Modeling of Induction Heating Systems Using Artificial Neural Networks

Prof. Dr. A.K.M. AL-Shaikhli* Assist. Prof. Dr. Hanan A. R. Akkar*
Lec. Dr. Mohammed Moanes Ezzaldean Ali*
Received on:3/6/2009
Accepted on:7/4/2010

Abstract

Induction heating system has a number of inherent benefits compared to traditional heating systems. Many analytical and numerical approaches have been applied to solve the problem of induction heating. Artificial Neural Networks possess many advantages and have the ability to tackle problems that cannot be accomplished by more analytical and numerical methods. This paper involves modeling many artificial neural networks, and training them based on the results of analysis induction heating systems, by using ANSYS package, to enable them to evaluate the heat distribution inside the workpiece of any induction heating system. Also neural networks are used to specify the time and the power supply required for any desired heat distribution inside the workpiece. The neural networks are simulated by using Neural Network Toolbox in MATLAB, and the networks are trained according to supervised scaled conjugate gradient algorithm until the performance function (mean square error) reach the goal ($=10^{-4}$). Artificial Neural Networks show a good success in solving the problem of induction heating through obtaining results with high accuracy and very short run time.

Keywords: Induction heating, neural network, ANSYS, Scaled Conjugate Gradient Algorithm

الخلاصه

منظومات التسخين الحثي تمتلك عده ميزات مقارنة بمنظومات التسخين التقليديه. ميزات التسخين الحثي جعلته يشكل الحصه الرئيسيه من سوق معدات التسخين و الصهر. عدد كبير من الطرق التحليليه والعدديه استخدمت لتحليل و تصميم منظومات التسخين الحثي. الشبكات العصبيه الاصطناعيه لها عده فوائد فهي قادره على معالجه مشاكل لا يمكن لاكثر الطرق التحليليه و العدديه التعامل معها. التعامل مع الافران الحثيه في هذه الدراسه اعتمد على الفوائد التي تقدمها الشبكات العصبيه. في هذا البحث تم تصميم عده شبكات عصبيه و تدريبها اعتمادا على نتائج تحليل الافران الحثيه , باستخدام برنامج التحليل ANSYS , لتقوم الشبكات العصبيه باستنتاج التوزيع الحراري داخل قطعه الشغل لاي منظومه تسخين حثي و كذلك يمكن للشبكات العصبيه تحديد الزمن و متطلبات مجهز القدره اللازمين للحصول على التوزيع الحراري المطلوب داخل قطعه الشغل. في هذا البحث تم تصميم وتمثيل الشبكات العصبيه الاصطناعيه باستخدام صندوق ادوات الشبكه العصبيه ضمن برنامج MATLAB , اما تدريبها فقد تم باستخدام خوارزمية تدريب الميل المرافق المتدرج (Scaled conjugate gradient training algorithm) حيث استمرت عمليات التدريب لغايه هبوط داله الاداء (متوسط مربع الخطأ) للهدف المحدد(= 10^{-4}). و الشبكات العصبيه الاصطناعيه اثبتت نجاحها في معالجه مشاكل التسخين الحثي من خلال الحصول على نتائج ذات دقه عاليه و بزمن تنفيذ قصير جدا.

* Department of Electrical and Electronic Engineering- University of Technology – Baghdad

1 Introduction

The basic electromagnetic phenomena of induction heating are quite simple. An alternating voltage applied to an induction coil (e.g. solenoid coil) will result in an alternating current in the coil circuit. An alternating coil current will produce, in its surrounding, a time-variable magnetic field. This magnetic field induces eddy currents in the workpiece located inside the coil. These induced currents have the same frequency as the coil current but their direction is opposite to the coil current. These currents produce heat by the Joule effect [1]. The induction coil is always constructed of single layer multi turn winding. The range of application of induction heating at mains frequency can be increased if a multi-layered coil is used instead of the conventional single layer winding [2].

2 Analysis of Induction Heating Furnaces by ANSYS Package

Employment of ANSYS package to simulate and analyze induction heating systems has demonstrated great advantages. A full description of the mathematical model used to simulate the induction heating systems under ANSYS environment is published in reference [3]. Despite the several advantages of this model, the following imperfections have to be pointed out:

1- Analyzing one furnace requires relatively long time, and the analysis of a multilayer induction furnace with more layers requires more time. For example, the running time for one case takes about 78 minutes [4].
2- There is no ability to take advantage of previous results to deduce a result of a new case even if there are many close and solved cases.
3- ANSYS package provides only the forward solver (i.e. power density or temperature distribution inside workpiece etc.) and it is obtained when the values of frequency, induction current density and the heating time are available. ANSYS cannot be rearranged to estimate an inverse solver i.e. frequency, current density and the time cannot be obtained although the heat distribution is available [4].

These imperfections can be overcome by using artificial neural networks. Artificial Neural Networks (ANN) are computational models that share some of the properties of the brain. These networks consist of many simple units, called neurons, working in parallel with no central control [5]. One of the various applications of the neural network is large scale function approximation where the neural network is employed to construct the function that generates approximately the same output for a given vector based on the available training data [6].

In this paper, the neural networks are modeled and simulated by using Neural Network Toolbox in MATLAB (R2007a). The networks are trained according to supervised scaled conjugate algorithm.

In most of training algorithms a learning rate is used to determine the length of the weight update (step size) but in the Conjugate Gradient Algorithms a search is made along the conjugate gradient direction to determine the step size which minimizes the error function along that line. Each of the Conjugate Gradient Algorithms requires a line search at each iteration, and that is computationally expensive. The Scaled Conjugate Gradient (SCG) Algorithm was designed to avoid the time consuming line search [7].

