International Science and Technology Journal
المجلة الدولية للعلوم والتقنية

Volume 36 العدد
Part 1 المجلد

المجلة الدولية للعلوم والتقنية
International Science and Technology Journal
ISTJ

http://www.doi.org/10.62341/hjas1651

Received	**2024/11/22**	**تم استلام الورقة العلمية في**
Accepted	**2024/12/20**	**تم قبول الورقة العلمية في**
Published	**2024/12/30**	**تم نشر الورقة العلمية في**

SOLVING THE THERMAL ECONOMIC DISPATCH PROBLEM USING ARTIFICIAL NEURAL NETWORKS

Hashem Jamil Alsalti
Electrical & Computer Engineering Department
Garabouli Engineering Faculty, Elmergib University- Libya
hjelsalti@elmergib.edu.ly

Abstract
The problem of economic dispatch (generation) is considered as one of the optimization problems, and many different methods have been used to solve it. In this study, the problem is solved using artificial neural networks (ANNs) which are considered as a branch of the artificial intelligence. The tool used in the study is the ANNs toolbox in MATLAB, which contains advanced capabilities. Through the toolbox, many options can be tested to reach the optimal solution to the problem. ANNs need to be trained on samples of inputs-outputs (solution) of problems before they can solve them on their own. During this study computer programs have been written to solve the thermal dispatch problem using a traditional mathematical method; the λ-iteration method. Through it, training samples are provided to the ANN. From the results of the study it has been proved that the solutions of the problem obtained using the trained ANN almost match with the solutions obtained using the λ-iteration method. The goal of using ANNs is to solve the thermal of economic dispatch problem in a faster time in the case of power systems that contain a large number of generating units.

Keywords: Thermal Economic Dispatch, Optimal Generation, Lambda Iteration, Multi-Layer Feed-Forward Artificial Neural Networks, Error back propagation algorithm.

http://www.doi.org/10.62341/hjas1651

حل مسألة الإرسال الإقتصادي الحراري باستخدام الشبكات العصبية الصناعية

هاشم جميل السلطي

قسم الهندسة الكهربائية والحاسوب، كلية الهندسة القره بوللي، جامعة المرقب – ليبيا
hjelsalti@elmergib.edu.ly

الملخص

تعتبر مسألة الإرسال (التوليد) الإقتصادي من مسائل الحل الأمثل، وقد تم استخدام العديد من الطرق المختلفة لحلها. في هذا البحث تم حل المسألة باستخدام الشبكات العصبية الصناعية (ANNs) والتي تعتبر من أفرع علم الذكاء الصناعي. الأداة التي استخدمت في البحث هي حزمة أدوات الشبكات العصبية الصناعية في الماتلاب والتي تحتوي على إمكانيات متطورة. يمكن من خلال الحزمة تجربة العديد من الخيارات المتعلقة بالشبكات العصبية الصناعية وصولا إلى الحل الأمثل للمسألة. تحتاج الشبكات العصبية الصناعية للتدريب على عينات من دخل–خرج (حلول) المسائل قبل أن تتمكن من حلها بنفسها. خلال هذا البحث تم كتابة برامج حاسوب لحل مسألة الإرسال الإقتصادي بطريقة رياضية تقليدية هي طريقة λ–التكرارية، ومن خلالها تم توفير عينات التدريب للشبكة العصبية الصناعية. من نتائج البحث تم أثبات أن حلول المسألة المتحصل عليها بواسطة الشبكة العصبية الصناعية المدربة تطابقت بشكل شبه تام مع النتائج المتحصل عليها بواسطة طريقة λ–التكرارية. الهدف من استخدام الشبكات العصبية الصناعية هو أن يتم حل مسألة الإرسال الإقتصادي في زمن أسرع في حالة منظومات القدرة التي تحتوي على عدد كبير من وحدات التوليد.

الكلمات المفتاحية: الإرسال الإقتصادي الحراري، التوليد الأمثل، λ–التكرارية، الشبكات العصبية الصناعية ذات التغذية الأمامية متعددة الطبقات، خوارزمية الإنتشار العكسي.

