The current issue and full text archive of this journal is available at
www.emeraldinsight.com/0961-5539.htm

Received July 2005
Revised January 2006
Accepted February 2006

Neuro-genetic optimization of micro compact heat exchanger

H.W. Lee
School of Mechanical Engineering, University of Science Malaysia, Pulau Pinang, Malaysia
Y.J. Teng
SIRIM Berhad, Shah Alam, Malaysia
I.A. Azid
School of Mechanical Engineering, University of Science Malaysia, Pulau Pinang, Malaysia, and
K.N. Seetharamu
M S Ramaiah School of Advanced Studies, Bangalore, India

Abstract

Purpose – This paper seeks to introduce an optimization method for maximizing the effectiveness of the micro compact heat exchanger (MHE) under various geometrical parameters.

Design/methodology/approach – Optimization is realized using the neuro-genetic methodology which combines the application of artificial neural network (ANN) together with genetic algorithms (GA). The analyses are divided into two main sections; the first being the modeling and prediction using finite element method, the second being the neuro-genetic optimization. Initial results obtained from the finite element modeling are utilized for training in ANN. Subsequently, optimization is done using GA, once a well trained ANN is achieved.

Findings – ANN accurately predicts the effectiveness of the MHE and compares well with those obtained from the finite element simulation. Optimization shows a significant improvement in the maximum effectiveness of the MHE achievable for the given range of input parameters. Additionally, computational effort has been minimized and simulation time has been drastically reduced.

Research limitations/implications – This analysis is valid for constant fluid properties and for steady-state conditions. Additionally, optimization is limited to the range of the trained input parameters.

Practical implications – This paper is very useful for practical design of various types of heat exchangers.

Originality/value – This paper will be useful for the design of the MHE where its performance can be analyzed for a given range of geometries with minimal effort. This methodology will also be applicable for other types of heat exchangers.

Keywords Finite element analysis, Heat exchangers, Optimization techniques, Neural nets, Programming and algorithm theory

Paper type Research paper

International Journal of Numerical Methods for Heat & Fluid Flow
Vol. 17 No. 1, 2007
pp. 20-33

0961-5539
DOI 10.1108/09615530710716063

Nomenclature

a = thickness of wall that separates the water and air channels (μm)
b = width of water channel (μm)
c = height of water channel (μm)
C_p = specific heat (kJ/kgK)
Eff. = effectiveness

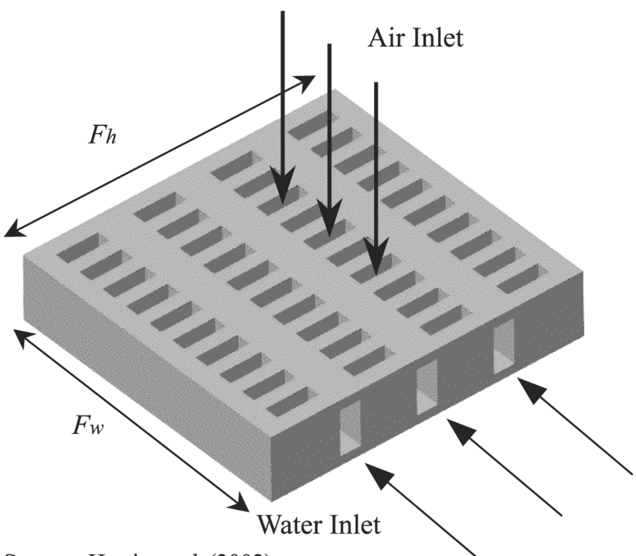
F_w	= overall width of heat exchanger (cm)	Q	= total heat transfer (W)
F_h	= overall length of heat exchanger (cm)	T	= temperature of air or water (°C)
		V	= velocity (m/s)
H	= height of air channel (μm)	w	= width of air channel (μm)
K	= thermal conductivity (W/m K)	y	= width between air channels (μm)
L	= length of air channel (μm)	w_c	= $w + y$ (μm)
$\dot{m}$	= mass flow rate (kg/s)	x, y, z	= Cartesian coordinates

