IJST, Transactions of Mechanical Engineering, Vol. 38, No. M1+, pp 269-274
Printed in The Islamic Republic of Iran, 2014

"Research Note"

PREDICTION OF THE TEMPERATURE OF THE HOLE DURING THE DRILLING PROCESS USING ARTIFICIAL NEURAL NETWORKS*

A. R. TAHAVVOR** AND S. SEPEHRINIA

Dept. of Mechanical Eng., Shiraz Branch, Islamic Azad University, Shiraz, I. R. of Iran
Email: tahavvor@iaushiraz.ac.ir

Abstract– Information about the temperature of drilling hole during the drilling process is important in work-piece quality and tools life aspects. In this study temperature of the drilling hole is determined using Artificial Neural Networks according to certain points' temperature of the work piece and two parameters, drill diameter and ambient temperature. To achieve this aim, two-dimensional model of work piece is provided; then by Computational Heat Transfer simulations based on Finite Volume Method, temperature in different nodes of the work piece is specified. Obtained results are used for training and testing the neural network. Temperature of specified points, drill diameter and ambient temperature are selected as inputs of the network and temperature of drilling hole is considered as an output data. Also, for comparison, temperature is obtained experimentally. Comparison between numerical results and experimental data shows that neural network can be used more efficiently to determine temperature of hole in a drilling process.

Keywords– Temperature of drilling hole, artificial neural network, Levenberg-Marquardt

1. INTRODUCTION

Drilling is a complex machining process used in almost 50 percent of industrial machining processes [1]. The temperature of the drilling hole has a great influence on the tool lifetime and the surface quality. Therefore, determination of drilling temperature section is important. To achieve this aim a new approach such as Artificial Neural Networks (ANN) can be an alternative to reduce cost of studies and computational time. ANN and other soft computing methods have been used by various researchers in the field of engineering systems [2-4] and simulation of experimental processes [5, 6]. Also, various numerical, analytical and experimental methods are used to estimate the temperature of drilling hole [7-13]. The purpose of this study is to understand whether an ANN approach is an appropriate method for determination of temperature of the hole or not. Some works are available in this field [14-16], but simulation and prediction of temperature with ANN is a novel study.

a) Problem definition and computational procedure

Work-piece and the related grid arrangement for CHT solution are shown in Fig. 1. For simulation, two-dimensional with quasi steady-state conditions with constant thermo-physical properties are assumed. The temperatures of the hole and ambient are varied from 313 K to 423 K and 24°C to 28°C respectively. Simulations are provided with a 10×10 cm^2 quadrangular model and 4mm to 10 mm in diameter drill bit. For numerical model, a convergence criterion is taken as 10^{-6}. Steel is used as a work-piece material. Some structured grid dimensions are tested to obtain optimum grid dimension. After several tests, it was decided that 2304 grid points are sufficiently fine to ensure a grid independent solution. Temperature of

*Received by the editors January 4, 2013; Accepted January 20, 2014.
**Corresponding author

four desired points, as shown in Fig. 2, are measured by using four channels thermometer with thermocouples type K. Points 1,2,3 and 4 have 1,2, √5, and √8 cm distance from hole axis respectively. Therefore, according to Richardson extrapolation method [17] temperature of the hole is determined.