INTRODUCTION

The problem of economic dispatch arose from the time that two or more thermal (fossil-fired) generating units were committed (switched on and brought on-line) to supply a load. The problem was: how to distribute the load between the committed thermal units such that the operating fuel cost is minimum. Solving the economic dispatch problem could lead to saving of a considerable amount of

International Science and Technology Journal
المجلة الدولية للعلوم والتقنية
Volume 36 العدد
Part 1 المجلد
http://www.doi.org/10.62341/hjas1651

fuel cost. The economic dispatch can be considered as an optimization problem with the objective function as the operating fuel cost. To achieve this goal the operating hourly fuel cost ($/h) vs. net output power (MW) curve must be provided for all thermal units either by the manufacturers of the thermal generating units or by measurement. Different electric companies use different representations for this curve. Of these representations: quadratic, piecewise quadratic, and piecewise linear. The operating fuel cost could include the labor and maintenance costs. Traditionally, the quadratic representation is used in thermal economic dispatch computation [1]. In this study the quadratic operating fuel cost is used; Figure 1.

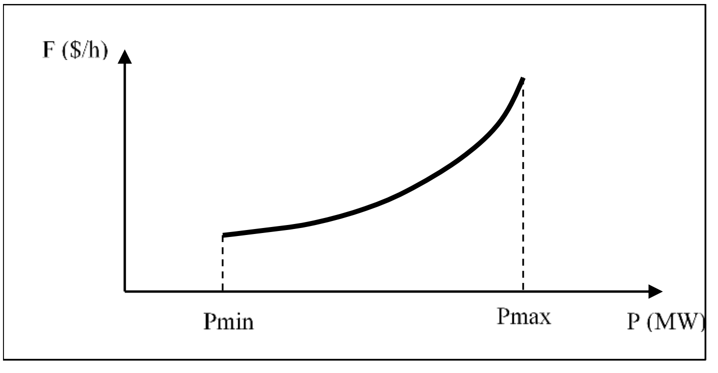

Figure 1 Quadratic hourly fuel cost.

There are many methods to solve the economic dispatch problem such as: gradient search, Newton's method, and λ-iteration method [2]. There have been many efforts to solve the problem using different types of ANNs; [3, 4]. In this study, the problem is solved using a common type of ANNs called feed-forward multi-layer ANNs.

METHODOLOGY OF THE STUDY

- Studying the economic dispatch problem with focus on thermal economic dispatch.
- Studying the different methods for solving the problem of thermal economic dispatch with focus on λ-iteration method.
- Refer to specialized references to choose an appropriate power system to be used as a test system.
- Writing computer programs to solve the thermal economic dispatch problem using the λ-method.

http://www.doi.org/10.62341/hjas1651

- Studying the ANNs with focus on its applications in electrical power systems.
- Studying the MATLAB's ANN toolbox and its application to solve the different problems.
- Construct an ANN and train it to solve the thermal economic dispatch with the results obtained using the written programs.
- Apply the trained ANN to solve the problem.
- Compare the results obtained using the ANNs with the results of the programs.

THERMAL ECONOMIC DISPATCH PROBLEM

The hourly operating fuel cost of thermal generating unit *i* as a function of the unit's net output is given by,

$$\boldsymbol{F_i(P_i) = a_i + b_i\,P_i + c_i\,P_i^2} \tag{1}$$

Where: F_i is the hourly fuel cost of unit *i* ($/h), P_i is the net output (MW), a_i is the constant coefficient of the cost function ($/h), b_i is the linear coefficient of the cost function ($/MWh), and c_i is the quadratic coefficient of the cost function ($/MW^{2}h).
The objective function to be minimized is,

$$\boldsymbol{F_T = \sum_{i=1}^{N_{gen}} F_i = \sum_{i=1}^{N_{gen}} (a_i + b_i\,P_i + c_i\,P_i^2)} \tag{2}$$