Introduction

In recent years, with the rapid emerging of micro electromechanical systems (MEMS), many micro-fabrication technologies have been developed and being introduced in the field of heat transfer engineering to realize micro-channel devices. Compared with conventional heat exchangers, the main advantage of micro compact heat exchanger (MHE) is their extremely high heat transfer area per unit volume. However, the well-known benefits of small channels with respect to heat transfer must be weighed against the cost of the steep pressure gradient associated with flow through micro-channels. Thus, both the heat transfer and the flow resistance should be considered in the design of MHE.

A basic schematic of the cross flow MHE is shown in Figure 1. This MHE is designed based upon the performance criteria of a car radiator where its main function is for the dissipation of heat to air and prevent the engine from overheating. Figure 2 shows dimensions that specify the internal geometry of the heat exchanger while details for other parameters associated with the basic model of the MHE are given in Table I.

Not much analysis has been carried out for the MHE using two fluids to exchange heat. This is due to the massive computational fluid dynamics calculations involved.



Source: Harris *et al.* (2002)

Figure 1. Schematic of cross flow MHE

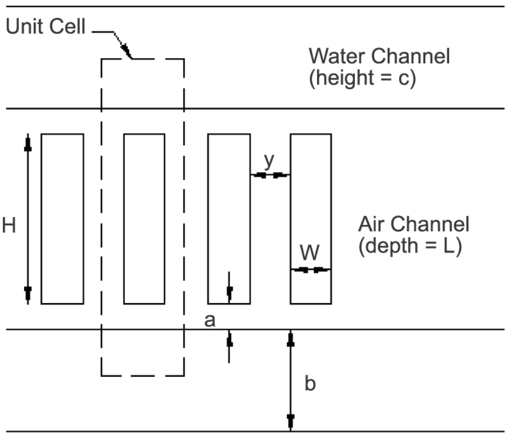


Figure 2.
Internal geometry for MHE

Table I.
Parameters associated with the basic model of the MHE

Parameters	Value
K (W/m K)	0.2
W (μm)	200
H (μm)	750
y (μm)	200
L (μm)	1,443
a (μm)	125
b (μm)	500
c (μm)	1,200
V_a (m/s)	4-12

An improved model using finite element method (FEM) has been earlier established and validated against the cross flow MHE fabricated by Harris *et al.* (2000, 2002). It has been observed that FEM takes considerable time and effort in modeling and prediction especially when the behavior is nonlinear and properties are temperature dependent. Therefore, there is a need to supplement the FEM so that we can do the parametric studies in a shorter time and with little effort. This paper serves the purpose where it will present a neuro-genetic methodology to simulate and optimize the performance of the MHE for different working parameters and geometries.

This methodology combines both the artificial neural network (ANN) and genetic algorithms (GA) together to achieve the optimization goal. Questions have been raised as to why ANN was the preferred choice for parametric studies. Leong *et al.* (2002) has presented the capabilities of ANN for thermal performance prediction of plastic ball grid array. In his paper, he has shown that result obtained from ANN at one instance is 69 times more accurate than multiple linear regressions for the predictions of thermal resistance. For this reason, ANN should be considered for parametric studies since it offer higher accuracy results and a platform for better integration with GA.

In general, ANN is capable of building up the quite complex and nonlinear model through training by making use of available data obtained through FEM simulation.

Once trained, the network can then be fed with any unknown input and are expected to predict the output with a high degree of accuracy. The trained ANN is then embedded as a fitness function into the GA analysis. The combination of GA and ANN forms the neuro-genetic tool for search and optimization purposes. The purpose of the optimization is to find the maximum effectiveness achievable using different range of parameters. GA can accurately predict the maximum effectiveness for a given range of input parameters such L, c, w, H, y, a, b and the capacity ratio. Commercially available software MATLAB version 7.0 is used to generate both the ANN and GA.