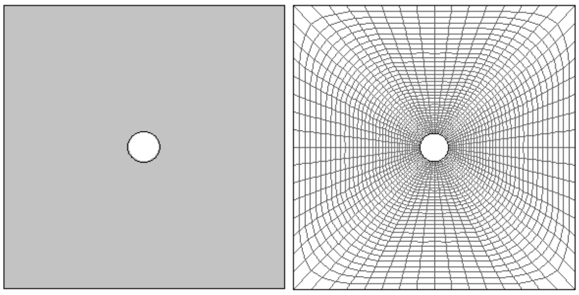

Fig. 1. Physical geometry and grid distribution

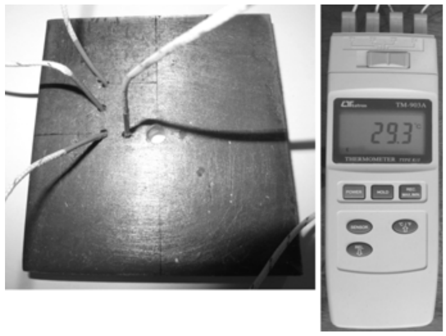

Fig. 2. Thermometer and work piece

2. ARTIFICIAL NEURAL NETWORKS AND ITS ARCHITECTURE

ANNs are composed of simple elements which operate in parallel. These elements are inspired by biological nervous systems (Fig. 3). We can train a network to perform a particular function by adjusting the values of the weights between elements according to the suitable learning rule. In this work a Multi-Layer Perceptron network is used and it is found that the back-propagation algorithm with modified Levenberg-Marquardt learning rule is the best choice because of its accuracy and speed. Input vector consists of six elements: temperatures of four nodes, drill bit diameter and ambient temperature. Target vector consists of one element: temperature of the drilling hole. Input-target pairs are applied to the network, and weights are adjusted to minimize the error between the network output and the target. Figure 4 and Table 1 show the network architecture and its parameter respectively.

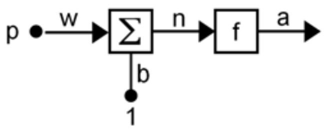

Fig. 3. An artificial neuron

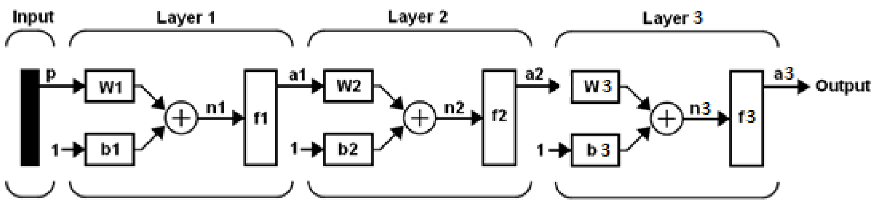

Fig. 4. Three-layer feed-forward neural network

Table 1. ANN architecture and training parameters

Architecture parameters			
The number of neurons	in each layer: Layer 1: 20, Layer 2: 20 and output Layer: 1		
Transfer functions	Layer 1 and 2: logistic sigmoid, Layer 3: pure linear		
Training parameters			
Learning parameter μ	0.001	Mean squared error goal	10^{-12}
Increasing factor of μ	10	Decreasing factor of μ	0.1

a) Levenberg-marquardt algorithm

The LM algorithm [18] is a technique that locates the minimum of a multivariate function. The performance function of the LM algorithm is defined as [19]:

$$F(w)=\sum_{p=1}^{P}\left[\sum_{k=1}^{K}\left(d_{kp}-o_{kp}\right)^2\right] \tag{1}$$

where w consists of all weights of the network, d_{kp} and o_{kp} are the desired and actual value of the k_{th} output and the p_{th} pattern respectively, P is the number of patterns, and K is the number of the network outputs. Weights are calculated using the following equation:

$$w_{i+1} = w_i - (H + \mu_i I)^{-1} \nabla F(w_i) \tag{2}$$

Where I is identity matrix, μ is a learning parameter and *H* is Hessian matrix. For μ = 0 and very large μ it becomes the Gauss-Newton and steepest-descent method respectively. Update rule is used as follows. If the error goes down we decrease *μ* to reduce the influence of gradient descent and if the error goes up, we follow the gradient more and so μ is increased by the same factor.