Subject to the following constraints:
- Global (system) constraint:

$$\boldsymbol{\sum_{i=1}^{N_{gen}} P_i = P_{load}} \tag{3}$$

- Local (unit) constraint:

$$\boldsymbol{P_{i,min} \le P_i \le P_{i,max}} \tag{4}$$

Where: F_T is the total hourly fuel cost ($/h), N_{gen} is the total number of committed units, P_{load} is the load, $P_{i,min}$ is the minimum allowable output of unit *i*, and $P_{i,max}$ is the maximum allowable output of unit *i*,
The system constraint assumes that all the committed units are connected to a single busbar to supply the system load; i.e., the network topology is ignored (Figure 2). The transmission losses are either neglected or incorporated into the load.

http://www.doi.org/10.62341/hjas1651

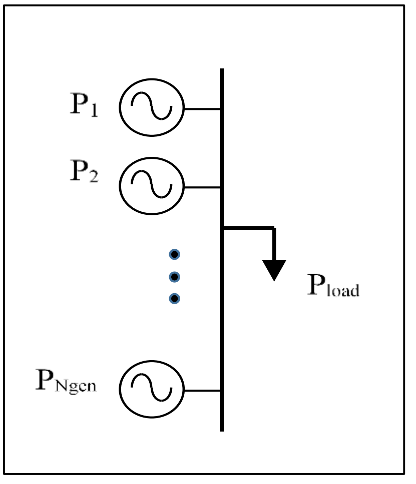

Figure 2 Thermal units supplying load directly.

Minimization of fuel cost can be achieved by applying calculus of optimization that includes the Lagrange function. Following the appropriate steps would lead to the following condition, which must objective function [2],

$$\frac{dF_i}{dP_i} = b_i + 2c_iP_i = \lambda \tag{5}$$

Where: $\frac{dF_i}{dP_i}$ is the increment hourly fuel cost (\$/MWh), and λ is the Lagrange multiplier (\$/MWh).
That is, the incremental hourly fuel costs of all committed units should be equal, at which the Lagrange multiplier is the optimal one ($\lambda_{optimal}$). Given a certain load, it is required to find $\lambda_{optimal}$.
The λ-iteration method is an effective way to find $\lambda_{optimal}$. In this method an initial value of λ is estimated from which the output power of all committed units is computed as follows,

$$P_i = \frac{\lambda - b_i}{2c_i} \tag{6}$$

If $P_i > P_{i,max}$ then P_i is set equal to $P_{i,max}$, and if $P_i < P_{i,min}$ then P_i is set equal to $P_{i,min}$.
Then λ is corrected depending on the error between the load and the sum of output of the committed units. The process is repeated until the error falls below a certain specified value.

http://www.doi.org/10.62341/hjas1651

To obtain a feasible solution using the λ-iteration method the load value must lie within the sum of limits of all the committed unit, that is $\sum_{i=1}^{N_{gen}} P_{i,min} \leq P_{load} \leq \sum_{i=1}^{N_{gen}} P_{i,max}$, because if $\sum_{i=1}^{N_{gen}} P_{i,min} > P_{load}$ then some units have to be switched off, and if $\sum_{i=1}^{N_{gen}} P_{i,max} < P_{load}$ then more units have to be brought on-line. The process of correcting λ from iteration to iteration can be carried on by different methods such as interpolation and binary search. In this study binary search method is used. A good estimation of λ is important to ensure convergence of solution.

ARTIFICIAL NEURAL NETWORKS (ANNs)

ANNs have found many applications in different areas including electrical power systems. ANNS are inspired by the biological neural networks. ANN is a computational model used to solve linear and nonlinear problems by training rather by equations. Therefore sets of input-output data samples must be provided to train the ANN. During the training process the ANN generalizes (understands) the relation between the input and output. Once an ANN has been trained successfully it would be able to find the output for an input without knowing the mathematical relation between them.

In this study a common type of ANNs called feed-forward multi-layer ANN is used [5]. A multi-layer ANN consists of input, hidden, and output layers. Each circle represents a processing units and is called a neuron; Figure 3.