Artificial neural network

ANN is based on the working process of human brain in decision making. It is categorized under artificial intelligence method and has been applied in many different fields such as control (Baratti *et al.*, 2003), finance, aerospace, industrial and manufacturing (Ferreira *et al.*, 2001; Jain and Vemuri, 1999).

Figure 3 shows a typical neural network which consists of sets of input, sets of hidden layer, sets of output and weighting functions. The artificial neurons are organized in layers with one or more intermediate hidden layers placed between the input and output layers, and send their signals "forward." Each layer has a number of neurons connected with neurons in the adjacent layers through unidirectional connections. The information flow is only allowed in one direction during the training process; that is from the input layer to the output layer through the hidden layers. There can be any number of hidden layers in the architecture. The hidden layer has a synaptic weighting matrix, W_m associated with all the connections made from the input layer to the hidden layer.

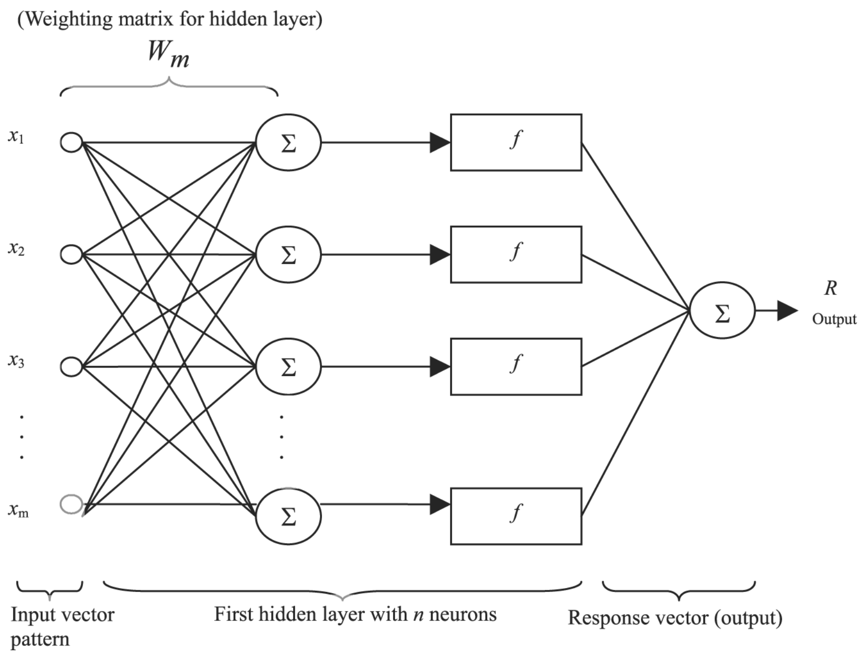


Figure 3.
Feed-forward multilayer perceptron ANN

The synaptic weights are assumed to be fixed and training process must be carried out to adjust the weights to perform a desired mapping. One of the training algorithms that can be used is the back-propagation method.

In this paper, the multi layer perceptron neural network has been trained using the back-propagation algorithm. The back-propagation algorithm uses supervised learning, which means that we provide the algorithm with examples of the inputs and outputs we want the network to compute, and then an error defined as the sum square error (SSE) is calculated. SSE is nothing but the difference between the actual and the predicted results. In other words, the network is defined to correlate between the inputs and the outputs by training the network with available data. Initially, the network obtains some information signals by neurons in the input layer and the output produced from the first layer is then fed subsequently into the second layer and so on. The errors are then propagated backwards. The idea of the back-propagation algorithm is to reduce this error until the ANN learns the training data. The training begins with random weights and the goal is to adjust them so that the error will be minimal. Training continues until the errors converged to a specified value or after reaching the maximum epoch. Once the network is trained, it can then be fed with any unknown input and is expected to predict the output with a high degree of accuracy.

