SOLAR RADIATION PREDICTION BASED ON PARTICLE SWARM OPTIMIZATION AND EVOLUTIONARY ALGORITHM USING RECURRENT NEURAL NETWORKS

Nian Zhang
Department of Electrical and Computer Engineering,
University of the District of Columbia, 4200 Connecticut Avenue, NW, Washington, D.C. 20008 USA
nzhang@udc.edu

Pradeep K. Behera
Department of Civil and Mechanical Engineering,
University of the District of Columbia, 4200 Connecticut Avenue, NW, Washington, D.C. 20008 USA
pbehera@udc.edu

Charles Williams
Department of Electrical and Computer Engineering,
University of the District of Columbia, 4200 Connecticut Avenue, NW, Washington, D.C. 20008 USA
charles.williams4@udc.edu

***Abstract*—Over the last decade, there has been emphasis on the reduction of the dependency of fossil fuels that resulting in the growth of renewable energy industries. These industries have been significant economic drivers in many parts of the United States supported by both government and private sectors. As a part of renewable energy industries, there is a strong growth in solar power generation industries that often requires prediction of solar energy to develop highly efficient stand-alone photovoltaic systems as well as hybrid power systems. Specifically solar radiation prediction is a important component in the solar energy production.**

However, some computational intelligence methods that have most successful applications on time series prediction have not yet been investigated on solar radiation prediction. Only a limited number of neural networks models were applied to the solar radiation monitoring. Therefore, we propose an Elman style based recurrent neural network to predict solar radiation from the past solar radiation and solar energy in this research. A hybrid learning algorithm incorporating particle swarm optimization and evolutional algorithm was presented, which takes the complementary advantages of the two global optimization algorithms. The neural networks model was trained by particle swarm optimization and evolutional algorithm to forecast the solar radiation. The excellent experimental results demonstrated that the proposed hybrid learning algorithm can be successfully used for the recurrent neural networks based prediction model for the solar radiation monitoring.

Keywords—Solar radiation prediction; time series prediction, neural network; particle swarm optimization, evolutionary algorithm

I. INTRODUCTION

The energy demand is increasing at a breathtaking rate globally, requiring significant investment in new power generation capacity and grid infrastructure. By 2030 world energy needs are predicated to be between 30 and 60% higher than current levels depending on the implemented efficiency measures. The International Energy Agency (IEA) has estimated the requirement of new energy capacity of approximately 4,800 GW before 2030 to meet the energy demand, which entails an investment of almost US$ 4 trillion [1].

Due to strong increase of solar power generation, the predictions of solar energy are critical in terms of importance. Photovoltaic and solar thermal are the main sources of electricity generation from solar energy. The accurate predictions of the solar radiation evolution enable efficient sizing and improved performance of stand-alone photovoltaic systems [2], and of hybrid power systems [3][4].

Many research studies have been performed to forecast the solar radiation in recent years. They benefit substantially from the progress of computational intelligence techniques [5]. The techniques include wavelet neural network [6][7], support vector machine [8], recurrent neural network [9][10], echo state network [11], adaptive neural fuzzy inference systems (ANFIS) [12], and radial basis function (RBF) neural network [13], and other kinds of neural networks [14][15][16][17]. In comparison with the conventional statistical approach [18], neural networks based forecast models perform much better in terms of forecast accuracy. However, a fact could not be neglected that most of the existing computational intelligence based models have not yet satisfied researchers in forecast precision, and the generalization capability of these networks needs further

978-1-4673-3108-1/13/$31.00 ©2013 IEEE

improving. In addition, none of the above computational intelligence methods is used for the solar radiation prediction in the District of Columbia and the suburbs.

To address the above problems, it is extremely important to investigate state-of-the-art computational intelligence techniques that have potential for improving the forecasting of solar radiation. Based on the fact that neural networks [19], genetic regulatory network [20], echo state network [21], particle swarm optimization [22][23], and other computational intelligence methods [24][25] have very successfully applications on the time series prediction problems, and because time series prediction is a generalized form of solar radiation prediction, we expect these methods will also work the best for the solar radiation prediction problem.