The simulations of the single layer induction heater include 64 cases and these cases are listed in table (1), eight different frequencies (1st column on the left) were used and for each frequency, there are eight power supply levels (current densities). The simulations of multilayer induction heater include 64 cases and these cases are listed in table (2), eight different numbers of layers (1st column on left) were used and for each number of layers, there are eight power supply levels (current densities). By simulating any induction furnace, a table of heat distribution can be obtained at any time during the heating cycle. For each one of the cases tabulated in tables (1) and (2), eight tables were prepared; these tables display the heat distribution of the workpiece at every 30 seconds. The total number of tables obtained is 512 tables for single layer furnaces and 512 tables for multilayer furnaces. Since the heat distribution is symmetrical in the upper and lower parts of the workpiece, it is enough to tabulate only the temperatures in the lower half of the workpiece (400 elements).

3 Neural Networks for Forward Solver

3.1 Modeling of the Forward Solver Neural Network

The forward solver can be defined as a solution method that has the heat distribution as an output and the following items as an input: heating cycle period, induction current density and one of the frequencies or number of layers for the single or multilayer furnace, respectively. By training an appropriate neural network on data obtained by ANSYS package, a forward solver can be achieved. The best model of the neural network is reached by trial and error. These attempts include changing the parameters and observing the performance of the network. The final model of the neural network acting as a forward solver involves the parameter listed in table (3), and a schematic diagram of the neural network is shown in figure (1).

3.2 Training the Forward Solver Neural Networks

Before training the network, the weights and the biases must be initialized. Random initial values within range of (-1,1) are used. Once the network has been initialized, it is ready for training. First , the network was trained on single layer induction furnace cases with 512 sets of examples, and these sets are arranged as input and target matrices, the input matrix is of 3×512 size and the target matrix is of 400×512 size.

During training, weights and biases of the network are iteratively adjusted until the performance function (MSE) drops below a certain threshold (goal), since the data size is very large, and the training of the neural network has to take a long time. A suitable technique is used depending on saving the values of the weights and biases after a specific number of epochs, and the stored values become the initial values of the weights and biases for the next run of the training processes. By using this technique and after a series of training processes, the goal is achieved. The parameters of training the neural network for single layer induction furnaces data are listed in table (4), and the training plot of the training process is shown in figure (2). For multilayer induction furnaces, the same neural network is used to be trained on the data of the multilayer furnaces, which are input matrix of 3×512 size and target matrix of 400×512 size. The training parameters are listed in table (5), and training plot of the training process is shown in figure (3).

4 Neural Networks for Inverse Solver:

4.1 Modeling of the Inverse Solver Neural Network

Inverse solver can be defined as a solution method that has the heat distribution through the workpiece as an input and the following items as an output: heating cycle period, induction current density and one of the frequencies or number of layers for the single or multilayer furnace, respectively [8]. Inverse solver can be achieved by training an artificial neural network on data obtained by ANSYS. The model of the network acting as an inverse solver involves the parameters listed in table(6), and a schematic diagram of the neural network shown in figure(4).

4.2 Training the Inverse Solver Neural Network

Weights and biases of the neural network are initiated as random values. The input matrix has a size of 400×512 and the target matrix has a size of 3×512. The parameters of the training process for the data of the single layer induction furnaces are listed in table(7), and the training plot of the training process is shown in figure(5). The parameters of training the network for the data of multilayer furnaces are listed in table(8), and the training plot of the training process is shown in figure(6).

5 Simulations of Neural Networks

The performance of a trained neural network can be measured to some extent by the Mean Square Error (MSE) on the training, but it is often useful to investigate the network response by performing regression analysis between the network output and the corresponding desired target. The following figures illustrate the graphical output of regression analysis. The network outputs are plotted versus the targets as open circles. The best linear fit is indicated by a solid line, and the perfect fit (outputs equal to targets) is indicated by a dashed line. Figure (7) shows regression analysis of forward solver neural network for single layer furnaces. Figure (8) shows regression analysis of forward solver neural network for multilayer furnaces. Figure (9) shows regression analysis of inverse solver neural network for single layer furnaces. Figure (10) shows regression analysis of inverse solver neural network for multilayer furnaces. In figures (7) to (10), it is difficult to distinguish the best linear fit line from the perfect fit line, because the fit is so good, which indicates that the neural networks have undergone adequate training, and they can perform good estimation with low error.

6 Results of the Neural Networks

6.1 Results of the Forward Solver Neural Networks

For single layer furnaces, by applying 25 kHz, 7.5A/mm^2 and 190 second as an input, the temperature distribution inside the workpiece obtained by ANSYS program is shown in figure (11a). Temperature distribution, obtained by neural network, is shown in figure (11b), and the percentage error is shown in figure (11c).The percentage errors for other cases are shown in figure (12). For multilayer induction furnaces, by applying 5A/mm^2, 6 layer induction coil and 150 second as an input, the temperature distribution inside the workpiece obtained by ANSYS program is shown in figure (13a), the temperature distribution obtained by neural network is shown in figure (13b), and the percentage error is shown in figure (13c). The percentage errors for other cases are shown in figure (14). From the results shown in figures (11) to (14), it is observed that the estimated values of the neural networks are in a good agreement with the reference values obtained by ANSYA program. Therefore, the forward solver neural networks can be used to obtain an accurate temperature distribution for any single or multilayer induction furnace with nil run time.