3. RESULTS

ANN is used to predict new results from the generated data for various drill bit diameters. Results are compared with CHT solutions. Data of d = 6, 7, and 10 mm are used as training data and temperature values are used as target. Some results of training procedure are illustrated in Figs. 5 and 6. Comparison between experimental data, ANN and CHT are presented in Table 2. These results show that ANN and CHT differences are negligible. The new results are used as testing data for ANN. In testing procedure very good agreement is observed between results of ANN and CHT. Also, values of weights/biases of the trained network are presented in Tables 3. It is shown that ANN method is capable of accurately determining temperature from the generated data. For closeness, R-square and Sum-Square Error are used to evaluate fitting of network as follows:

$$R^2 = 1 - \frac{\sum(\Omega_{EXP} - \Omega_{ANN})^2}{\sum(\Omega_{EXP} - \Omega_{m,EXP})^2} \ , \ SSE = \sum(\Omega_{EXP} - \Omega_{ANN})^2 \tag{3a, b}$$

where Ω_{EXP}, Ω_{ANN}, and $\Omega_{m,EXP}$ are the value obtained from the experiment, ANN, and mean value of experimental data respectively. Table 4 shows the R^2 and the SSE for testing cases. These tables show excellent correspondence between ANN and CHT.

Table 2. Comparison between numerical and experimental results

Temperature (K)							
diameter (mm)	Ambient	n1	n2	n3	n4	Hole(EXP)	Hole (ANN)
10	299	343.3	326.8	323.6	320.1	369.09	358.224
7	299	330.3	318.8	315.8	313	355.73	348.1771
6	299	328.4	317.6	315.2	312	353.09	347.6267
5	299	329.3	318.5	316	313	350.1	348.1998
4	299	328.3	314.1	312.1	310.4	341.55	347.8265

diameter (mm)	Ambient	n1	n2	n3	n4	Hole (CHT)	Hole (ANN)
6	297	308.24	303.57	303.11	301.55	317	316.9746
6	299	322.64	312.84	311.86	308.57	341	341.0004
6	301	316.75	310.21	309.56	307.36	329	329.0139
7	297	312.32	306.06	305.20	303.40	323	322.9820
7	299	317.86	310.16	309.09	306.88	331	330.9900
7	301	332.85	319.85	318.05	314.31	355	355.0096
10	297	332.88	317.84	316.17	311.46	349	349.0002
10	299	341.79	323.85	321.86	316.24	361	361.0022
10	301	346.55	327.46	325.34	319.36	367	366.9974

Table 3. Values of weights of each layer

Input Layer Weights Matrix, size: 20×6

-2.984	-1.453	2.839	0.177	0.395	0.741
0.122	2.238	0.476	2.624	-1.435	-2.673
-1.855	-0.558	-2.305	2.284	2.083	1.667
-3.248	1.405	-1.959	0.534	-2.515	-0.568
2.174	-0.295	-0.477	2.674	1.286	-2.755
-1.191	1.416	1.480	2.239	0.558	-0.449
-1.088	0.824	1.564	2.928	1.288	-2.062
1.630	-1.252	0.647	-2.702	2.129	-1.151
-1.431	-1.209	-0.623	2.031	2.993	-1.634
1.425	2.313	0.632	0.897	2.227	-2.816
-0.803	1.589	-3.026	2.519	0.194	-1.509
1.463	-0.207	-3.270	0.908	2.835	-0.818
0.816	1.082	-3.801	0.079	0.520	2.093
-0.755	-1.026	0.785	2.594	-3.420	-0.372
-2.143	1.351	-0.150	-1.105	-1.976	1.114
-1.386	2.411	-2.174	0.145	2.872	1.559
-0.124	-2.455	-0.396	3.378	-0.218	-0.357
-1.594	2.057	-2.421	0.744	2.691	0.536
-0.999	2.207	-2.352	1.722	-1.396	0.190
1.646	0.683	2.575	1.403	0.801	-2.802

