Annual Electricity Demand Prediction for Iranian Agriculture Sector Using ANN and PSO

Seyyed Ali Pourmousavi Kani, and Nima Farrokhzad Ershad

Abstract--**In this study, we used PSO algorithm and ANN to predict annual electricity consumption in Iranian agriculture sector. The economic indicators used in this paper are price, value added, number of customers and consumption in the previous periods. To predict the future values, a linear-logarithmic model of electrical energy demand is considered. The PSO algorithm applied in this study has been tuned for all its parameters and the best coefficients with minimum error are identified, while all parameter values are tested concurrently. Consumption in the previous periods has been used for testing estimated model. The estimation errors of PSO algorithm are less than that of estimated by genetic algorithm and regression method. In addition, ANN is used to forecast each independent variable and then electricity consumption is forecasted up to year 2010. Electricity consumption in Iranian agriculture sector from 1981 to 2005 is considered as the case for this study.**

***Index Terms*—Artificial Neural Networks, Electricity demand, Linear-logarithmic model, Prediction, PSO algorithm.**

I. Introduction

Due to essential needs to synchronize electricity production and consumption, technological constraints for storing electricity in large scales, and optimal operation and planning in power systems, it is vital to predict and estimate electricity demand in different time scales. Predicting demand provides the operators needed information about future conditions of the network. This information provides possibility to predict essential improving actions such as putting power plants in their maximum production, electricity purchasing, switching and etc to operate the power system in safe conditions. On the other hand, accurate prediction of demand causes to decrease costs and improve power system safety, making it possible for utilities to produce, purchase and sell energy in optimal prices. Also, an accurate demand prediction model is a vital part of energy management systems for Appling different tariffs in different time ranges.

Demand prediction depends on time ranges and their applications, divided in to four categories: very-short-term, short-term, mid-term and long-term. Where the forecasting period is a few minutes this is referred to as very-short-term (VSTLF), if the time spanned ranges from a few hours up to 1 week then this is covered by the most widely available models called short-term (STLF), STLF calculates the load of every hour of the day and this estimate can be used to control the number of generators in use and the shutdown of some units when the load forecast is low or their start-up when high loads are foreseen. In the case in which the time period lasts from a few weeks up to 1 year then we are dealing with mid-term (MTLF). Lastly, if the activity forecast spans a period from a few years up to one or two decades then these are long-term (LTLF) and they are generally used to aid resource planning activities and to assess the need for restructuring or extension of production facilities, determination of electricity production, transmission and distribution capacity, types of equipments and planning for production and transmission development and make maintenance schedules. In this study, long-term load forecasting is considered for Iranian agriculture sector (LTLF).

The estimation of electrical energy demand based on economic indicators may be done with different kinds of mathematical models. These equations might be linear or non-linear. Due to the fluctuations of economic indicators, the non-linear forms of the equations can predict electrical energy demand more effectively. The non-linearity of economical indicators and electrical energy demand has lead to search for different solution approach methods of evolutionary algorithms. Particle Swarm Optimization (PSO) algorithm is a form of evolutionary computation technique developed by Kennedy and Eberhart [13-15]. PSO algorithms are optimizing and stochastic search techniques which possess vast and powerful applications. PSO algorithm has not been used for optimizing parametric values of predictor equations. Almost all works in this area has been performed using genetic algorithm. The estimation of Turkey's energy demand based on economic indicators using genetic algorithm was reported by Ceylan and co-workers in 2003 [1]. Hepbasli estimated industrial electricity demand using genetic algorithm [2]. Osman et al. presented a combined GA – fuzzy logic controller technique for constrained nonlinear programming problems so that the search region is able to adapt toward the promising area [3]. Tang, Quek and Ng have used a genetic algorithm based Takagi–Sugeno–Kang fuzzy neural network to tune the parameters in Takagi–Sugeno–Kang fuzzy neural network [4]. Muni et al. proposed genetic programming methodology simultaneously selects a good subset of features and constructs a classifier using the selected features [5]. Some researches have been carried out recently to estimate the energy consumption using genetic algorithm [6,7]. Azadeh et al. proposed an integration of ANN and GA to predict electricity demand in Iranian agriculture sector. They used GA to optimize predictor equation parametric values and ANN to

S. A. Pourmouavi Kani is with the Department of Electrical Engineering, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran (purmousavi @aut.ac.ir).

Nima Farrokhzad Ershad is with Department of Electrical Engineering, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran (e-mail: nima_f_ershad@yahoo.com).

forecast each of independent economic indicators [8]. As we had access to the data used in [8], we adopted Iranian agriculture sector as a case study to compare our results with results in [8].