6.2 Results of the Inverse Solver Neural Networks

For the neural network of single layer induction furnaces, the temperature distribution shown in figure (15) is applied as an input and the values of the number of layers, frequency, current density and time are obtained. To verify the accuracy of the results of inverse solver neural network, the results are compared with reference values used by ANSYS program to obtain the temperature distribution of figure (15). The results of neural network, the reference values and the percentage error are listed in table (9). More cases of single layer induction furnaces are listed in table (10). For the neural network of multilayer induction furnaces, the temperature distribution, shown in figure (16), is applied as an input and the values of the number of layers, current density and time are obtained. To verify the accuracy of the results of inverse solver neural network, the results are compared with reference values used by ANSYS program to obtain the temperature distribution of figure (16). The results of neural network, the reference values and the percentage error are listed in table (11). More cases of multilayer induction furnaces are listed in table (12). Tables (9) to (12) show clearly the ability of the inverse solver neural networks to estimate the supply requirements for any desired temperature distribution with high accuracy and nil run time.

7 Conclusions

In this paper, attempts are made to support the numerical methods of analysis of induction furnaces by the advantages of neural networks. The neural networks were used to perform: a) Analysis of the induction heating furnaces. b) Specifying the power supply requirements of the induction furnace depending on the desired temperature distribution inside the workpiece.

Processing a large-scale data set requires a very large neural network. High accuracy results need a long training time. Many training algorithms fail to train the large-scale neural networks, where “out of memory “ error occurs with Levenberg-Marquardt training algorithm, and the required training goal is never reached with Resilient

Backpropagation training algorithm. The Scaled conjugate gradient training algorithm is used and it offered a good performance by reaching the training goal.

References

1- Rudnev, R., Loveless, D., Cook, R. and Black, M., "*Handbook of Induction Heating* ", MARCEL DEKKER Inc., New York, 2003.

2- Martin, D.H., "*High Efficiency Induction Billet Heater*", 10th Congress of Electroheat, Stockholm, 1984.

3- A.K.M. AL-Shaikhli, Hanan A.R. Akkar and Mohammed Moanes Ezzaldean Ali, " FEM-Simulation of Single and Multi Layered Induction Heating Systems", Engineering and Technology Journal, Vol.27, No.13, 2009.

4- Mohammed Moanes E. Ali, "*Analysis of Induction Heating Systems Based on Artificial Neural Networks*", Ph.D. Thesis, University of Technology-Baghdad, 2008.

5- Sivanandam, S.N. and Paulraj, M., "*Introduction to Artificial Neural Networks*", Vikas Publishing House, 2004.

6- Alexander, I.G., "*Neural Networks Theory*", Springer-Verlag, 2007.

7- Moller, M., "*A Scaled Conjugate Gradient Algorithm for Fast supervised Learning*", Neural Networks, Vol.-6, pp. 525-533, 1993.

8- Maizza, G. and Cali, M., "*Modeling of Single Shot Induction Heating by Inverse Finite Element Method* ", Wiley Interscience website, 1994.

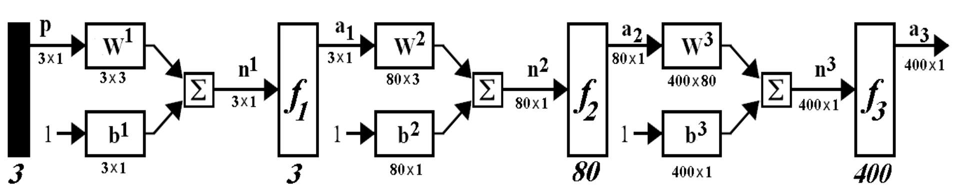


Fig. (1) Schematic diagram of the forward solver neural network

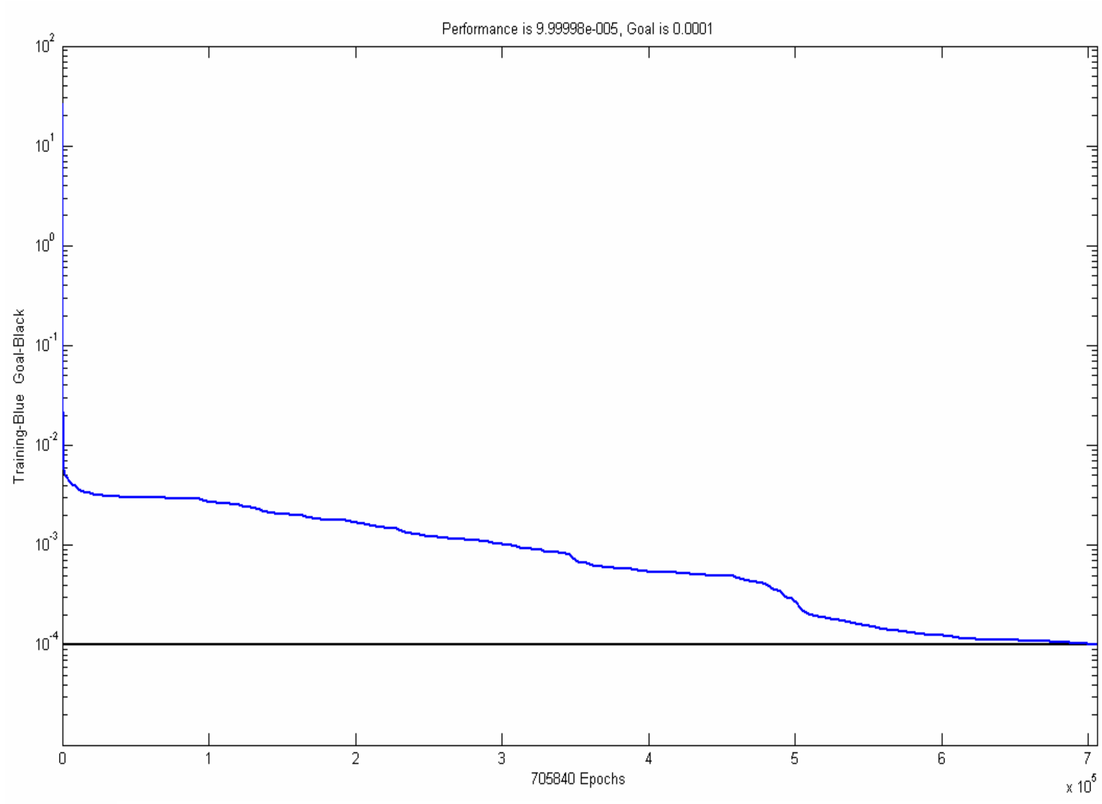


Fig. (2) Training plot for forward solver (single layer furnaces)