This paper is organized as follows. In Section 2, the background of the Zero Energy Center located at the University of the District of Columbia campus Washington DC. is presented. The solar radiation from the center is presented. In Section 3, The Elman style recurrent neural network is presented, followed by the description of the evolutionary algorithm (EA), particle swarm optimization (PSO), and the hybrid of these two methods. In Section 4, the experimental results are demonstrated including the training error for the PSO-EA method and the predicted values of the training data. In Section 5, the conclusions are provided.

II. BACKGROUND

A. Zero Energy Center at University of the District of Columbia, Washington D. C.

Real-time solar energy data and solar radiation data are obtained from the Zero Energy Center located at the University of the District of Columbia (UDC) campus. This station supported by the NSF funding plays an extremely important role in recording real-time solar data and wind data in the District of Columbia. While the Renewable Resource Data Center at the National Renewable Energy Laboratory (NREL) only provides solar data till year 2005, this station can provide solar data for the District of Columbia from Year 2006 till present.

The Zero Energy Center is capable of delivering 4.5KW nominal renewable energy power. It consists of a solar tracking photovoltaic (PV) array and a Whisper H80 wind turbine to monitor, and record solar radiation data and weather data, as shown in Fig. 1.

Fig. 1. Zero Energy House

B. Solar Radiation Data

The solar radiation data were retrieved from the Zero Energy Center between June 27, 2011 and July 15, 2011 and is plotted in Fig. 2. Real-time data typically are recorded at 5-minute intervals, and thus Fig. 2 plots 5114 data. The entire data set was randomly divided up the 5114 time steps. The input, solar radiation data has a 5114x1 matrix, representing dynamic data, i.e. 5114 time steps of 1 element.

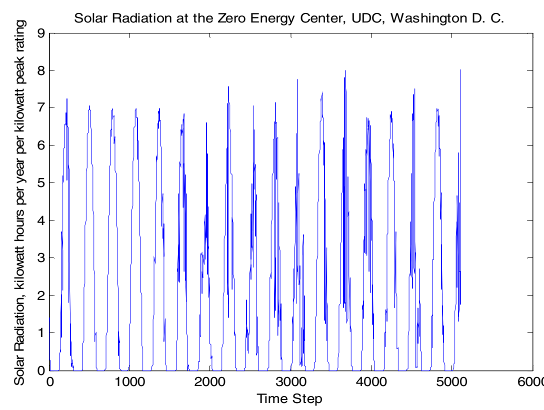

Fig. 2. The solar radiation data collected at the Zero Energy Center at the University of the District of Columbia at Washington D. C. between June 27, 2011 and July 15, 2011.

III. ARCHITECTURE AND ALGORITHMS

A. Recurrent Neural Network

We proposed an Elman based recurrent neural network, which is composed of five layers, input layer, hidden layer 1, hidden layer 2, and output layer. There are feedback connections from the outputs of the hidden layer 1 to the inputs of the context layer, as shown in Fig. 3.

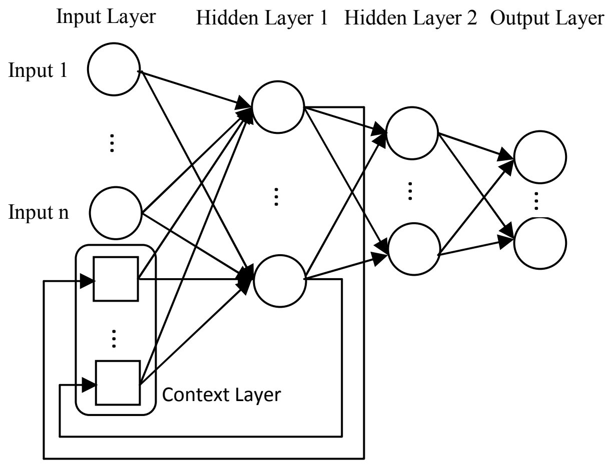

Fig. 3. The topological structure of the Elman neural networks model. The network is composed of five layers, with a feedback connection from the outputs of the hidden layer 1 to the input of the context layer.

In the recurrent neural network model, the input layer has 100 neurons, the context layer has 40 neurons, the hidden layer 1 has 40 neurons, the hidden layer 2 has 20 neurons, and the output layer has 1 neuron. Neurons between adjacent layers are fully connected, as indicated in Fig. 3. The transfer functions of the two hidden layers and the output layer are *tansig*.