The economic indicators used in this paper are price, value added, number of customers and consumption in the previous periods. First three economic indicators have been used as variables in model. These parameters are selected because they have great effects on electricity demand fluctuations in this sector. This model can be used to predict electricity demand in the future by optimizing parameter values. The rate of changes in these parameters from 1981 to 2005 is similar with that in the electricity consumption trend.

To predict the future values, a linear-logarithmic model of electrical energy demand is considered. The PSO algorithm applied in this study has been tuned for all its parameters and the best coefficients with minimum error are identified, while all parameter values are tested concurrently. Consumption in the previous periods has been used for testing estimated model. In addition, ANN is used to forecast each independent variable and then electricity consumption is forecasted up to year 2010. We used electricity consumption in Iranian agriculture sector from 1981 up to 2005 to find fine-tuned model and test its performance in prediction. Besides, values for each independent variable includes price, value added, and number of customers from 1981 up to 2005 are available. The latest data is used for prediction each of independent variable. Fig. 1, shows electricity consumption for Iranian agriculture sector from 1981 up to 2005 [17].

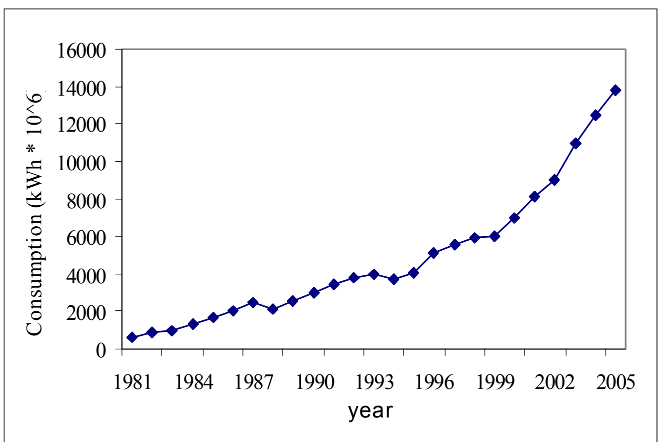

Fig. 1. Electricity consumption for Iranian agriculture sector from 1981 to 2005.

II. 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 [11]. One of the most powerful swarm intelligence-based optimization techniques, named Particle Swarm Optimization (PSO), was introduced by Kennedy and Eberhart [9, 10]. 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 [9-14]. 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 [12], maximum velocity limit, parallel optimization [13], deflection, repulsion, stretching [11], mutation [14] 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 [11]. 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 an n-dimensional vector, and velocity $V_i=[v_{i1},v_{i2},...,v_{in}]$, 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 $P_i=[p_{i1},p_{i2},...,p_{in}]^T$, and the best position of the best particle in the swarm is referred as $G=[g_1,g_2,...,g_n]^T$ [15].

Each particle i adjusts its position in next iteration $t+1$ with respect to Eqs. (1) and (2) [11]:

$$V_i(t+1)=\omega(t)V_i(t)+c_1r_1(P_i(t)-X_i(t+1)) + c_2r_2(G_i(t)-X_i(t+1)) \tag{1}$$

$$X_i(t+1)=X_i(t)+\chi V_i(t+1) \tag{2}$$

Each variables of Eq. (1) is described as follows:

A. Parameter selection

There are several guidelines about the selection of the key parameters in PSO as available from the literature [9-14].

B. The number of particles

The typical range for the number of particles is 20–40. For most of the problems, 10 particles are large enough to get good results. For some difficult or special problems, 100–200 particles can be tried as well. In this work, a particle size of 60 is chosen which gives results close to the optimal.

C. Range of the particles

The range of the particles depends on the problem to be optimized. One can specify different ranges for different dimension of the particles. In this work, the range is depended on the problem and change between -5 to 5 time ranges.

D. Maximum velocity v_{max}

The maximum velocity v_{max} determines the maximum change one particle can take during one iteration. Usually, the range of the particle is set as v_{max}. In this work, a $v_{max} = (4 \times \textit{Number of Decision Variables})$ is chosen for each particle as this gives better optimal results.

E. The inertia parameter

The inertia parameter is introduced by Shi and Eberhart [16] and provides improved performance in a number of applications. It has control over the impact of the previous history of velocities on current velocity and influences the balance between global and local exploration abilities of the particles. A larger inertia weight favors a global optimization and a smaller inertia weight favors a local optimization.