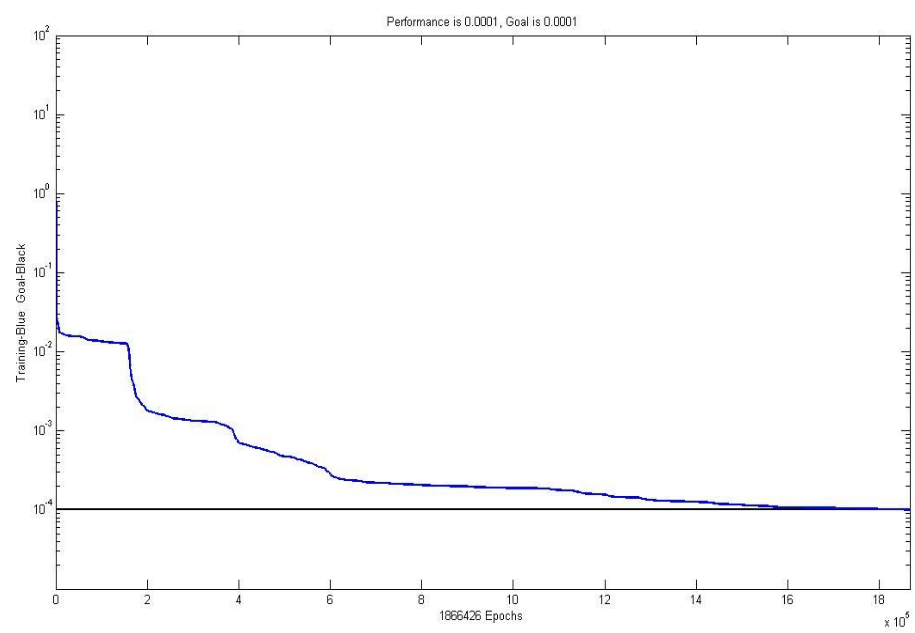


Fig. (3) Training plot for forward solver (multilayer furnaces)

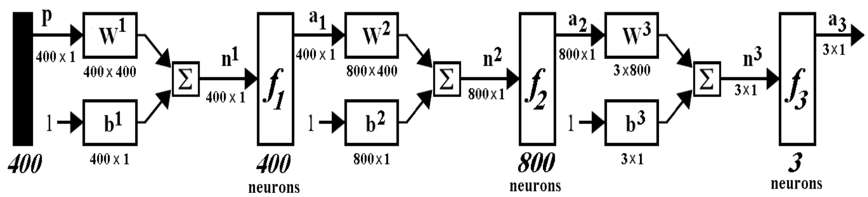


Fig. (4) Schematic diagram of the inverse solver neural network

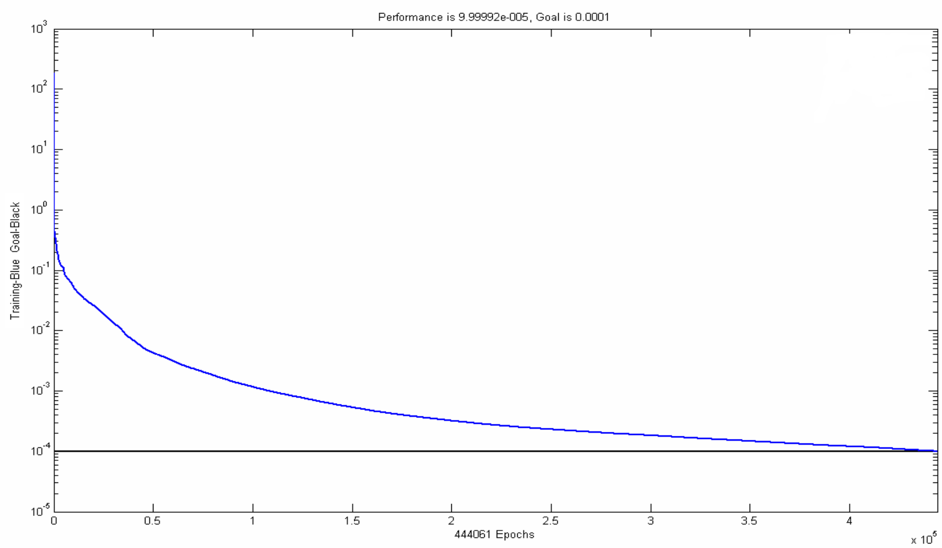


Fig. (5) Training plot for inverse solver (single layer furnaces)

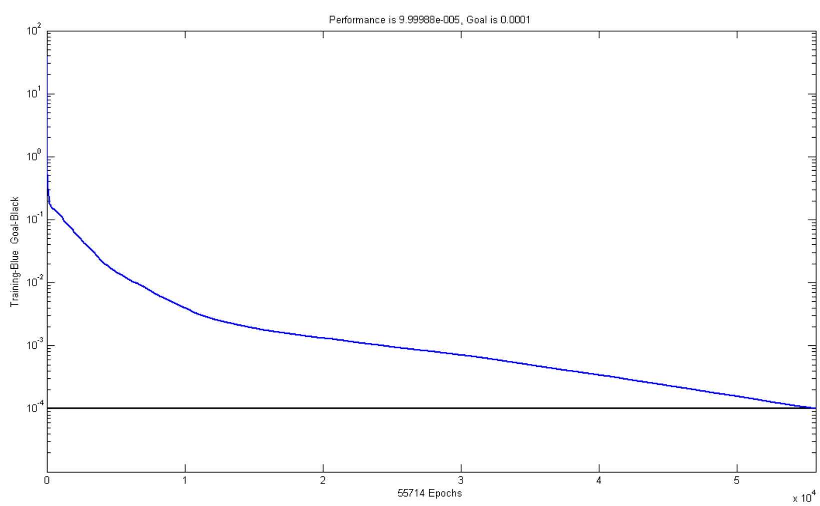


Fig. (6) Training plot for inverse solver (multilayer furnaces)