B. Evolutionary Algorithm (EA)

To begin the evolutionary algorithm, a population of *n* neural networks, P_i, i=1,.., *n*, defined with weights and bias for each network, was created at random. Each neural network had an associated self-adaptive parameter vector σ_i, i=1,…, *n*, where each component corresponded to a weight or bias and served to control the step size of the search for new mutated parameters of the neural network. Each parent generated an offspring strategy by varying all of the associated weights and biases. Specifically, for each parent P_i, i=1,…, *n*, an offspring $P_i^{'}$, i=1,…, *n*, was created by

$$\sigma_i^{'}(j) = \sigma_i(j)\exp(\tau N_j(0,1)), \qquad j = 1,..., N_w \qquad (1)$$

$$w_i^{'}(j) = w_i(j) + \sigma_i^{'} N_j(0,1), \qquad j = 1,..., N_w \qquad (2)$$

where N_w is the number of weights and biases in the recurrent neural network, $\tau = 1/\sqrt{2\sqrt{N_w}}$, and $N_j(0,1)$ is a standard Gaussian random variable resampled for every *j* [26].

C. Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is a form of evolutionary computation technique developed by Kennedy and Eberhart [27][28][29]. Similar to Evolutionary Algorithms (EA), particle swarm optimization algorithm is a population based optimization tool, where the system is initialized with a population of random solutions and the algorithm searches for optima satisfying some performance index over generations. It is unlike an EA, however, in that each potential solution is also assigned a randomized velocity, and the potential solutions, called *particles*, are then "flown" through the problem space.

Each particle has a position represented by a position vector $\vec{x}_i$. A swarm of particles moves through the problem space, with the velocity of each particle represented by a vector $\vec{v}_i$. At each time step, a function f representing a quality measure is calculated by using $\vec{x}_i$ as input. Each particle keeps track of its own best position, which is recorded by a vector $\vec{p}_i$, where $f(\vec{p}_i)$ is the best fitness it has achieved so far. Furthermore, the global best position among all the particles obtained so far in the population is kept track of as $\vec{p}_g$, and its corresponding fitness as $f(\vec{p}_g)$.

At each time step *t*, by using the individual best position, $\vec{p}_i(t)$, and global best position, $\vec{p}_g(t)$, a new velocity for particle *i* is updated by

$$\vec{v}_i(t+1) = w \times \vec{v}_i(t) + c_1\phi_1(\vec{p}_i(t) - \vec{x}_i(t)) + c_2\phi_2(\vec{p}_g(t) - \vec{x}_i(t)) \qquad (3)$$

where c_1 and c_2 are positive constants, ϕ_1 and ϕ_2 are uniformly distributed random numbers in [0, 1] and *w* is the inertia weight. The term $\vec{v}_i$ is limited to the range $\pm\vec{v}_{\max}$. If the velocity violates this limit, it is set at its proper limit. Changing velocity this way enables the particle *i* to search around its individual best position, $\vec{p}_b$, and global best position, $\vec{p}_g$. Based on the updated velocities, each particle changes its position according to the following:

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t+1) \qquad (4)$$

Based on (3) and (4), the population of particles tends to cluster together with each particle moving in a random direction. Fig. 4 illustrates the procedure of the PSO algorithm [30].