In Ref. [16], it is suggested to range $\omega(t)$ in a decreasing way from 1.4 to 0 adaptively. In this work, inertia parameter modify linearly from 1 to 0.05 as it facilitates reaching a better optimal value in lesser number of iterations.

The inertia coefficient in Eq. (1) is employed to manipulate the impact of the previous history of velocities on the current velocity. Therefore, $\omega(t)$ 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 [11].

F. The parameters c_1 and c_2

The acceleration constants c_1 and c_2 indicate the stochastic acceleration terms which pull each particle towards the best position attained by the particle or the best position attained by the swarm. Low values of c_1 and c_2 allow the particles to wander far away from the optimum regions before being tugged back, while the high values pull the particles toward the optimum or make the particles to pass through the optimum abruptly. In Ref. [9], the constants c_1 and c_2 are chosen equal to 2 corresponding to the optimal value for the problem studied. In the same reference, it is mentioned that the choice of these constants is problem dependent. In this work, $c_1 = 2$ and $c_2 = 2$ are chosen which give better optimal results in lesser iterations.

G. The stop condition

The stopping criterion can be adopted as the number of iterations the PSO algorithm execute and the minimum error requirement. In this work, after about 200 iterations the improvement in the objective function is not significant and this value is taken as the maximum number of iterations the algorithm can execute.

With this algorithm, we can find best values of predictor model coefficients as described in section III.

III. Linear-Logarithmic Model

As mentioned before, in this study a linear–logarithmic model of electrical energy demand is considered in the agricultural sector as different kind of mathematical equations can be estimated by PSO algorithm. Eq. (3) presents the linear–logarithmic model:

$$y = a_\circ + a_1.\ln x_1 + a_2.\ln x_2 + a_3.\ln x_3 \tag{3}$$

where x_1, x_2 and x_3 are price, value added and number of customers respectively and y is the electricity consumption for the next year. a_o, …, a_4 are the Linear-Logarithmic Model coefficients that must be tuned with PSO algorithm. The usual way of estimating model parameters is to use data partially; with values of all dependant and independent variables (includes price, value added, number of customers and electricity consumption in this sector) from 1981 to 2005, one series (at least 20 years) is to estimate the parameters and saving the reminders for testing purpose. The testing procedure is to obtain minimum relative error between estimated and actual values.

To introduce fitness function for PSO algorithm, the variables should be put in the model and then the difference between estimated values and actual data for each individual should be calculated. This function is named Mean Absolute Percentage Error (MAPE) and is commonly used in PSO algorithm applications. In each generation the particles with minimum difference must be returned. The fitness function is shown below:

$$\min f = 1/n\sum_{j=1}^{n}(D_{actual} - D_{estimated})/D_{actual} \tag{4}$$

where D_{actual} and $D_{estimated}$ are actual and estimated energy demand, respectively and n is the number of observations.

Twenty years data was used to estimate the model parameters and five years data is also saved for testing purpose. The required parameters on PSO algorithm are as follows:

- Population size (n): 60
- Iterations (number of the generation): 200
- Inertia coefficient ($\omega(t)$): change linearly from 1 to 0.05

After applying PSO on Eq. (3) the resulted model is as follows:

$$y = 0.1251 - 0.0616.\ln x_1 + 0.8115.\ln x_2 + 0.6402.\ln x_3 \tag{5}$$

In [8], Twenty-one years data was used to estimate the model parameters and five years data is also saved for testing purpose. To identify the best fitness, the required parameters on GA algorithm are as follows in [8]:

- Population size (n): 90
- Iterations (number of the generation): 200
- Mutation rate: 0.03
- Crossover rate: 84%

The obtained average relative error for the GA and regression model in [8] is 3.75% and 15.12%, respectively. Fig. 2 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. On the other hand, due to stochastically 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).

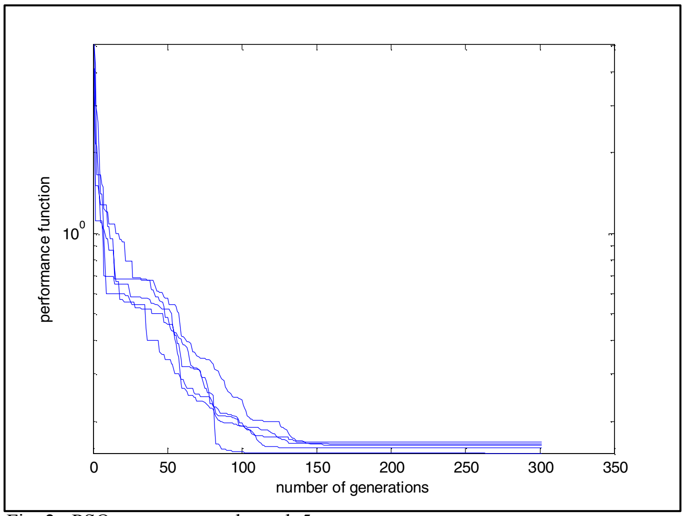

Fig. 2. PSO convergence through 5 runs.