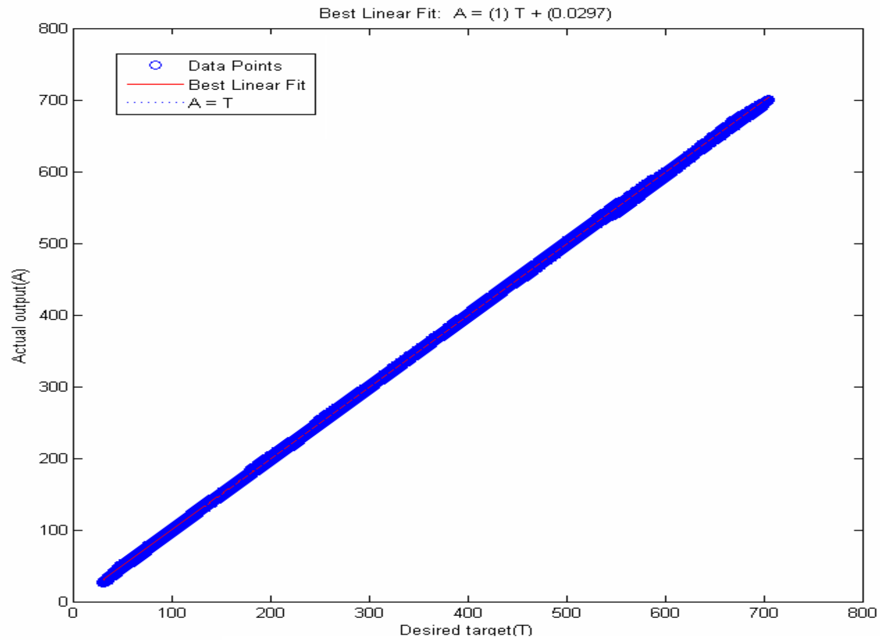


Fig. (7) Regression analysis of forward solver for single layer furnaces

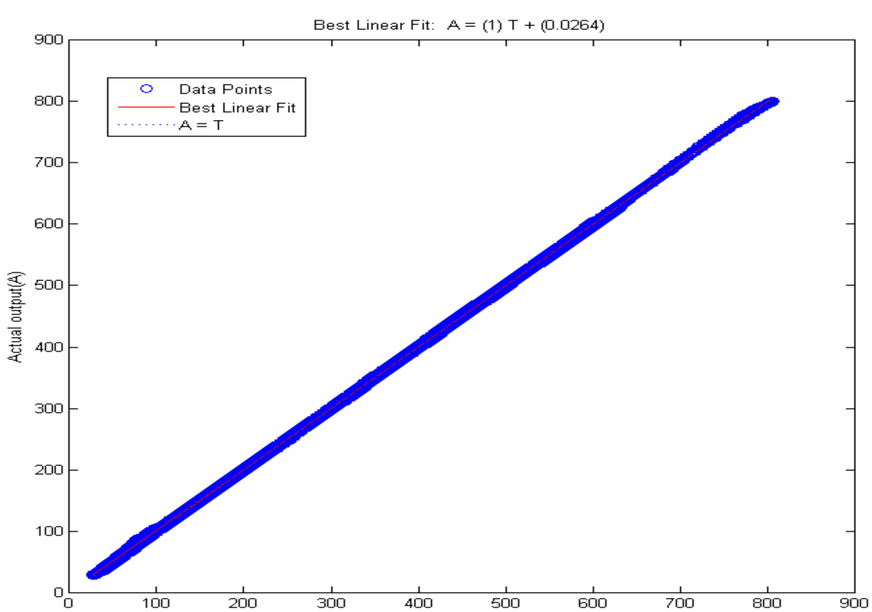


Fig. (8) Regression analysis of forward solver for multilayer furnaces