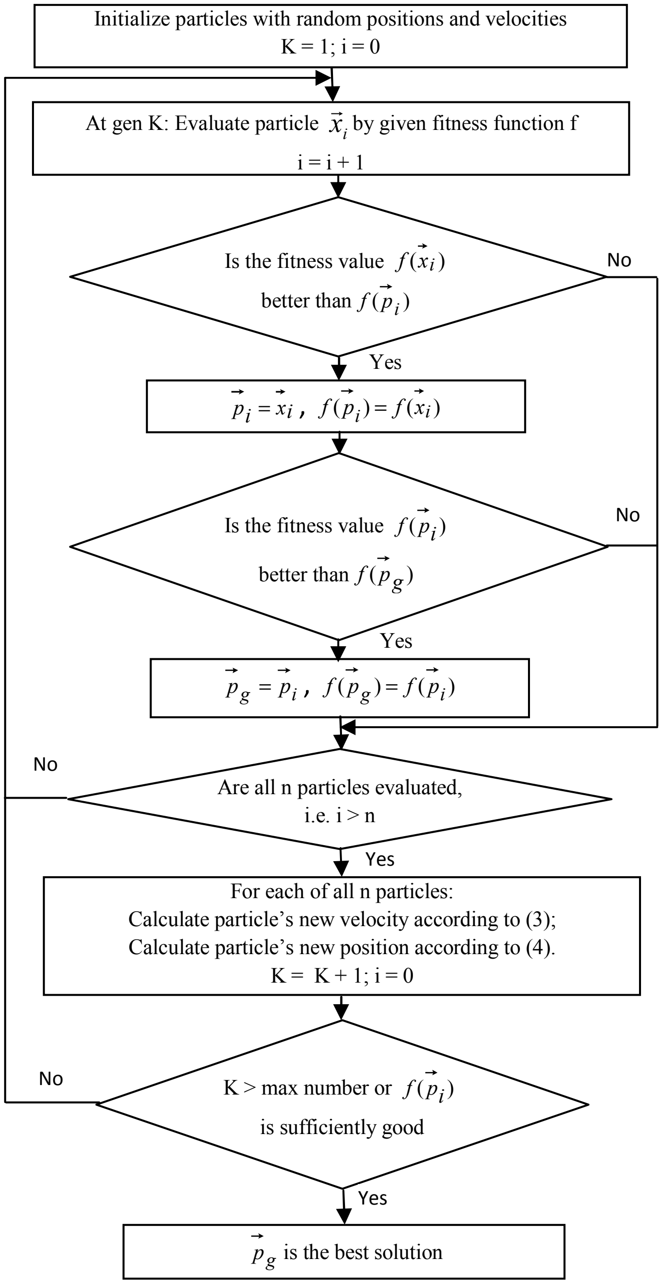

Fig. 4. The flow chart of the PSO algorithm.

D. Integration of EA and PSO

PSO works based on social and cognitive adaptation of knowledge, and all individuals are considered to be of same generation. On the contrary, EA works based on evolution from generation to generation, so the changes of individuals in a single generation are not considered. EA discard valuable information at the end of generation and starts almost randomly at next generation, while PSO keeps such information with the memory of local and global best throughout the entire evolution. On the hand, the property of mutation in EA helps to maintain the diversity of PSO population in "flying" to the new search area.

Based on the complementary property of PSO and EA, a hybrid algorithm is created that combines the concepts of both algorithms. In each generation, the winners, which constitute half of the population, are enhanced by PSO. These winners are sharing information with each other as well as benefiting from their learning history, compared to EA where they are stagnant. The other half of the population which consists of individuals with lower fitness is replaced by the offspring created from the EA process with influence from the PSO enhanced parents. This procedure enhances the entire population.

The pseudo code for hybrid PSO–EA is summarized as follows [22]:

Initialize a population of individuals with random positions and velocities in an n-dimensional problem space.
Do
  Evaluate the fitness according to same given fitness function.
  Compare the fitness values to find the winners.
  Enhance the winner with PSO.
  For each elite:
    Update the p_b if the current particle's fitness value is better than the p_b;
    Determine p_g : choose the particle with the best fitness value of winners;
    Calculate particle's new velocity according to (3);
    Calculate particle's new position according to (4).
  Use the enhanced elites as parents to produce offspring with EA to replace losers for the next generation.
  For each offspring:
    Save parent's p_i as current p_i for further comparison;
    Use parent's velocity as self-adaptive parameters;
    Calculate the self-adaptive parameter according to (1);
    Calculate the position according to (2).

IV. EXPERIMENTAL RESULTS

A. Recurrent Neural Network Trained with PSO-EA

The recurrent neural network in Fig. 3 was trained by using the proposed PSO-EA algorithm. The input vector is composed of both original samples and the network's previous predictions. The recurrent neural network is not only trained with the original time series, but also trained by the series of sequence differences, represented by $y^*(n) = y(n) - y(n-1)$, where $y(n)$ and $y(n-1)$ are obtained from the given time series data. This dynamic signal is fed into RNN the same way as the original data.