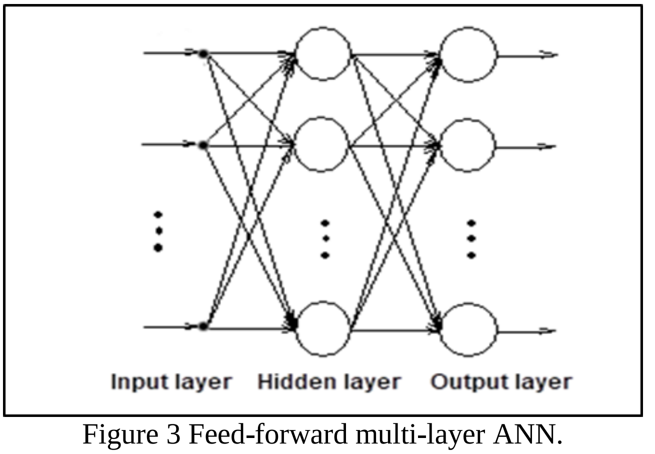

Figure 3 Feed-forward multi-layer ANN.

The input layer receives the input data and signals it to each neuron in the hidden layer through connections which have weights. A

http://www.doi.org/10.62341/hjas1651

neuron in the hidden layer sums the signals multiplied by weights, process the sum (applies an activation function on it), and sends the resulting signal through weighted connections to each neuron in the output layer. Similarly each output neuron sums the received signals multiplied by the weights, and applies an activation function on the sum. The output neurons send the results as output data corresponding to the input data.

In addition to receiving signals from all neurons from the previous layer, each neuron in the hidden and output layer receives additional signal from a separate weighted connection multiplied by 1. This connection is called bias. Figure 4 shows a neuron in the hidden layer.

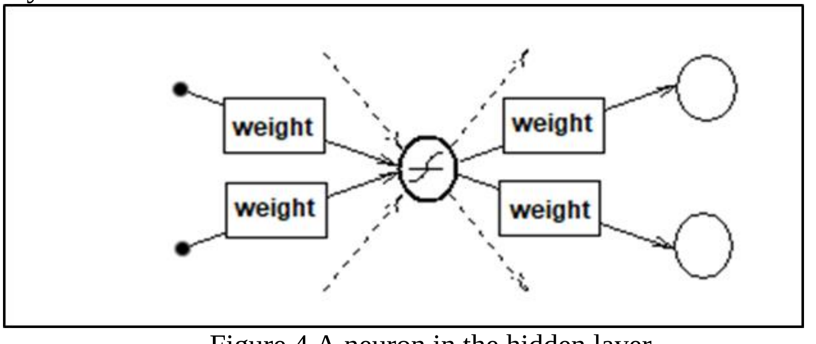

Figure 4 A neuron in the hidden layer.

The number of hidden layers in an ANN is either 0 (none), 1 layer, or many layers. ANNs without hidden layers can solve limited types of problems. Most common ANNs have a single hidden layer. ANNs with more than one hidden layer requires a longer training time, yet the success of the training process is not guaranteed.

There are many types of activation functions, such as log sigmoid $f(x) = \frac{1}{1+ e^{-x}}$, and hyperbolic tangent sigmoid $f(x) = \tanh(x)$; Figure 5. Different activation functions suit specific problems. Usually the activation function in the output layer is simply the linear function $f(x) = x$.

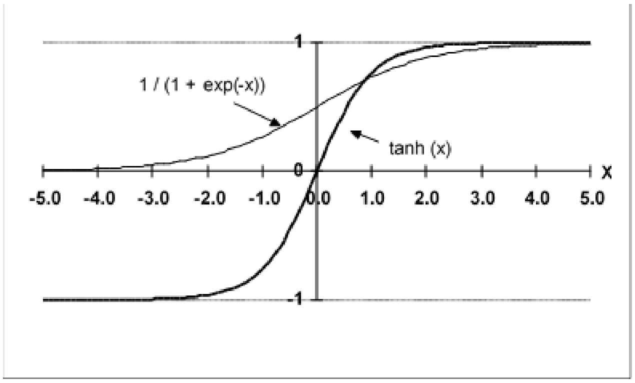

Figure 5 Sigmoid-type functions.