Genetic algorithms

GA are adaptive search methods based on Darwinian's principles of natural selection, survival of the fittest and natural genetics. They combine survival of the fittest among string structures with a structured yet randomized information exchange to form a search algorithm with some of the innovative flair of human search. As in human genetics, GA exploits the fittest traits of old individuals to create a new generation of artificial creatures (strings). With each generation, a better population of individuals is created to replace the old population. Based on these principles, genetic algorithm is developed as a search tool that efficiently exploits historical information to speculate on new search points with expected improved performance.

The genetic algorithm determines which individual should survive, which should reproduce and which should die. It also records statistics and decides how long the evolution should continue. In GA, a population of individuals (parents) within a machine represented by chromosomes is generated. Each individual in the population represents a possible solution to a given problem. Figure 4 shows a typical chromosome with two 4-bit genes. Each chromosome is made up of a sequence of genes that contain alleles. As shown in Figure 4, alleles may not only consist of binary digits (0 and 1) but can also consist of floating point numbers, integers, symbols or matrices. The representation scheme is determined depending on the nature of the problem structure.

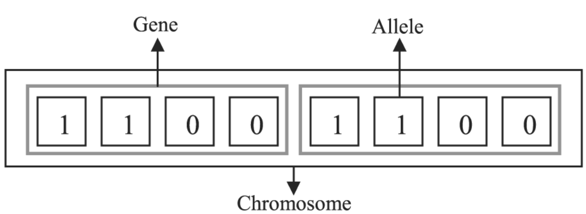


Figure 4. Chromosome with two 4-bit genes

Individuals in the population then go through a process of evolution. This process involves each individual being evaluated and are given a fitness score according to how well it is suited to be a solution to the problem. Individuals with a high fitness score will be selected and allowed to reproduce with other individuals in the population.

The properties of each individual are described by using a chromosome and reproduction within individuals occurs through the crossover and mutation process. Figure 5 shows the crossover and mutation processes of an 8-bit chromosomes. The crossover process is the exchange of alleles between two chromosomes whereas the mutation process involves changing allele values within a single chromosome. These two processes produce new individuals that will become the new population of solutions for the next generation. The new individual share some traits and features from the parent. Members of the population with a low fitness score will be discarded and are unlikely to be selected for the next evolution process.

The entire process of evaluation and reproduction then continues until either the population converges to an optimal solution for the problem or the genetic algorithm has run for a specified number of generations. A more detailed description of GA can be found in the text written by Goldberg (1989) and Mitchell (1996).

The Genetic Algorithm Toolbox (GAOT) used in the present study was written by Christopher Houck at North Carolina State University. The current version of GAOT files is available in the public domain (http://lancet.mit.edu/ga/dist/). The overview of GAOT including its implementation as well as some examples solved may be obtained from the same site.

In this paper, the data range for each parameter for optimization purpose has been reduced to avoid the size of the MHE to become of macro scale. Thus, the length of each parameter for the dimension of the MHE will be kept less than 1 mm except for the length of the MHE, L which will be kept below 2 mm.

Analysis

Modeling and prediction using finite element method

In order to train the ANN for extensive parametric studies, we require a preliminary database. This database is provided through simulations carried out using FEM.

The first step for numerical solution of any differential equation is to discretise the domain over which the solution is sought. The MHE is divided into a number of elements along the z-direction, i.e. length of air channel. The actual unit cell for analysis is very complex. In order to reduce modeling and computational effort, a simplified element has been made. Figure 6 shows the simplified model along with the nodes and the heat transfer mode. The walls are discretised by four noded bilinear elements. For each wall, heat transfer by conduction is only considered in two directions, i.e. x- and z-axis. However, heat transfer by conduction in the direction of y-axis will also be considered for the two vertical walls at both the left- and the right-hand sides. The walls made up of nodes 1, 2, 6, 5 and 4, 3, 7, 8 are treated as fins for the transfer of heat. Air and water streams are discretised by two noded linear elements.