We used batch training method, and the weights are updated based on a cumulative error function.

The original data were normalized. The mean value of the original data was subtracted first. Such a zero mean sequence is then divided by its maximum absolute value to fit between -1 and 1.

B. Results

The inertia weight w in Eq. (3) controls the balance of global and local search ability. A large w facilitates the global search while a small one enhances local search. In addition, since the search process of a PSO is nonlinear and complicated, static parameters, if well selected, can do a good job, but much better performance can be achieved if a dynamically changing scheme for the parameters is well designed, either a linearly decreasing inertia weight [31], a nonlinearly fuzzy adaptation [32], or involving a random component rather than time-decreasing [33]. All intuitively assume that the PSO should favor global search ability at the beginning and local search at the end. Based on the previous research in literature [22][31], the swarm size was chosen to be 40, the inertia weight was chosen to be 0.8. The acceleration constants c_1 and c_2 in Eq. (3) were set to 2 and 2, respectively.

A population of 40 particles was evolved for 100 generations. The training error for the PSO-EA method is shown in Fig. 5. As shown in Fig. 5, the cumulative root mean square error (RMSE) for the best individual using the hybrid PSO-EA algorithm dropped to 0.028 after 100 iterations.

The actual and predicted values of the first 500 training data are illustrated in Fig. 6. As shown in Fig. 6, the actual data and predicted data matched very well. The experimental results demonstrated that the algorithm is performing very well and is suitable for solar radiation prediction.

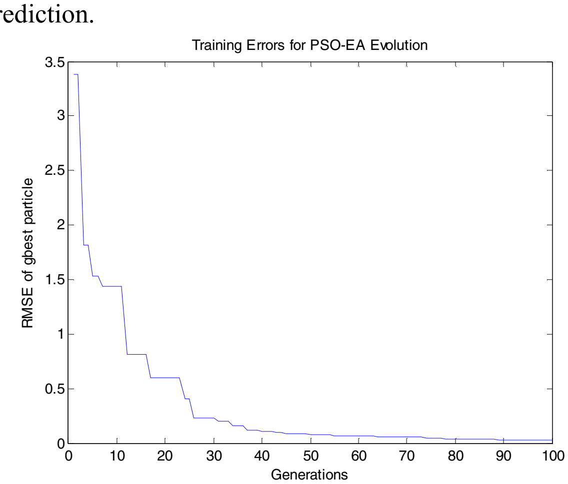

Fig. 5. Training error for the PSO–EA algorithm. The errors reflect the performance of the best particle, i.e. the Pg, at each generation.

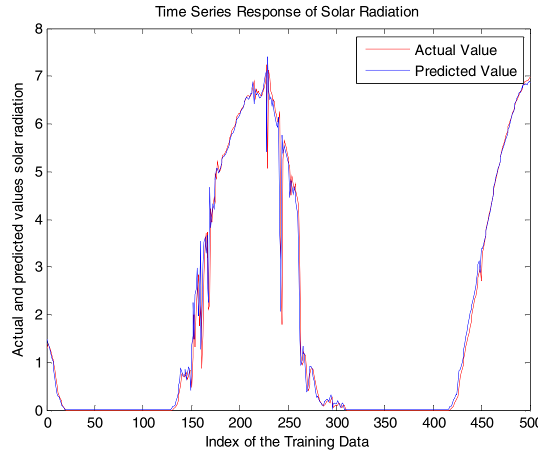

Fig. 6. The actual and predicted values of solar radiation.

V. CONCLUSIONS

This paper provides a recurrent neural network based predictive model trained by a combination of particle swarm optimization and evolutionary algorithm to forecast the solar radiation. This method explored a new neural network based solution for monitoring and controlling solar radiation.

In this paper, the solar radiation data at the Zero Energy Center at UDC was studied. An Elman style based recurrent neural network was constructed. A hybrid training algorithm incorporating particle swarm optimization and evolutional algorithm was investigated, which takes the complementary advantages of the two global optimization algorithms. PSO keeps valuable information with the memory of local and global best throughout the entire evolution. On the hand, the property of mutation in EA helps to maintain the diversity of PSO population in "flying" to the new search area.