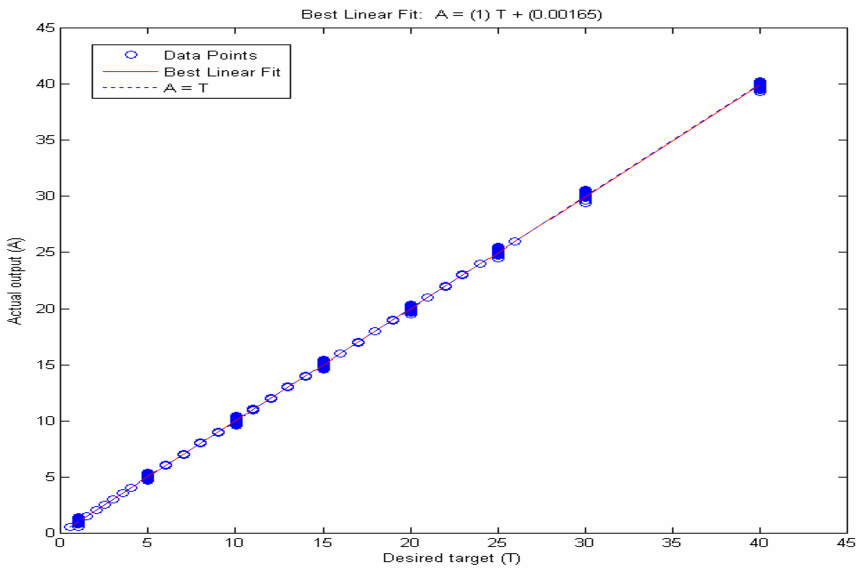


Fig. (9) Regression analysis of inverse solver for single layer furnaces

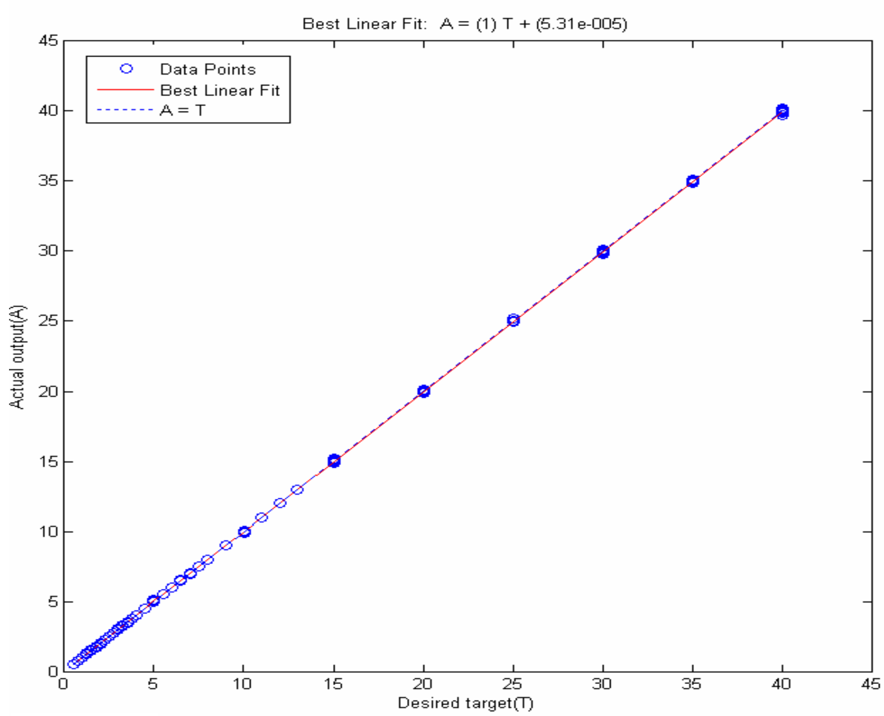
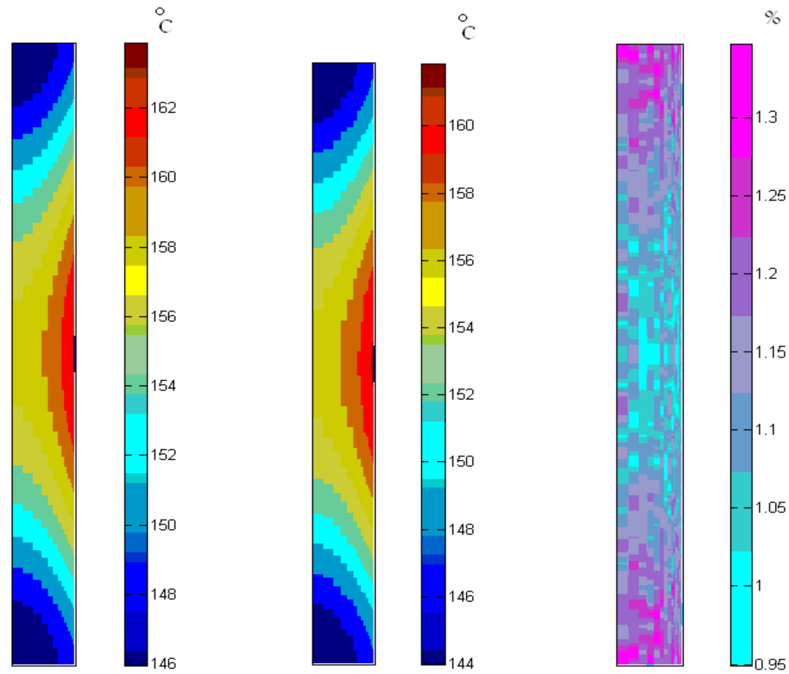
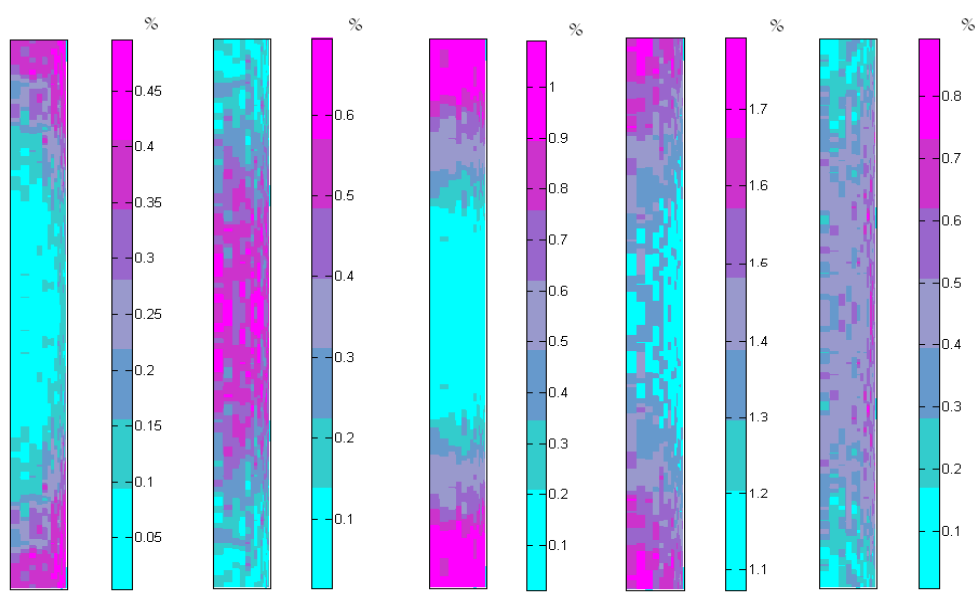