The relative error between estimated and actual values with PSO algorithm is reported in Table I.

TABLE I
THE RELATIVE ERROR BETWEEN ESTIMATED DATA FROM PSO ALGORITHM AND ACTUAL DATA.

Year	Actual Data	PSO Algorithm	
		Estimated	Relative Error (%)
2001	11079	11349	2.44
2002	12435	12492	0.46
2003	13859	13997	1.00
2004	14526	14645	0.82
2005	14823	15126	2.05
Average Relative Error (%)			1.35

Fig. 3, shows electricity consumption for Iranian agriculture sector from 1981 up to 2005 in comparison with estimated data using linear-logarithmic model optimized with PSO.

As we can see, the average relative error for proposed method in this study is less than others. Furthermore, table I and Fig. 3. show the great performance of PSO algorithm for tuning of model parameters and electricity consumption prediction precisely.

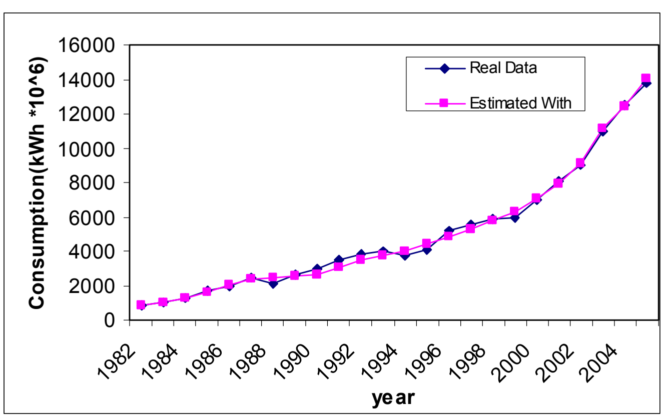

Fig. 3. Comparison between real electricity consumption in this sector and estimated with linear-logarithmic model that optimized with PSO.

IV. ARTIFICIAL NEURAL NETWORKS

The MLP neural network consists of simple processing elements (artificial neurons) arranged in layers: an input layer receiving the input variables, one or more hidden layers performing the required non linear input–output mappings, and an output layer producing the network outputs. Each neuron receives weighted inputs from all neurons in the preceding layer. Let W_{ij} be the weight associated with the link from neuron i in one layer to neuron j in the next downstream layer.

The neuron sums all weighted inputs and, with reference to a threshold value, uses a non-linear activation function to determine its output. The modeling problem is solved by training on a set of solved examples in the form of input–output records. Training attempts to minimize the error between known and calculated network outputs over all training examples through optimizing the network weights. The mean square error (MSE) criterion is given by:

$$E = \frac{1}{2}\left[\sum_{p}\sum_{p}\left|t_{kp} - O_{kp}\right|^2\right] \tag{6}$$

where t_{kp} and O_{kp} are the true and observed outputs, respectively, for neuron k in the output layer when input vector x_p corresponding to the p_{th} training record is applied to the network. Training with the back propagation algorithm involves iterative application of the training records, determining observed output errors for neurons in the output layer, back propagating these errors to all previous layers, and adjusting the weights so as to minimize the error criterion. The output from neuron j in a given layer (other than the input layer) is calculated as:

$$O_j = f\left(\sum_i W_{ij}.O_i\right) \tag{7}$$

where i indicates a neuron in the preceding layer and f is the activation function for neuron j. The activation function is often a sigmoid function of the form:

$$f(x) = \frac{1}{1+e^{-x}} \tag{8}$$

With the gradient descent approach to error minimization, weights are changed in proportion to the error gradient, i.e.

$$\Delta W_{ij} = -\eta\frac{\partial E}{\partial W_{ij}} \tag{9}$$

where η is a constant that determines the learning rate. To improve convergence characteristics, weight changes are also related to changes introduced in the previous iteration. At the nth iteration, the change in W_{ij} for the link from neuron i to neuron j is given by [19]:

$$\Delta W_{ij}(n) = \varepsilon\delta_j O_i + \alpha\Delta W_{ij}(n-1) \tag{10}$$

where ε is the learning rate, α is the momentum factor, and δ_j is the error signal for the destination neuron j. When neuron j is in the output layer, δ_j is given by:

$$\delta_j = (t_j - O_j)O_j.(1-O_j) \tag{11}$$

When neuron j is in a hidden layer, δ_j is given by:

$$\delta_i = O_j.(1-O_j)\sum_k \delta_k.W_{jk} \tag{12}$$

where k indicates neurons in the succeeding layer next to that containing neuron j.

