the optimal parameters of a given wavelet network, this vector can be find by means of some different optimization methods. This optimization problem can be solved deterministically by mathematical method, such as Gradient Descent, or by random based intelligent techniques like Genetic Algorithm (GA), Simulated Annealing (SA), Particle Swarm Optimization (PSO), and so on.

In this paper, PSO is applied to a one layer wavelet network to find its optimal parameters. Considering any given vector $X_i = \left[x_{i1}, x_{i2}, ..., x_{in}\right]^T$, where n is the number of network's parameters, represents a potential solution for X^*, optimal network parameters can be determined through a procedure consists of Eqs. (5) and (6). This network is utilized in order to predict very short term wind speed. The data is gathered from a Danish wind farm, where an anemometer records wind speed every 2.5 seconds. Assuming the wind speed as a time series, the wind speed in a given time step, $V(t)$, can be defined as a function of wind speeds in m preceding time steps. In our study m is set to 10. Consequently:

$$V(t) = f\left(V(t-1), V(t-2), \ldots, V(t-10)\right) \tag{7}$$

According to (7) and Fig. 1, proposed network consists of 10 inputs and one output.

Using MATLAB® based software provided by Zhang [22] and some edit on it we implement both Wavenet and PSO trained Wavenet.

One of the most important issues in training a good neural network is the number of hidden neurons (wavelons in case of wavenets). However, wavenet can approximate functions better as far as the number of wavelons increases, but this fact can not guarantee the accuracy of network's generalization. Consequently, we should make a trade off between number of wavelons and prediction error. In order to come over this problem, we divided the data base into 2 subsets, first one is used for training the wavenet and second one is utilized for validating the trained network's ability in predicting the wind speed during predicting intervals. Optimum number of wavelons can be obtained by trial and error, such that, the network is trained for a reasonable range of wavelons and optimum number is which that provides the least predicting error.

V. Results and Discussion

In this section, results are presented from the validation of the developed methodology for a wind farm in Denmark. Wind speed data resolution is 2.5 seconds. The prediction model is the wavelet-based networks with PSO for training.

A database consist of 900 successive samples is chosen for training the network and testing its accuracy in generalization of a proper function. The database is divided into two subsets. Each one includes 450 samples. Odd samples are used for training and even samples are used for validating of network's prediction ability. The procedure is run for a range of hidden neurons (HN) within 1 to 10. For more confidence about getting to nearest to optimal solution, the problem is solved for several times for each situation and the best attained result will be chosen as the optimal.

To show the efficiency of the proposed approach (WN-PSO), the method is compared with wavenet network trained by Gradient Descent (WN-GD) method and a Multi-Layer Neural Network (MLP-GD) trained by Gradient Descent method. Results indicate that proposed method can predict the short term wind speed better than conventional methods.

TABLE I
AVERAGE AND MAXIMUM PERCENTAGE PREDICTION ERROR FOR DIFFERENT NUMBER OF HIDDEN NEURONS AND METHODS.

# HN	WN-GD		MLP-GD		WN-PSO	
	$\bar{e}$	$\hat{e}$	$\bar{e}$	$\hat{e}$	$\bar{e}$	$\hat{e}$
1	3.304	19.40	3.314	19.79	3.217	19.13
2	3.347	19.28	3.281	19.87	3.291	19.65
3	3.298	18.40	3.273	21.42	3.291	19.65
4	3.441	19.99	3.266	21.64	3.291	19.65
5	3.400	18.92	3.263	21.61	3.291	19.65
6	3.503	21.52	3.260	21.65	3.291	19.65
7	3.362	19.38	3.267	22.62	3.291	19.65
8	3.540	21.38	3.394	40.78	3.291	19.65
9	3.507	19.18	3.401	42.44	3.291	19.65
10	3.409	20.02	3.385	37.97	3.291	19.65

Table I represents the ability of different aforementioned methods in predicting of wind speed. Considering the mean absolute percentage error ($\bar{e}$) as the criterion of efficiency, the optimum structure of wavenet network trained by Gradient Descent is obtained when it has 3 hidden neurons and corresponding error is equal to 3.298%. The optimal Multi-Layer Neural Network consists of one hidden layer including 6 neurons. The hidden layer utilizes well-known sigmoid activation function. Also, the output layer employs a linear activation function. The minimum percentage error obtained through several runs is 3.260%.

Table I confirms that a wavelet network trained by PSO achieves more accurate predictions rather than other methods (3.217%), although, the improvement is not significant.

Maximum absolute percentage error ($\hat{e}$), can be taken into account as another efficiency criterion. Results indicate that the maximum error of MLP networks is greater than wavenet's.

The interesting point is that, all wavenet networks, trained by PSO, converge to same optimum point. The network consisting of one wavelon is an exception. Justification of this phenomenon is above the aspects of this study. This fact must be considered in details in proceeding studies.