Fig. (10) Regression analysis of inverse solver for multilayer furnaces



-a- -b- -c-

Fig. (11) a-Temperature distribution (^{o}C) (by ANSYS), b- Temperature distribution (^{o}C) (by neural network), c- The percentage error for forward solver neural network (single layer furnaces)



(10kHz,13A/mm^2,85sec) (16.25kHz,9A/mm^2,120sec) (30kHz,5A/mm^2,190sec) (21.5kHz,7A/mm^2,110sec) (27.5kHz,8A/mm^2,60sec)

Fig. (12) Percentage errors for forward solver neural network (single layer furnaces)

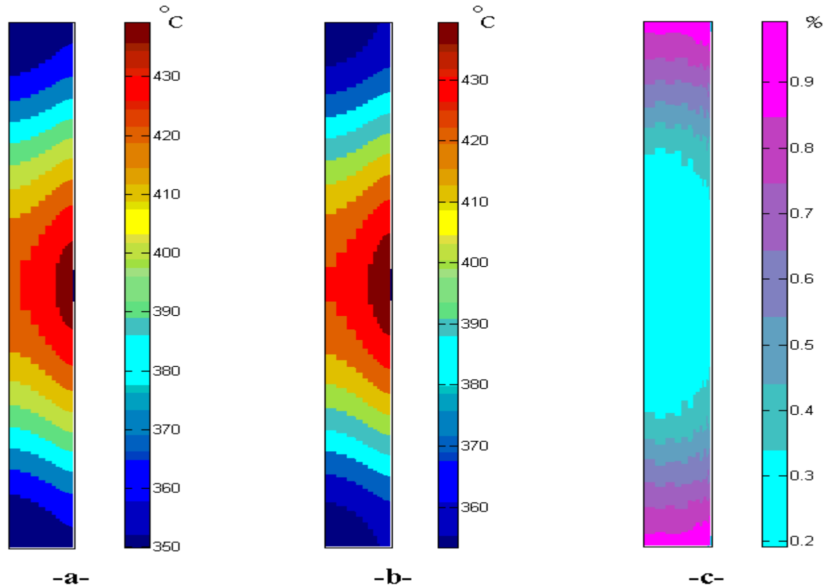


Fig. (13) a-Temperature distribution (°C) (by ANSYS), b- Temperature distribution (°C) (by neural network), c-The percentage error for forward solver neural network (multilayer furnaces)

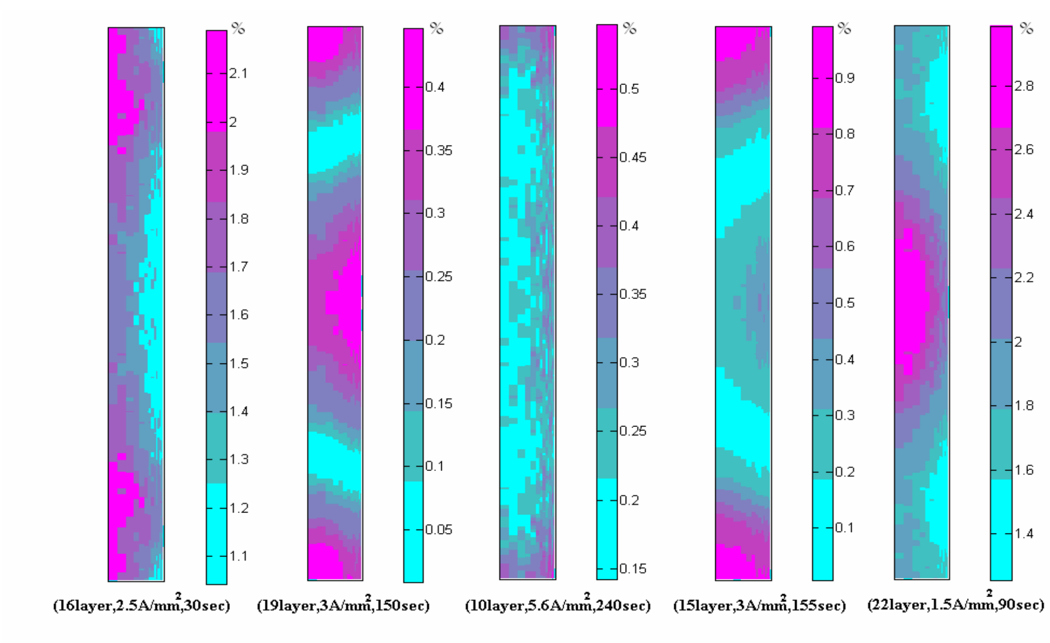


Fig. (14) Percentage errors for forward solver neural network (multilayer furnaces)

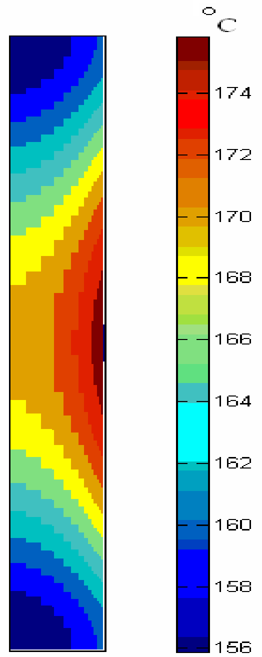


Fig. (15) Sample of temperature distribution used as an input to inverse solver neural network (single layer induction furnaces)