The learning rate and the momentum factor influence the speed and stability of network training. The process continues until the error criterion on the training set is reduced below a specified limit. To improve generalization on new out-of-sample data, early stopping criteria are often employed where a separate test data set is used to validate the resulting model and training is stopped when error on that data set starts to increase indicating the start of overfitting.

In this paper, we used ANN for prediction of each independent variable. After that, we use estimated values for prediction of annual electricity consumption using (5). According to the fact that the available data are yearly, y_{t-1}, y_{t-2} and y_{t-3} are sufficient to justify y_t where y_t is the amount of consumption in t^{th} year. As the available data are for 25 years, totally 22 rows of data are developed according to this structure. For each independent variable, an ANN is constructed with similar topology to predict each of them up to year 2010. Table II shows the Mean Absolute Percentage Errors (MAPE) of prediction of each independent variable. Sensitivity analysis was done to find best ANN architecture. Different architecture of ANN with different number of layers and neurons in each layer is applied to find best architecture of ANN is calculated. The best architecture was a network with 3 layer perceptrons; 2 neurons in input layer, 3 neurons in hidden layer and 1 neuron in output layer.

This network is used for all three independent parameters that are needed for prediction of electricity demand up to year 2010.

TABLE II
THE MAPE ERROR FOR EACH INDEPENDENT VARIABLE CALCULATED OF ANN TEST.

Variables	Price	No of customers	Value added
MAPE error (%)	0.135	0.011	0.192

V. RESULTS

PSO algorithm is applied in section III on linear-logarithmic model and the best coefficients are obtained. Eq. (5) shows the final model that is used for prediction of electricity consumption up to year 2010.

On the other hand, as mentioned above, forecasting electricity demand by linear-logarithmic model optimized by PSO, needs to forecast the three independent variables separately. In this paper, it is done by ANN. All methodology is described in section IV and the results of each variable prediction up to year 2010 are reported in table III. Then we can use these values for prediction of electricity demand using (5).

TABLE III
PREDICTED VALUES OF PRICE, NUMBER OF CUSTOMERS AND VALUE ADDED UP TO YEAR 2010.

Prediction	Price (¢/kWh)	No of customers	Value added
2006	5.86	7971	112.5
2007	5.70	7805	119.6
2008	5.63	7486	127.1
2009	5.42	7362	130.7
2010	5.20	7154	139.5

With independent variables prediction values and using (5), the table IV shows the prediction of electricity consumption in Iranian agricultural sector up to year 2010.

TABLE IV
PREDICTION OF ELECTRICITY CONSUMPTION FOR IRANIAN AGRICULTURAL SECTOR UP TO YEAR 2010.

Years	2006	2007	2008	2009	2010
Prediction (10^6*kWh)	15685	15912	16410	16783	17102

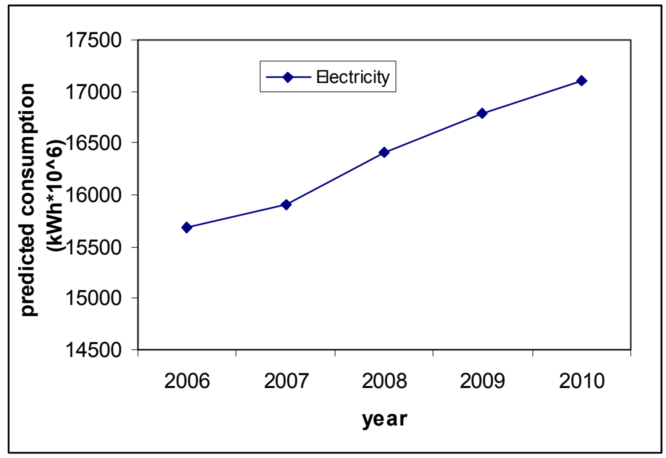

Fig. 4. Comparison between real electricity consumption in this sector and estimated with linear-logarithmic model that optimized with PSO.