To train an ANN a set of input-output data pairs (called samples) must be available, either by mathematical methods or gathered statistical data. In the beginning, the weights of all connections (including biases) of an ANN are given random values; so when an input data (with known output) is given to the ANN the output would be completely erroneous compared to the actual output. The Mean Squared Error (MSE) will be used to update (correct) the weights using a method called error back-propagation [5]. MSE is the average squared difference between ANN's output and actual (target) outputs. The data is shown to the ANN repeatedly (iterations) and each time the weight values are updated depending on the error.

The ANN would be considered as successfully trained when the error becomes less than a pre-specified value. It is said that the ANN generalizes the relation between input and output. A trained ANN is supposed to give the correct output for an input with unknown (unsolved) output. This input must belong to the same pattern of the training data.

MATLAB ARTIFICIAL NEURAL NETWORKS TOOLBOX

MATLAB provides an advanced ANNs toolbox with many options; such as: network type, training function, adaptation training function, performance function, number of hidden layers, number of neurons in each hidden layer, transfer functions, etc.

During the training process, the training samples are divided into three groups:

- Training samples: presented to the ANN during training and the ANN is adjusted according to its error.

http://www.doi.org/10.62341/hjas1651

- Validation samples: used to measure ANN generalization, and to halt training when generalization stops improving.
- Testing samples: have no effect on training and so provide an independent measure of ANN performance during and after training.

The training group consists of 70% of the training samples. The percentage for validation and testing group can optionally be set between 5% and 35% of the training samples.

During the training process, many measures are shown to demonstrate the progress of the process. Of the important measures are the MSE (also referred to as performance) and regression. Regression R values measure the correlation between ANNs' outputs and actual (target) outputs. An R value of 1 means a close relation, 0 a random relation. After the finish of the training, many kinds of curves can be generated to show the training progress in details; that is, iteration by iteration (called epoch in MATLAB terminology).

Once the training process is completed successfully, the resulting ANN can be saved and used anytime to obtain a correct output corresponding to given input with unknown output.

TEST RESULTS AND DISCUSION

The 26 thermal generating units system of Reference [6] is used in the case study; Table 1.

Table 1 Generating units' data [6]

Unit	P max (MW)	P min (MW)	a ($/h)	b ($/MWh)	C ($/MW^{2}h)
1	12	2.4	24.3891	25.5472	0.02533
2	12	2.4	24.4110	25.6753	0.02649
3	12	2.4	24.6382	25.8027	0.02801
4	12	2.4	24.7605	25.9318	0.02842
5	12	2.4	24.8882	26.0611	0.02855
6	20	4	117.7551	37.5510	0.01199
7	20	4	118.1083	37.6637	0.01261
8	20	4	118.4576	37.7770	0.01359
9	20	4	118.8206	37.8896	0.01433
10	76	15.2	81.1364	13.3272	0.00876
11	76	15.2	81.2980	13.3538	0.00895
12	76	15.2	81.4641	13.3805	0.00910
13	76	15.2	81.6259	13.4073	0.00932
14	100	25	217.8952	18.0000	0.00623
15	100	25	218.3350	18.1000	0.00612
16	100	25	218.7752	18.2000	0.00598

http://www.doi.org/10.62341/hjas1651

17	155	54.25	142.7348	10.694	0.00463
18	155	54.25	143.0288	10.7154	0.00473
19	155	54.25	143.3179	10.7367	0.00481
20	155	54.25	143.5972	10.7583	0.00487
21	197	68.95	259.1310	23.0000	0.00259
22	197	68.95	259.6490	23.1000	0.00260
23	197	68.95	260.1760	23.2000	0.00263
24	350	140	177.0575	10.8616	0.00153
25	400	100	310.0021	7.4921	0.00194
26	400	100	311.9102	7.5031	0.00195

For this system the sum of minimum allowable generation $\sum_{i=1}^{26} P_{i,min} = 927.65$ MW and the sum of maximum allowable generation $\sum_{i=1}^{26} P_{i,max} = 3105$ MW. During the study, it is assumed that all loads lie between these two values, permitting to assume that all the units are committed (on-line). It is required to find the economic (optimal) generation of all units corresponding to a given load using ANNs. The implementation is achieved using MATLAB's ANN toolbox.
To implements ANNs it is necessary to provide a set of training data; so computer programs have been written to solve the economic dispatch problem using the traditional λ-iteration method with binary search. Using the programs a total of 100 training samples have been obtained with random load values. The input of each sample is the load value, and the output is the corresponding optimal generation of the 26 units. Figure 6 shows the constructed ANN.