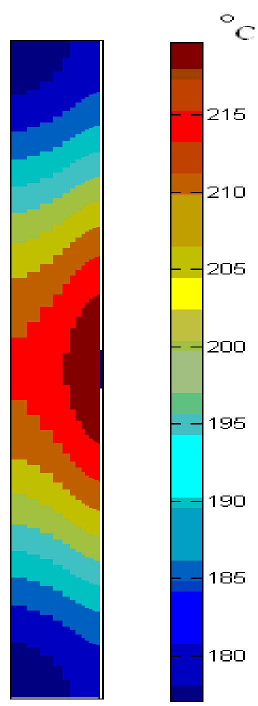


Fig (16) Sample of temperature distribution used as an input to inverse solver neural network (multilayer induction furnaces)

Table (1) Frequency and current density of simulated single layer furnaces

Frequency(kHz)	Current density (A/mm^2)							
1	19	20	21	22	23	24	25	26
5	12	13	14	15	16	17	18	19
10	9	10	11	12	13	14	15	16
15	8	9	10	11	12	13	14	15
20	7	8	9	10	11	12	13	14
25	5	6	7	8	9	10	11	12
30	4	5	6	7	8	9	10	12
40	4	5	6	7	8	9	10	11

Table (2) Number of layer and current density of simulated multilayer furnaces

Number of layers	Current density (A/mm^2)							
5	6	7	8	9	10	11	12	13
10	3	4	5	5.5	6	6.5	7	7.5
15	1	2	2.5	3	3.5	4	4.5	5
20	1	1.5	2	2.5	3	3.25	3.5	3.75
25	1	1.5	2	2.25	2.5	2.75	3	3.25
30	1	1.25	1.5	1.75	2	2.25	2.5	2.75
35	1	1.25	1.5	1.75	2	2.25	2.5	2.75
40	0.5	0.75	1	1.25	1.5	1.75	2	2.25

Table (3) Parameters of the forward solver neural network

Number of layers	3
Number of neurons in input layer	3
Number of neurons in 1st hidden layer	3
Number of neurons in 2nd hidden layer	80
Number of neurons in output layer	400
Minimum values for input elements	0
Maximum values for input elements	50
Transfer function of the hidden layers	tan-sigmoid
Transfer function of the output layer	Linear
Network training algorithm	SCG
Performance function	MSE

Table (4) Parameter of training forward solver on single layer furnaces data

Initial weights and biases	Random
Goal	10^{-4}
Number of epoch	705840
Time for one epoch	0.53 second
Number of training examples	512

Table (5) Parameter of training forward solver on multilayer furnaces data

Initial weights and biases	random
Goal	10^{-4}
Number of epoch	1866426
Time for one epoch	0.74 second
Number of training examples	512

Table (6) Parameters of the inverse solver neural network

Number of layers	3
Number of neurons in input layer	400
Number of neurons in 1st hidden layer	400
Number of neurons in 2nd hidden layer	800
Number of neurons in output layer	3
Minimum values for input elements	20
Maximum values for input elements	1000
Transfer function of the hidden layers	Tan-sigmoid
Transfer function of the output layer	Linear
Network training algorithm	SCG
Performance function	MSE
Total number of weights and biases	483603

Table (7) Parameter of training inverse solver on single layer furnaces data

Initial weights and biases	Random
Goal	10^{-4}
Number of epoch	444061
Time for one epoch	3.64 second
Number of training examples	512

Table (8) Parameter of training inverse solver for multilayer furnaces data

Initial weights and biases	random
Goal	10^{-4}
Number of epoch	55714
Time for one epoch	3.84 second
Number of training examples	512

Table (9) Verifying the accuracy of inverse solver neural network (single layer furnaces)

	Reference values (ANSYS)	Estimated values (Neural network)	Percentage error (%)
Frequency(kHz)	11	10.67	3
Current density(A/mm^2)	10	9.96	0.4
Time (second)	175	175.2	0.114

Table (10) Result of inverse solver neural network (single layer furnaces)

Reference values (ANSYS)			Estimated values(neural network)			Percentage error (%)		
Frequency (kHz)	Current density (A/mm^2)	Time (second)	Frequency (kHz)	Current density (A/mm^2)	Time (second)	Frequency	Current density	Time
40	4	125	39.93	3.97	124.8	0.17	0.755	0.160
25	6.5	180	24.59	6.5	180.6	1.66	0	0.332
25	6.5	185	24.56	6.45	185.4	1.791	0.775	0.215
23	7	120	22.67	7.05	121.2	1.455	0.709	0.99
30	7.1	115	30.4	7.14	115.2	1.315	0.560	0.173

Table (11) Verifying the accuracy of inverse solver neural network
(multilayer furnaces)

	Reference values (ANSYS)	Estimated values (Neural network)	Percentage error (%)
Number of layers	20	20.01	0.05
Current density(A/mm^2)	3.29	3.31	0.607
Time (second)	120	120	0

Table (12) result of inverse solver neural network
(multilayer furnaces)

Reference values (ANSYS)			Estimated values (neural network)			Percentage error (%)		
No. of layers	Current density (A/mm^2)	Time (sec.)	No. of layers	Current density (A/mm^2)	Time (sec.)	No. of layers	Current density (A/mm^2)	Time (sec.)
25	2.71	60	24.99	2.72	60	0.040	0.367	0
5	10.93	240	5.02	10.95	240	0.398	0.182	0
22	3.25	210	22.07	3.26	210	0.317	0.306	0
40	1	95	40.04	0.97	95.4	0.099	3.092	0.419
35	1.75	155	35	1.71	154.8	0	2.339	0.129