TABLE II
AVERAGE AND MAXIMUM ERROR IN WIND SPEED PREDICTION FOR DIFFERENT NUMBER OF HIDDEN NEURONS AND METHODS [M/S].

# HN	WN-GD		MLP-GD		WN-PSO	
	$\bar{e}$	$\hat{e}$	$\bar{e}$	$\hat{e}$	$\bar{e}$	$\hat{e}$
1	0.139	0.969	0.138	0.950	0.135	0.937
2	0.137	0.973	0.140	0.944	0.138	0.962
3	0.137	0.980	0.138	0.901	0.138	0.962
4	0.137	0.978	0.144	0.979	0.138	0.962
5	0.137	0.980	0.142	0.926	0.138	0.962
6	0.137	0.979	0.147	1.054	0.138	0.962
7	0.137	0.974	0.141	0.949	0.138	0.962
8	0.142	1.483	0.149	1.047	0.138	0.962
9	0.142	1.543	0.145	0.939	0.138	0.962
10	0.142	1.381	0.143	0.980	0.138	0.962

Table II shows the average and maximum error in wind speed prediction in m/s. Fig. 2 shows the results of proposed method and its comparison with real wind speed and prediction obtained by wavenet networks trained by Gradient Descent.

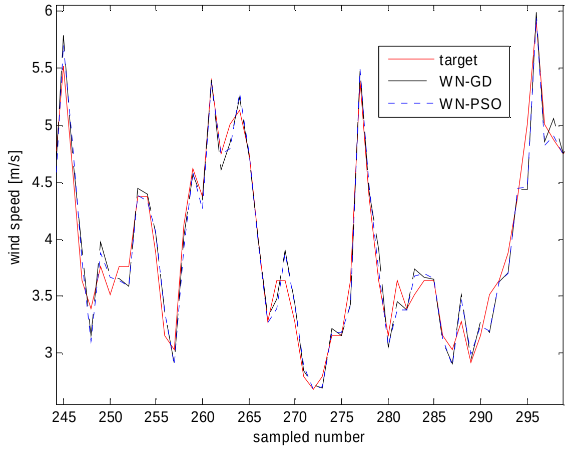

Fig. 2. wind speed prediction by WN-GD and WN-PSO.

Fig. 3 depicts the convergence of PSO in 5 runs. Almost in all runs PSO converges to same optimum point. In contrast, gradient descent method is very sensitive to initial points. This fact may conduct Gradient Descent to a local optimum point too far from global optimum point. For example, once Gradient Descent method reached to a maximum percentage error 108.39%, although its mean percentage error was 3.57%. It is a common problem during training neural networks by Gradient descent method. On the other hand, due to stochastical inherence of PSO, this method has the capability of avoiding from getting into local minima. PSO owes this capability to random coefficients r_1 and r_2 stated in Eq. (5).

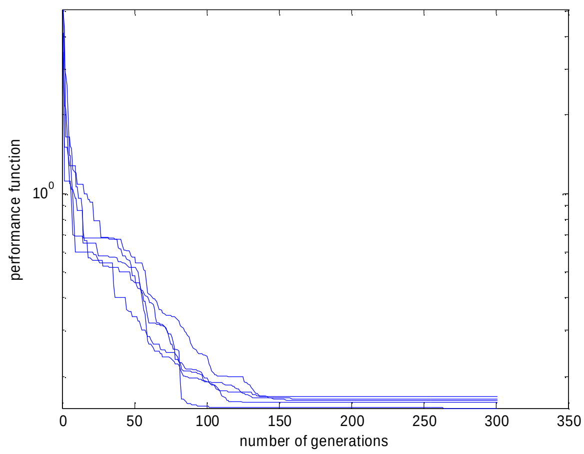

Fig. 3. PSO convergence through 5 runs.

VI. CONCLUSION

In this paper, a new method for very short term forecasting of wind speed is proposed. We approached the problem as time series. On this way, we tried to define the wind speed in any given time step as a function of wind speeds in 10 preceding time steps.

A wavelet based neural network is developed in order to find the nonlinear relation between wind speed and its history. An intelligent optimization method named PSO was exploited in order to train this network. Results showed that PSO can find optimal network parameters. Optimum structure was obtained when our wavenet had only one wavelon in its hidden layer.

To measure the efficiency of proposed method, results were compared with results provided by wavenet and Multi Layer Neural Network trained by well known Gradient Descent method. Comparison showed that proposed method offers a mean absolute percentage error less than both other methods.

Furthermore, PSO represents other privileges over conventional methods. Privileges include simplicity in implementation and robustness in converging to nearest point optimal solution. On the other hand, PSO is more time consuming than other conventional methods especially when number of optimization variables increases, the elapsed time severely goes up. This fact indicates that there is a nonlinear relation between number of variables and computation time.