2^{nd} Layer Weights Matrix, size: 20×20

-0.16	-2.32	1.90	-0.65	-2.21	-2.35	0.15	-1.80	-0.81	1.56	-0.30	-1.76	-1.55	-0.80	1.31	1.32	-1.55	-0.77	-0.67	-1.26
1.55	1.69	-0.31	-1.81	-1.39	-2.26	-0.63	1.56	-0.99	-1.47	-0.39	-1.15	-0.73	-0.01	1.29	-1.88	-1.77	2.31	-1.73	-0.84
1.27	2.23	-0.60	-0.09	-2.34	0.97	-2.75	-2.09	-2.45	0.35	0.62	-1.64	-0.60	-0.17	-2.04	0.52	1.03	-1.17	1.17	-1.95
-1.67	2.22	-2.12	-0.74	-0.27	1.67	1.15	-1.53	1.42	1.87	-2.07	-0.67	-1.85	-0.13	0.92	-0.96	0.79	0.60	-1.77	1.53
-1.58	2.10	-2.04	-2.13	-2.34	0.83	-0.14	1.34	-1.94	0.12	-1.78	-1.78	0.63	-1.10	-0.37	0.37	1.10	0.92	-2.37	2.29
0.56	1.72	1.14	-1.56	2.43	-2.36	0.64	-1.05	-0.61	1.30	1.74	-1.68	-0.56	0.38	-1.59	-0.66	0.90	-0.74	1.33	-2.78
0.22	0.08	-1.39	2.21	-1.81	2.35	1.67	0.25	-0.36	1.55	1.13	-1.93	2.22	0.95	-2.24	-0.46	-0.49	-2.09	-0.79	0.42
-0.66	-1.59	1.72	0.91	-2.07	-0.86	-1.97	1.03	-1.37	-0.53	-1.59	1.54	-0.62	-0.38	1.81	-1.31	-0.71	-2.29	1.01	-2.18
-2.16	-0.63	-2.49	-0.33	-1.24	-1.34	-2.52	0.29	0.86	0.96	-2.37	1.16	-0.02	-0.94	-1.66	-1.84	2.36	-0.51	-0.72	2.76
1.34	-1.41	1.47	1.51	-0.15	1.05	0.42	0.46	-0.44	-1.11	-1.65	-0.40	-1.93	-2.05	-1.42	-2.05	-0.76	-1.24	0.87	2.47
-1.20	-1.81	-0.62	-1.37	-1.59	-1.86	-0.06	-1.10	-1.53	2.39	-1.98	-1.40	-1.32	-2.19	0.89	1.15	0.82	0.95	-2.03	-1.25
-1.79	2.00	0.69	0.95	2.16	-1.23	2.14	0.71	1.13	-0.85	-0.42	-1.01	-1.74	-1.59	0.86	0.59	1.95	-0.43	1.79	2.12
-1.72	-0.98	2.02	0.81	-0.80	0.79	1.48	1.89	-1.57	-1.31	0.46	-2.28	0.69	-1.10	-0.11	1.82	1.68	1.52	0.91	-2.24
-1.99	-1.32	-0.64	0.60	-1.28	-0.24	0.36	2.27	-0.20	1.02	1.52	3.06	2.29	1.52	0.88	-1.62	0.03	1.56	1.18	0.26
1.54	-0.84	-0.38	-1.90	-2.26	0.93	-1.50	-2.17	-1.06	0.65	0.27	0.60	-0.96	2.11	-1.56	2.42	0.86	0.50	1.80	1.66
-0.45	-0.25	-0.38	0.34	-1.31	0.87	-2.82	-1.06	-0.30	1.18	-1.96	0.19	1.39	2.26	2.12	1.80	2.42	-1.92	-0.75	-0.26
-2.39	0.69	1.78	-1.06	-2.37	1.38	-1.94	-0.72	-2.01	-0.75	1.13	-0.53	-0.55	-0.17	-0.22	0.57	-0.26	2.42	0.61	2.48
2.03	-2.29	1.45	-1.94	0.01	1.41	1.64	0.68	-0.63	-0.93	-1.75	-1.81	0.06	-1.50	0.90	-0.27	-2.21	-1.18	0.46	2.14
0.55	1.57	-1.81	-1.49	0.98	-1.20	1.56	0.04	-1.38	-2.32	-1.25	1.00	2.41	1.34	-2.14	-1.09	1.28	1.92	0.08	-0.24
0.89	0.35	-1.62	1.71	0.79	0.48	0.96	1.58	-2.60	0.68	-1.81	2.71	1.45	1.37	1.45	1.28	0.67	-1.57	-1.11	-1.72