Parent A: 1000 0000 } Crossover → Parent A’: 1000 <u>1111</u>
Parent B: 1001 <u>1111</u> Parent B’: 1001 0000

Parent C: 1111 10<u>10</u> → Mutation → Parent C’: 1111 101<u>1</u>

Figure 5.
The crossover and mutation processes

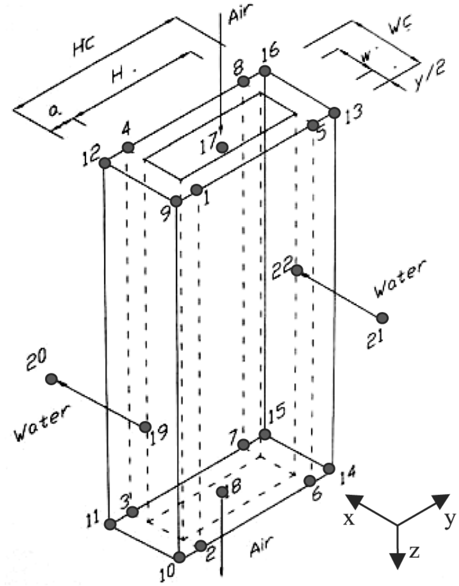


Figure 6.
Simplified unit cell with nodes for MHE

Both the walls made up of nodes 9-12 and 13-16 receive heat from the hot water flowing in the water channels by means of convection. The heat will then be transferred to both the walls made up of nodes 1-4 and 5-8 by conduction in the y-direction. These two new walls will dissipate heat to the air streams through convection and the fin effect.

For present study, the inlet temperature of air and water is set at 25 and 58°C, respectively. An assembly of elements in the z-direction will complete a full single air channel. The method of discretisation is very general and can be extended to either in the x- or y-direction, or in both directions.

The Galerkin finite element formulations have been developed by taking into consideration the required boundary conditions. The formulations are then solved using a self-developed code in MATLAB. For an element, it contains 22 nodes; therefore, it will produce a 22×22 matrix. The element matrices are stored in proper order to obtain the global stiffness matrix as explained in Lewis *et al.* (1996, 2004), Ranganayakulu *et al.* (1995) and Segerlind (1984). The matrix is then solved using Gauss elimination method to get the wall, air and also water temperatures.

Effectiveness of the MHE can also be calculated using equation (1). The ratio for different fluid's specific heat as given by equation (2) is simulated to study their effect on the effectiveness of the MHE. In order to include dimensionless study, the NTU is introduced and is given by the equation (3). The relationship between effectiveness and NTU for different value of capacity ratio can be expressed by equation (4):

$$\begin{aligned}\text{Eff.} &= \frac{\text{Actual heat transfered}}{\text{Max. heat that could be transferred}} \times 100 \\ &= \frac{T_{\text{air,outlet}} - T_{\text{air,inlet}}}{T_{\text{water,inlet}} - T_{\text{air,inlet}}} \times 100\end{aligned} \tag{1}$$

$$\text{Capacity ratio} = \frac{C_{p(\min)}}{C_{p(\max)}} = \frac{\dot{m}_{\text{air}} \times C_{p,\text{air}}}{\dot{m}_{\text{water}} \times C_{p,\text{water}}} \tag{2}$$

$$\text{NTU} = \frac{UA}{C_{\min}} = \frac{Q}{\Delta T_{\max} \times C_{\min}} = \frac{Q}{(T_{\text{water,inlet}} - T_{\text{air,inlet}}) \times C_{\min}} \tag{3}$$

$$\text{Eff.} = (R_{\text{ratio}} \times \text{NTU}^2) + 100\text{NTU} \tag{4}$$

where: $R_{\text{ratio}} = 15.016(\text{capacity ratio})^2 - 63.772(\text{capacity ratio}) + 26.404$.