Interestingly, it was witnessed that when the number of wavelons is more that 2, the PSO always converges to same optimal solution, whereas other techniques are very sensitive to variations of number of hidden neurons.

VII. REFERENCES

[1] G.H. Riahy, M.Abedi,"Short term wind speed forecasting for wind turbine applications using linear prediction method", Renewable Energy, 2007.

[2] M.C. Alexiadis, P. S. Dokopoulos,H. S. Sahsamanoglou, "Short-term forecasting of wind speed and related electrical power", Solar EnergyVol. 63, No 1,pp 61-68,1998.

[3] U.Hoppmann, S. Keonig, T. Tielkes, G. Matschke, "A short-term strong wind prediction model for railway application: design and verification", Wind Eng & Industrial aerodynamics, 2002, 1127-1134.

[4] G. N. Kariniotakis, P. Pinson, "Uncertainty of short-term wind power forecasts- A methodology for on-line assessment.", IEEE conference on probabilistic methods applied to power systems, Iowa State University ,September 12-16,2004.

[5] C. W. Potter, M. N. Negnevitsky, "Very short-term wind forecasting for Tasmanian power generation." IEEE Transaction on power systems, Vol. 21, NO. 2, MAY 2006.

[6] M. N. Negnevitsky, C. W. Potter, Innovative short-term wind generation prediction techniques." PSCE, IEEE 2006.

[7] Ahmet Oztopal, "Artificial neural network approach to spatial estimation of wind velocity data." Energy Conversion and Management (2006) 395-406.

[8] E. Cadenas, W. Rivera, "Wind speed forecasting in the south coast of Oaxaca, Mexico." Renewable Energy 32 (2007) 2116-2128.

[9] Ahmet D. Sahin, Zekai Sen, "First-order Markov chain approach to wind speed modeling.", Journal of Wind Engineering and Industrial Aerodynamics 89 (2001) 263-269.

[10] A. Shamshad, M.A. Bawadi, W.M.A. Wan Hussin, T.A. Majid, S.A.M. Sanusi, "First and second order markov chain models for synthetic generation of wind speed time series.", Energy 30 (2005) 693-708.

[11] Daubechies I. Ten lecture on wavelets. CBMS-NSF regional series in applied mathematics. Philadelphia, PA: SIAM, 1992

[12] Q. Zhang and A. Benveniste, "Wavelet networks," IEEE Trans. Neural Netw., vol. 3, no. 6, pp 889-898, Nov. 1992.

[13] Yongyong He, Fulei Chu and Binglin Zhong, "A Hierarchical Evolutionary Algorithm for Constructing and Training Wavelet Networks", Neural Comput & Applic (2002)10:357–366

[14] Q. Zhang, "Using wavelet network in nonparametric estimation," *IRISA/INRIA*, Campus de Beaulieu, Rennes Cedex, France, Tech. Rep. 833, May 1994.

[15] J. Kennedy and R. C. Eberhart, "Particle Swarm Optimization", in proc. IEEE int. Conf. Neural Networks, Perth, Australia, 1995, vol. IV, pp. 1942-1948.

[16] R. C. Eberhart and J. Kennedy, "A new Optimizer Using Particle Swarm Theory", in proc. 6^{th} Symp. Micro Machine and Human Science, Nagoya, Japan, 1995, pp. 34-44.

[17] Konstantinos E. Parasopoulos and Michael N. Vrahatis, "On the Computation og All Global Minimizers Through Particle Swarm optimization", IEEE Trans. On Evolutionary Computation, vol. 8, No. 3, June 2004.

[18] K.E. Parsopoulos and M.N. Vrahatis. Particle swarm optimization method for constrained optimization problems. In P. Sincak, J. Vascak, V. Kvasnicka, and J. Pospichal, editors, Intelligent Technologies--Theory and Application: New Trends in Intelligent Technologies, volume 76 of Frontiers in Artificial Intelligence and Applications, pages 214--220. IOS Press, 2002.

[19] J. F. Scutte, J. A. Reinbolt, B. J. Fregly, R. T. Haftka and A. D. George, "Parallel global optimization with the particle swarm algorithm", Int. Journal for Numerical Methods in Engineering 2004; 61:2296-2315.

[20] Raymond R. Tan, "Hybrid evolutionary computation for the development of pollution prevention and control strategies", Journal of Cleaner Production 15 (2007) 902-906.

[21] B. Jaboui et al., "A combinatorial particle swarm optimization for solving …", Appl. Math. Comput. (2007), doi: 10.1016/j.amc.2007.04.096.

[22] Q. Zhang, 1998, *Wavenet*, Matlab file, Available: ftp://ftp.irisa.fr/local/wavenet.