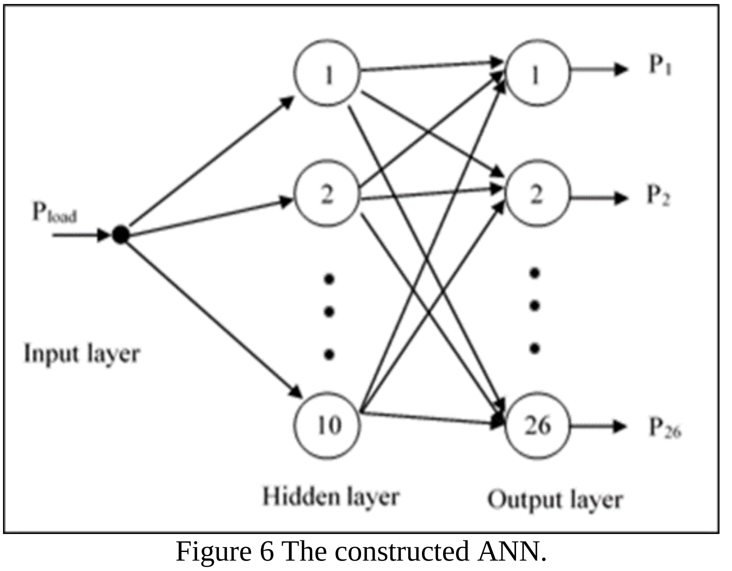

Figure 6 The constructed ANN.

International Science and Technology Journal
المجلة الدولية للعلوم والتقنية
Volume 36 العدد
Part 1 المجلد
المجلة الدولية للعلوم والتقنية
International Science and Technology Journal
ISTJ
http://www.doi.org/10.62341/hjas1651

Some of the parameters used during the training process are: a single hidden layer with 10 neurons, and Levenberg-Marquardt training algorithm.

Figure 7 shows the mean squared error (MSE) after the completion of the training process and Figure 8 shows the progress of the learning process through the 16 iterations.

Results

	Samples	MSE	R
Training:	70	1.07075e-0	9.99956e-1
Validation:	15	1.87872e-0	9.99915e-1
Testing:	15	2.48826e-0	9.99864e-1

Figure 7 ANN training results.

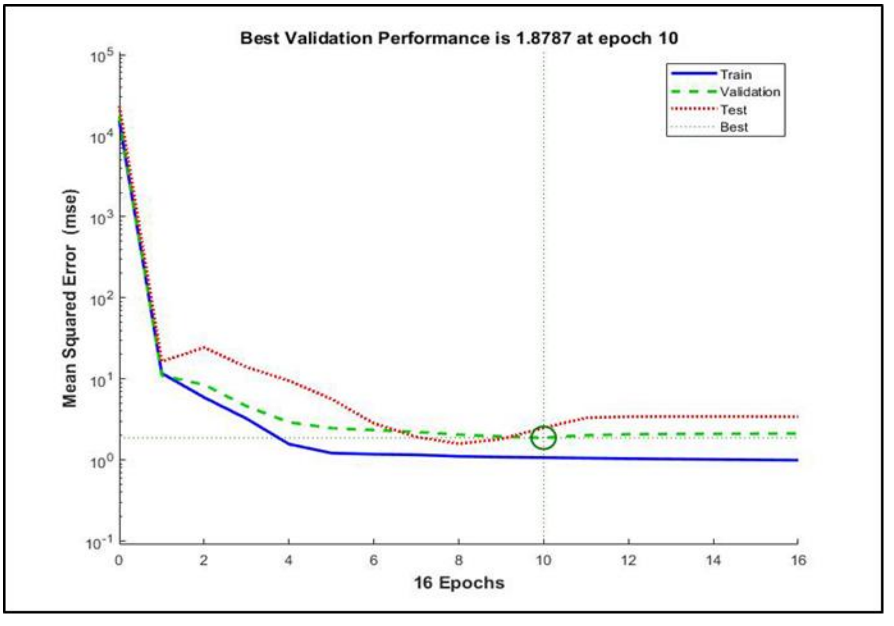

Figure 8 Progress of the training process.