Neuro-genetic optimization methodology

Results obtained from the finite element simulation will subsequently be used to train the ANN algorithm.

The ANN used in this study is shown in Figure 7 where there are eight neurons in the input layer and one neuron in the output layer. For our analysis, the ANN has been created using the ANN toolbox available in MATLAB version 7.0. Analysis is done to predict the effectiveness of the MHE for different input parameters. The ANN architecture used in this work is the feed-forward multilayer perceptron neural network while the training method implemented is the back-propagation algorithm with two hidden layers. Table II shows the architecture of the ANN used for our analysis while Table III shows the range of input parameters investigated using the ANN.

There will be two sets of data from each category of parameters for the FEM simulation that will not be used as training inputs to the ANN. This is because simulation data that were not trained will be used to verify the correctness of the ANN results. This is achieved by comparing the predicted ANN results to the FEM simulation results.

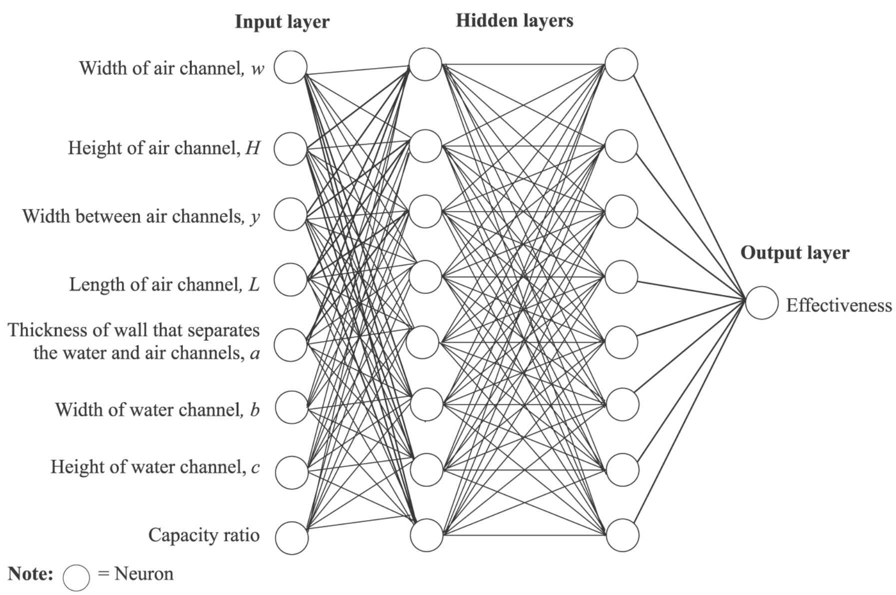


Figure 7.
Illustration of a multi layer perceptron neural network modeling

Once the ANN results are validated, a well trained ANN has been obtained. Thus, new values for parameters such as L, c, w, H, y, a, b and the capacity ratio will be fed into the ANN algorithm again. The ANN will be able to predict the effectiveness of the MHE for the desired input parameters with high level of accuracy.

After completing the training process for ANN, it is then combined with GA for optimization purpose. The objective of the optimization is to maximize the effectiveness of the MHE. GA will generate initial population for parameters L, c, w, H, y, a, b and capacity ratio and this data is then fed into the trained ANN algorithm as input variables to predict the performance of the MHE, namely the effectiveness. The predicted effectiveness of the MHE will be evaluated by GA and is given a fitness score which are then measured, recorded, ranked and compared with previous iteration result. The top grade chromosomes (highest value of fitness score) will be selected and allowed to reproduce with other individuals in the population where they will undergo the crossover and mutation processes. The population of the newly recombined and mutated chromosomes becomes the new input parameters for the trained ANN which subsequently predicts the effectiveness for new iteration in GA. GA continues to iterate until the values of the effectiveness converged. The converged result will yield the maximum effectiveness for the MHE. Figure 8 shows the neuro-genetic optimization process for the MHE.