Fig. 4. depicts forecasted values for electricity consumption in Iranian agriculture sector up to year 2010.

VI. CONCLUSION

In this study, a linear-logarithmic model for prediction of electricity consumption in Iranian agricultural sector up to year 2010 is applied. For tuning its coefficients, PSO algorithm is considered. In this case, the data of electricity consumption in this sector from 1981 to 2005 is used for optimizing linear-logarithmic model coefficients and test optimized model to prove its prediction ability. The results are compared with those presented in [8] as a similar case. The linear-logarithmic model includes three independent variables which have greatest effects on electricity consumption in this sector.

To prediction of demand in this sector, ANN is applied to predict each of these variables. Finally, electricity consumption up to year 2010 is calculated with optimized model.

Results show that PSO algorithm is much better than GA and regression method used in [8] for a similar case. This approach can be applied to other prediction problems.

VII. REFERENCES

[1] H. Ceylan, H. Ozturk, Estimating energy demand of Turkey based on economic indicators using genetic algorithm approach, Energy Conversion and Management 45 (2004) 2525–2537.
[2] H. Ozturk, H. Ceylan, O.E. Canyurt, A. Hepbasli, Electricity estimation using genetic algorithm approach: a case study of Turkey, Energy 30 (2003) 1003–1012.

[3] M.S. Osman, M.A. Abo Sinna, A.A. Mousa, A combined genetic algorithm-fuzzy logic controller (GA-FLC) in non-linear programming, Applied Mathematics and Computations (2005) 821–840.
[4] A. Tang, C. Quek, G. Ng, GA-TSKfnn: Parameters tuning of fuzzy neural network using genetic algorithms, Expert Systems with Applications 29 (2005) 769–781.
[5] D. Muni, N. Pal, J. Das, Genetic programming for simultaneous feature selection and classifier design, IEEE Transactions on Systems, Man and Cybernetics 36 (1) (2006).
[6] O.E. Canyurt, H. Ozturk, A. Hepbasli, Energy demand estimation based on two-different genetic algorithm approaches, Energy Source 26 (14) (2004) 1313–1320.
[7] S. Haldenbilen, H. Ceylan, Genetic algorithm approach to estimate transport energy demand in Turkey, Energy Policy (2004) 89–98.
[8] A. Azadeh, S.F. Ghaderi, S. Tarverdian, M. Saberi, Integration of artificial neural networks and genetic algorithm to predict electrical energy consumption, Applied Mathematics and Computation (2006).
[9] J. Kennedy and R. C. Eberhart, "Particle Swarm Optimization", in proc. IEEE int. Conf. Neural Networks, Perth, Australia, 1995, vol. IV, pp. 1942-1948.
[10] R. C. Eberhart and J. Kennedy, "A new Optimizer Using Particle Swarm Theory", in proc. 6th Symp. Micro Machine and Human Science, Nagoya, Japan, 1995, pp. 34-44.
[11] 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.
[12] 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.
[13] 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.
[14] Raymond R. Tan, "Hybrid evolutionary computation for the development of pollution prevention and control strategies", Journal of Cleaner Production 15 (2007) 902-906.
[15] B. Jaboui et al., "A combinatorial particle swarm optimization for solving …", Appl. Math. Comput. (2007), doi: 10.1016/j.amc.2007.04.096.
[16] Shi Y, Eberhart R. A modified particle swarm optimizer. In: Proceedings of the IEEE international conference on evolutionary computation, Anchorage. 1998. p. 69–73.
[17] Energy balances of Iran, Ministry of Energy, Iran, 2004, pp. 236–250.
[18] Artificial Neural Network, MATLAB ANN Toolbox Release 2007a.
[19] D.C. Park, M.A. El-Sharkawi, R.J. Marks II, L.E. Atlas, M.J. Damborg, Electric load forecasting using an artificial neural network, IEEE Trans. Power Syst. 6 (1991) 442–449.

VIII. BIOGRAPHIES

Seyyed Ali Pourmousavi Kani was born in Babol, Iran, on March 22, 1982. He received the B.Sc. degree from Mazandaran University, Mazandaran, Iran in 2005. He is currently studying at M.Sc. degree in Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran.

His research interests are Load prediction, Wind prediction, wind energy and its simulation.

Nima Farrokhzad Ershad was born in tehran, Iran, on june 12, 1982. He received the B.Sc. degree from Shahed University of tehran, Tehran, Iran, in 2005. He is currently studying at M.Sc. degree in Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran.

His research interests are Load prediction, fault detection, energy conservation.