The experimental results demonstrated that the proposed neural network based predictive model and the training algorithm ensure an accurate prediction on the solar radiation. This provides an excellent prediction method for the solar radiation monitoring, and has impact to the efficient sizing and improved performance of stand-alone photovoltaic systems and of hybrid power systems

ACKNOWLEDGEMENT

The authors would like to express thanks to the University of the District of Columbia STEM Center (NSF/HBCU-UP/HRD-0928444) grant and DC Water Resources Research Institute (WRRI) Grant.

REFERENCES

[1] International Energy Agency, "World Energy Outlook 2009 Edition," ISBN 978-92-64-06130-9, 2009.

[2] A. Mellit, M. Benghanem, A. Hadj, A. Guessoum, K. Moulai, "Neural Network Adaptive Wavelets for Sizing of Stand-Alone Photovoltaic Systems," *Proceedings of*

the 2nd International IEEE Conference on Intelligent Systems, vol. 1, pp. 365 – 370, 2004.
[3] N. Amjady, “Short-Term Hourly Load Forecasting Using Time-Series Modeling with Peak Load Estimation Capability,” *IEEE Transactions on Power Systems*, vol. 16, no. 4, pp. 798-805, 2001.
[4] T. Zhu, S. R. Shaw, S. B. Leeb, “Transient Recognition Control for Hybrid Fuel Cell Systems,” *IEEE Transactions on Energy Conversion*, vol. 21, no. 1, pp. 885-896, 2006.
[5] S. Haykins, *Neural Networks: A Comprehensive Foundation*, Second Edition, Prentice Hall, 1998.
[6] H. Zhou, W. Sun, D Liu, J. Zhao, N. Yang, “The Research of Daily Total Solar-Radiation and Prediction Method of Photovoltaic Generation Based on Wavelet-Neural Network,” *2011 Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, pp. 1-5, 2011.
[7] G. Capizzi, F. Bonanno, C. Napoli, “A Wavelet Based Prediction of Wind and Solar Energy for Long-Term Simulation of Integrated Generation Systems,” *2010 International Symposium on Power Electronics Electrical Drives Automation and Motion (SPEEDAM)*, pp. 586 - 592, 2010.
[8] F. Deng, G. Su, C. Liu, Z. Wang, “Prediction of Solar Radiation Resources in China Using the LS-SVM Algorithms,” *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, vol. 5, pp. 31-35, 2010.
[9] S. Cao, “Total Daily Solar Irradiance Prediction using Recurrent Neural Networks with Determinants,” *2010 Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, pp. 1-4, 2010.
[10] N. Zhang and P. Behera, “Solar Radiation Prediction Based on Recurrent Neural Networks Trained by Levenberg-Marquardt Backpropagation Learning Algorithm,” *The Third IEEE PES Conference on Innovative Smart Grid Technologies (ISGT 2012)*, Washington, D. C., January 16-20, 2012.
[11] S. M. Ruffing, G. K. Venayagamoorthy, “Short to Medium Range Time Series Prediction of Solar Irradiance Using an Echo State Network,” *15th International Conference on Intelligent System Applications to Power Systems (ISAP '09)*, pp. 1-6, 2009.
[12] A. Mellit, A. H. Arab, N. Khorissi, H. Salhi, “An ANFIS-Based Forecasting for Solar Radiation Data from Sunshine Duration and Ambient Temperature,” *IEEE Power Engineering Society General Meeting*, pp. 1-6, 2007.
[13] E. M. Crispim, P. M. Ferreira, A. E. Ruano, “Solar Radiation Prediction Using RBF Neural Networks and Cloudiness Indices,” *International Joint Conference on Neural Networks (IJCNN '06)*, pp. 2611 – 2618, 2006.
[14] A. Ghanbarzadeh, A. R. Noghrehabadi, E. Assareh, M. A. Behrang, “Solar Radiation Forecasting Based on Meteorological Data Using Artificial Neural Networks,” *The seventh IEEE International Conference on Industrial Informatics*, pp. 227 – 231, 2009.