Results and discussion

The trained ANN was able to predict accurately the effectiveness of the MHE. Figure 9 shows the comparison of the effectiveness obtained using ANN with the finite element simulation. The length of the MHE was varied from 1,500 to 4,000 μm while other parameters were fixed at values similar to the basic model (Table I).

Table II.
Architecture of the ANN used for the analysis

Parameters	Type/value
Architecture	Feed-forward
Training algorithm	Back-propagation
Transfer function	All logsig
Hidden layer and neurons	Two hidden layer with eight neurons each
Maximum epoch	1,000
Learning rate	0.000001
Sum square error	1×10^{-5}

Table III.
Range of parameters investigated in ANN

Parameters	Range
w (μm)	50-550
H (μm)	300-3,000
y (μm)	50-8,000
L (μm)	1,443-4,000
a (μm)	50-8,000
b (μm)	100-8,000
c (μm)	400-1,400
Capacity ratio	0-1

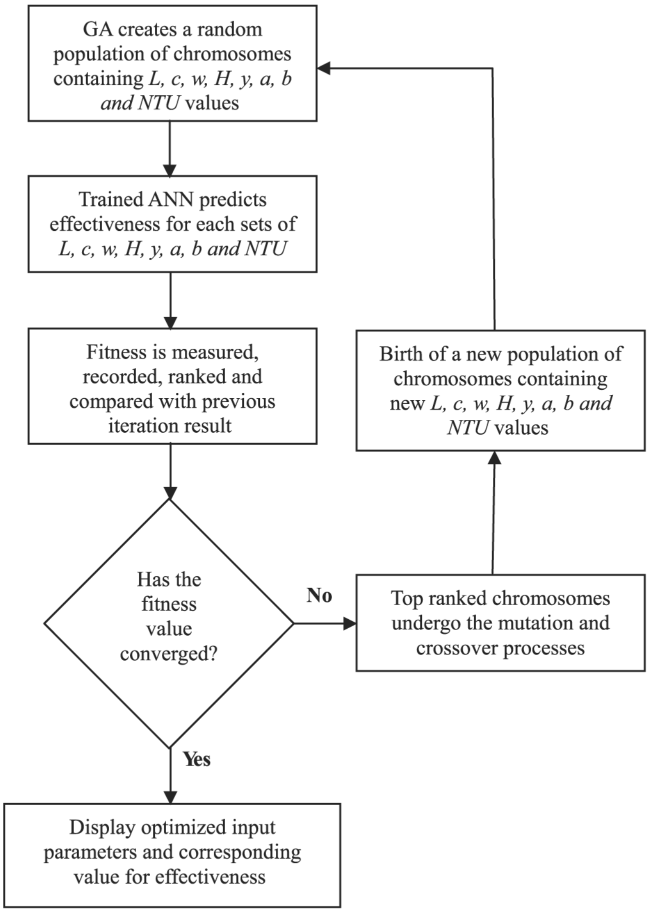


Figure 8.
Neuro-genetic optimization methodology

Additional comparisons of results obtained through the ANN are presented in Table IV. As mentioned earlier, two sets of data from each category of parameters from the FEM simulation results will be presented for the purpose of comparison with the ANN prediction. The results shown are for the value of the capacity ratio set at 0.50 while other parameters are fixed accordingly.

From Table IV, it can be seen that the percentage of errors are less than 2 percent while for some cases, it is negligible. ANN was able to provide accurate predictions because of the fact that there is no scattered data in the FEM simulation results.