The mean squared error (MSE) starts at a high value of $1.55*10^4$ and terminates at a low value of 1.07075 in 16 iterations (epochs). The regression (R) terminates at a value of approximately 1 which means a very close relation between the constructed ANN's output and actual (target) output.

http://www.doi.org/10.62341/hjas1651

The next step is to use the resulting trained ANN to obtain the optimal generation of the 26 units corresponding to certain loads. Three loads have been used: the first load is near sum of minimum allowable generation of all units (= 1200 MW), the second load is in the middle between sum of minimum- and maximum allowable generation of all units (= 2016 MW), and the last load is near sum of maximum allowable generation of all units (= 3000 MW).
The results are shown in Table 2. Units' outputs in MW.

Table 2 Results of the test system

	P_{load} = 1200 MW		P_{load} = 2016 MW		P_{load} = 3000 MW	
	ANN	λ-iteration	ANN	λ-iteration	ANN	λ-iteration
P_1	2.37	2.40	2.38	2.40	9.83	7.26
P_2	2.34	2.40	2.38	2.40	8.21	4.53
P_3	2.39	2.40	2.39	2.40	6.14	2.40
P_4	2.38	2.40	2.39	2.40	4.26	2.40
P_5	2.36	2.40	2.39	2.40	2.84	2.40
P_6	4.00	4.00	4.00	4.00	4.00	4.00
P_7	4.00	4.00	4.00	4.00	4.00	4.00
P_8	4.00	4.00	4.00	4.00	4.00	4.00
P_9	4.00	4.00	4.00	4.00	4.00	4.00
P_{10}	15.19	15.20	14.96	15.20	75.96	76.00
P_{11}	15.20	15.20	15.09	15.20	75.98	76.00
P_{12}	15.20	15.20	15.22	15.20	75.99	76.00
P_{13}	15.20	15.20	15.35	15.20	76.00	76.00
P_{14}	25.00	25.00	25.09	25.00	100.16	100.00
P_{15}	25.00	25.00	24.94	25.00	100.05	100.00
P_{16}	25.00	25.00	24.78	25.00	99.93	100.00
P_{17}	54.33	54.25	132.81	132.03	155.01	155.00
P_{18}	54.28	54.25	127.92	126.98	155.01	155.00
P_{19}	54.23	54.25	123.54	122.65	155.00	155.00
P_{20}	54.19	54.25	119.78	118.92	155.00	155.00
P_{21}	68.95	68.95	68.95	68.95	196.71	197.00
P_{22}	68.95	68.95	69.02	68.95	194.91	197.00
P_{23}	68.95	68.95	69.05	68.95	190.85	197.00
P_{24}	139.88	140.00	340.99	344.77	350.00	350.00
P_{25}	237.07	238.19	400.47	400.00	400.00	400.00
P_{26}	233.12	234.15	400.29	400.00	400.00	400.00
$\sum_{i=1}^{26} P_i$	1197.60	1199.99	2016.19	2016.01	3003.84	2999.99
F_T ($/h)	19317	19340	27862	27862	46496	46381

The outputs obtained using the trained ANN are very close to the outputs obtained using the λ-iteration method. The total hourly fuel cost values are almost equal for both methods. The percentage

International Science and Technology Journal
المجلة الدولية للعلوم والتقنية
Volume 36 العدد
Part 1 المجلد
المجلة الدولية للعلوم والتقنية
International Science and Technology Journal
ISTJ
http://www.doi.org/10.62341/hjas1651

relative error between ANN's result and λ-iteration's result (reference) for each generating unit is given in Table 3.