3^{rd} Layer Weights Matrix, size: 1×20

-0.5	-0.31	-0.41	0.052	1.541	1.045	-0.2	-0.48	0.227	0.308	0.825	0.912	0.571	-0.77	0.305	-1.01	-0.93	0.586	-1.3	-0.75

Transpose of Input Layer biases Matrix, size: 1×20

5.65	-4.11	3.23	3.30	-2.67	1.90	2.46	-1.98	0.10	-0.37	-0.38	-0.08	0.69	-2.26	-1.50	-2.19	-3.59	-4.19	-4.16	3.78

Transpose of 2^{nd} Layer biases Matrix, size: 1×20

9.903	2.758	0.565	2.79	5.616	-0.17	-3.79	5.376	5.005	0.993	6.385	-4.64	-3.56	-5.34	3.274	-1.13	0.16	4.131	2.088	0.141

3^{rd} Layer bias, size: 1×1

0.0757

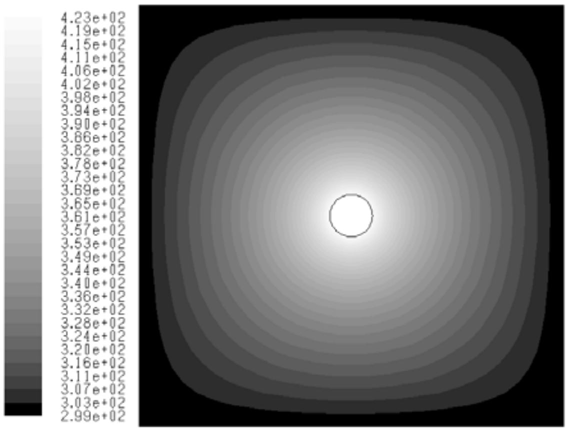

Fig. 5. Contours of static temperature from CHT for T_h=423 K , T_{amb}=299 K and D=10 mm

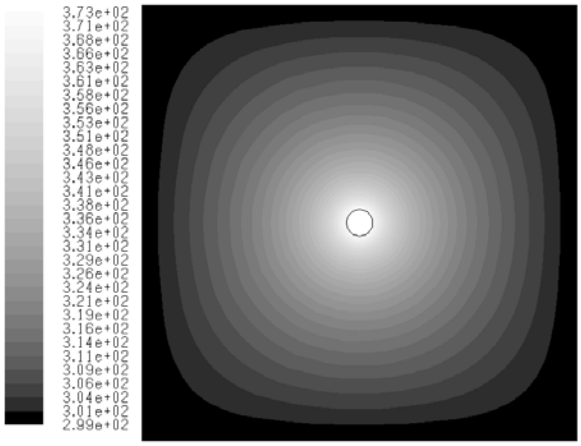

Fig. 6. Contours of static temperature from CHT for T_h=373 K , T_{amb}=299 K and D=6 mm

Table 4. R-square and Sum-Square-Error values of temperature for various tests

	d = 6 mm	d = 7 mm	d = 10 mm
R^2	0.999996	0.999999	0.999998
SSE	7.503×10^{-4}	5.162×10^{-4}	1.196×10^{-5}

4. CONCLUSION

The following conclusions are highlighted from the results:

- Very good agreement is found between ANN scheme and CHT simulation
- ANN method can easily be used to determine new results for temperature prediction in drilling process with considerably less computational cost and time.
- Back-propagation algorithm with Levenberg-Marquardt learning rule is the best choice for training this type of ANNs because of the accurate and faster training procedure.
- Replacement of the identity matrix with the diagonal of the Hessian in Levenberg-Marquardt update equation has great advantages in convergency and computation time.