[15] L. El Chaar, L. A. Lamont, M. Karkoub, “Modeling Climate Parameters for Renewable Energy Applications in the UAE Using Neural Networks,” *CIGRE/IEEE PES Joint Symposium on Integration of Wide-Scale Renewable Resources Into the Power Delivery System*, pp. 1-10, 2009.
[16] D. Paul, S. N. Mandal, D. Mukherjee, S. Chaudhuri, “Artificial Neural Network Modeling for Efficient Photovoltaic System Design,” *International Conference on Advanced Computer Theory and Engineering*, pp. 50 – 56, 2008.
[17] A. Mellit, M. Menghanem, M. Bendekhis, “Artificial Neural Network Model for Prediction Solar Radiation Data: Application for Sizing Stand-Alone Photovoltaic Power System,” *IEEE Power Engineering Society General Meeting*, vol. 1, pp. 40 – 44, 2005.
[18] J. Wu, C. Chan, J. Loh, F. Choo, L. H. Chen, “Solar Radiation Prediction Using Statistical Approaches,” *7th International Conference on Information, Communications and Signal Processing (ICICS 2009)*, pp. 1-5, 2009.
[19] D. K.. Barrow, S.F. Crone, N. Kourentzes, “An Evaluation of Neural Network Ensembles and Model Selection for Time Series Prediction,” *2010 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8, 2010.
[20] N.A. Barker, C.J. Myers, H. Kuwahara, Learning Genetic Regulatory Network Connectivity from Time Series Data, *IEEE/ACM Transactions on Computational Biology and Bioinformatics,* vol. 8, no. 1, pp. 152 – 165, 2011.
[21] A. Rodan, P. Tino, “Minimum Complexity Echo State Network,” *IEEE Transactions on Neural Networks,* vol. 22, no.1, pp. 131 – 144, 2011.
[22] X. Cai, N. Zhang, G. K. Venayagamoorthy, and D. C. Wunsch, II, Time Series Prediction with Recurrent Neural Networks Trained by a Hybrid PSO-EA Algorithm, *Neurocomputing,* vol. 70, no. 13-15, pp. 2342-2353, 2007.
[23] X. Cai, N. Zhang, G.K. Venayagamoorthy, D.C. Wunsch II, “Time Series Prediction with Recurrent Neural Networks Using Hybrid PSO-EA Algorithm,” *INNS-IEEE International Joint Conference on Neural Networks (IJCNN),* vol. 2, Budapest, Hungary, pp. 1647-1652, 2004.
[24] Artificial Neural Network & Computational Intelligence Forecasting Competition in 2006, 2008, and 2010. http://www.neural-forecasting-competition.com/
[25] A. Lendasse, E. Oja and O. Simula, M. Verleysen, Time Series Prediction Competition: The CATS Benchmark, *INNS-IEEE International Joint Conference on Neural Networks (IJCNN)*, Budapest, Hungary, pp. 1615-1620, 2004.
[26] K. Chellapilla, D.B. Fogel, “Evolution, Neural Networks, Games, and Intelligence,” *Proc. IEEE Spec. Issue Comput. Intelligence*, vol. 87, no. 9, pp. 1471–1496, 1999.
[27] R. Eberhart, Y. Shi, “Particle Swarm Optimization: Developments, Applications and Resources,” *Proceedings*

of the 2001 Congress on Evolutionary Computation, vol. 1, pp. 81–86, 2001.
[28] J. Kennedy, R. Eberhart, Y. Shi, *Swarm Intelligence*. Morgan Kaufmann, San Meteo, 2001.
[29] J. Kennedy, R. Eberhart, “Particle Swarm Optimization,” *IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.
[30] X. Cai, G. K. Venayagamoorthy, and D. C.Wunsch, “Evolutionary Swarm Neural Network Game Engine for Capture Go,” Neural Networks, vol. 23, no. 2, pp. 295-305, 2010.
[31] Y. Shi and R.C. Eberhart, “Parameter Selection in Particle Swarm Optimization,” *Proceeding of the 7th International Conference on Evolutionary Programming*, vol. 7, pp. 591-600, 1998.
[32] Y. Shi and R.C. Eberhart, “Fuzzy Adaptive Particle Swarm Optimization,” *Proceedings of the Congress on Evolutionary Computation*, 2001.
[33] Y. Shi and R.C. Eberhart, “Particle Swarm Optimization with Fuzzy Adaptive Inertia Weight,” *Proceeding of the Workshop on Particle Swarm Optimization*, 2001.