Table 3 Percentage relative error for ANN's outputs

	P_{load} = 1200 MW	P_{load} = 2016 MW	P_{load} = 3000 MW
P_1	-1.13	-0.66	35.33
P_2	-2.42	-0.69	81.27
P_3	-0.41	-0.33	155.97
P_4	-0.82	-0.24	77.64
P_5	-1.66	-0.22	18.46
P_6	0.00	0.00	0.00
P_7	0.00	0.00	0.00
P_8	0.00	0.00	0.00
P_9	0.00	0.00	0.00
P_{10}	-0.04	-1.60	-0.05
P_{11}	-0.02	-0.70	-0.03
P_{12}	0.00	0.11	-0.02
P_{13}	0.02	0.98	0.00
P_{14}	0.01	0.36	0.16
P_{15}	-0.01	-0.26	0.05
P_{16}	-0.02	-0.87	-0.07
P_{17}	0.15	0.59	0.01
P_{18}	0.05	0.74	0.01
P_{19}	-0.03	0.72	0.00
P_{20}	-0.10	0.72	0.00
P_{21}	0.00	0.00	-0.15
P_{22}	0.00	0.10	-1.06
P_{23}	0.00	0.15	-3.12
P_{24}	-0.08	-1.10	0.00
P_{25}	-0.47	0.12	0.00
P_{26}	-0.44	0.07	0.00

In most cases the percentage relative error is less than 3.5%. The exception is in the case of highest load of 2000 MW where the high error values result from units 1-5. These units have the lowest maximum allowable output values; so their influence on supplying load is minimum and can be compensated by other units.
The percentage relative error between sum of all units' output and load (reference) for the ANN method is shown in Table 4.

Table 4 Percentage relative error between $\sum_{i=1}^{26} P_i$ and P_{load}

	P_{load} = 1200 MW	P_{load} = 2016 MW	P_{load} = 3000 MW
% Relative error	-0.2000000	0.0094246	0.1280000

http://www.doi.org/10.62341/hjas1651

The error values for the three loads prove that the constructed ANN has successfully determined the necessary outputs for the given loads.

CONCLUSION

In this study, the thermal economic dispatch problem has been solved using a trained ANN. The training data has been provided using developed computer programs that depend on a traditional method; the λ-iteration method.

The results show that the thermal generating units' outputs obtained using the trained ANN are very close to the outputs obtained using the λ-iteration method. This proves the success of the training process and that the constructed ANN has successfully generalized (learned) the relation between the load and the corresponding economic outputs.

It is left for incoming studies to consider more operating constrains such as inclusion of the transmission losses and spinning reserve requirement. Of important issue is to consider shutting down some of the units in case that the sum of the minimum allowable outputs of the remaining committed units is sufficient to supply the load.

Also the effect of changing some of the training parameters of the ANN on the learning process is to be considered in the incoming studies. Examples of these parameters are: number of hidden layers, number of neurons in each hidden layers, activation functions, and learning algorithm.

REFERENCES

[1] Bhardwaj, Amit, "Literature Review of Economic Load Dispatch Problem in Electrical Power System using Modern Soft Computing," International Conference on Advanced Studies in Engineering and Science, pp. 922-931, December 2017.
[2] Wood, A.J., and Wollenberg, B.F., "Power generation Operation and Control," 2nd edition, John Wiley & Sons, 1996.
[3] Su, C., and Lin, C., "New Approach with a Hopfield Modeling Framework to Economic Dispatch," IEEE Transactions on Power Systems, Vol. 15, No.2, May 2000, pp. 541-545.
[4] Mohammadi, A., and Varahram, M. H., "Using Neural Network for Solving of On-Line Economic Dispatch Problem." 2006 International Conference on Computational Inteligence for Modelling Control and Automation and International Conference

http://www.doi.org/10.62341/hjas1651

on Intelligent Agents Web Technologies and International Commerce, 2006 IEEE.
[5] Fausett, L., "Fundamentals of Neural Networks Architecture Algorithms and Applications," Prentice Hall International, Inc., Englewood Cliffs, New Jersy, 1994.
[6] Wang, S.J., Shahidehpour, S.M., Kirschen, D.S., Mokhtari, S., and Irisarri, G.D., "Short-Term generation Scheduling With Transmission and Environmental Constraints Using an Augmented Lagrangian Relaxation," IEEE Transactions on Power Systems, Vol. 10, No. 3, August 1995, pp. 1294-1301.