NOMENCLATURE

d	target (desired value) of network
e	Error
o	output of network
I	Identity Matrix
J	Jacobian Matrix
H	Hessian Matrix
w	weights and biases

Greek letters

μ	Learning Parameter

Superscript

T	Transpose

REFERENCES

1. Rolt, L. T. (1965). *A short history of machine tools*. Cambridge, USA: MIT Press.
2. Kalogirou, S. (2000). Applications of ANN for energy systems, Appl. *Energy*, Vol. 67, pp. 17-35.
3. Salehi, J., Zadeh, P. M. & Mirshams, M. (2010). Collaborative optimization of remote sensing small satellite mission using genetic algorithms. *Iranian Journal of Science and Technology. Transactions of Mechanical Engineering, Vol. 36, pp. 117-128.*
4. Mousavi, S. A., Hashempour, M., Sadeghi, M., Petrofsky, J. S. & Prowse, M. A. (2010). A fuzzy logic control system for the Rotary dental instruments. *Iranian Journal of Science & Technology, Transaction B: Engineering*, Vol. 34, pp. 539-551.

5. Hasiloglu, M., Yilmaz, O. & Ekmekci, C. I. (2004). Adaptive neuro-fuzzy modeling of transient heat transfer in circular duct air flow. *Int. J. Therm. Sci*., Vol. 43, pp. 1075-1090.
6. Firat, M. & Goungor, M. (2007). River flow estimation using adaptive neuro-fuzzy inference system. *Math. Comput. Simul*., Vol. 75, pp. 87-96.
7. Ozcelik, B. & Bagci, E. (2006). Experimental and numerical studies on the determination of twist drill temperature in dry drilling: A new approach. *Mater. Des*., Vol. 27, pp. 920-927.
8. Bono, M. & Ni, V. (2001). The effects of thermal distortions on the diameter and cylindricity of dry drilled holes. *Int. J. Mach. Tools Manuf*., Vol. 41, pp. 2261-2270.
9. Wu, J. & Han, R. D. (2009). A new approach to predicting the maximum temperature in dry drilling based on a finite element model. *J. Manuf. Processes*, Vol. 11, pp. 19-30.
10. Apak, E. & Ozbayoglu, E. (2009). Heat Distribution within the Wellbore While Drilling. *Pet. Sci. Tech*., Vol. 27, pp. 678-686.
11. Chung, S. C. (2005). Temperature estimation in drilling processes by using an observer. *Int. J. Mach. Tools Manuf*., Vol. 45, pp. 1641-1651.
12. Palamà, B. A. (2003). A new method for evaluating formation equilibrium temperature in holes during drilling. *Geothermics*, Vol. 10, pp. 95-102.
13. Fuh, K. H., Chen, W. C. & Liang, P. W. (2003). Temperature rise in twist drills with a finite element approach. *Int. Commun. Heat Mass Transf*., Vol., pp. 345-358.
14. Biermann, D. & Iovkov, I. (2013). Modeling and simulation of heat input in deep-hole drilling with twist drills and MQL. *CIRP*, Vol. 8, pp. 87-92.
15. Sepúlveda, F., Rosenberg, M. D., Rowland, J. V. & Simmons, S. F. (2012). Kriging predictions of drill-hole stratigraphy and temperature data from the Wairakei geothermal field, New Zealand: Implications for conceptual modeling. *Geothermics*, Vol. 42, pp. 13-31.
16. Biermann, D., Iovkov, I., Blum, H., Rademacher, A., Taebi, K., Suttmeier, F. T. & Klein, N. (2012). Thermal Aspects in Deep Hole Drilling of Aluminium Cast Alloy Using Twist Drills and MQL. *CIRP*, Vol. 3, pp. 245-250.
17. Gerald, C. & Wheatley, P. (2002). *Applied numerical analysis*. Addison-Wesley, USA.
18. Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *SIAM, J. Appl. Math*., Vol. 11, pp. 431-441.
19. Wilamowski, B., Iplikci, S., Kaynak, O. & Efe, M. O. (2001). An algorithm for fast convergence in training neural networks. *proc. Int. Conf. Neural Netw*., Washington, DC, USA.