Once a well-trained ANN is found, it is then embedded as a fitness function into GA for the purpose of optimization. The goal is to find the maximum effectiveness achievable using different range of parameters for the MHE. From our analysis, GA can accurately predict the maximum effectiveness for a given range of input parameters using the crossovers and mutations parameters of Tables V and VI. Figure 10 shows predictions using GA to find the maximum effectiveness for a variation in w. The maximum effectiveness obtained is 24.776 percent. Table VII shows the optimized results obtained

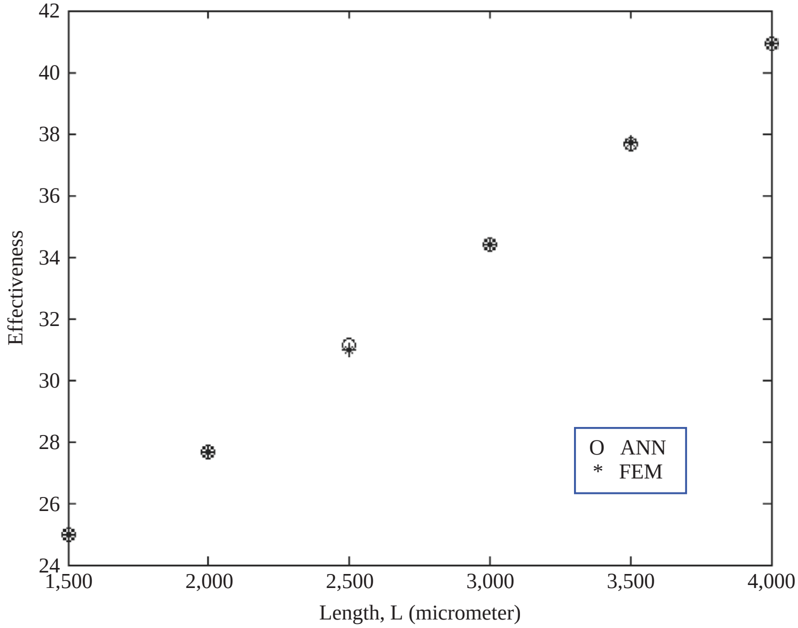


Figure 9.
Comparison of results between FEM and ANN simulation for variation in *L*

Table IV.
Comparison of FEM and ANN prediction in effectiveness of MHE for different values of parameters (capacity ratio at 0.5)

Parameters (μm)	FEM	ANN	Variation (percent)
$w = 250$	23.06	23.01	0.217
$w = 400$	20.64	20.63	0.050
$H = 700$	23.15	23.11	0.170
$H = 1{,}000$	23.73	23.73	0
$y = 400$	28.84	28.28	1.930
$y = 1{,}000$	36.82	36.46	0.990
$L = 1{,}443$	23.32	23.28	0.185
$L = 2{,}500$	28.77	28.85	0.270
$a = 500$	15.17	15.15	0.120
$a = 1{,}000$	10.34	10.54	1.930
$b = 250$	19.78	19.80	0.101
$b = 1{,}000$	20.05	20.04	0.040
$c = 700$	18.89	18.88	0.030
$c = 1{,}000$	21.30	21.27	0.140

using GA in bold for variation in w. One larger and one smaller value for w between the optimized w found using GA is used to verify that the optimized parameter, $w = 171.27\ \mu$m indeed produces the maximum effectiveness.

From this analysis, the optimized parameters that produce the maximum effectiveness for the MHE would be used as the proposed new design parameters for the MHE. Table VIII shows the GA optimized design parameters for the MHE where the maximum effectiveness obtained are 59.191 percent as opposed to the maximum effectiveness of 24.776 percent obtained from initial design. This gives a significant improvement of almost 35 percent.

The results clearly indicate that a well trained ANN combined with GA can be used to optimize the performance of the MHE with confidence. Furthermore, the computational time required for optimization using the neuro-genetic methodology was less than 5 min using an Intel Pentium IV 1.5 GHz processor. With the neuro-genetic optimization, the

Type	Parameters
Arithmetic	2
Heuristic	[2 0]
Simple	2

Table V. Crossovers parameters used for the GA analysis

Type	Parameters
Boundary	4
Multi non-uniform	[3 10 3]
Non-uniform	[3 10 1]
Uniform	4

Table VI. Mutations parameters used for the GA analysis

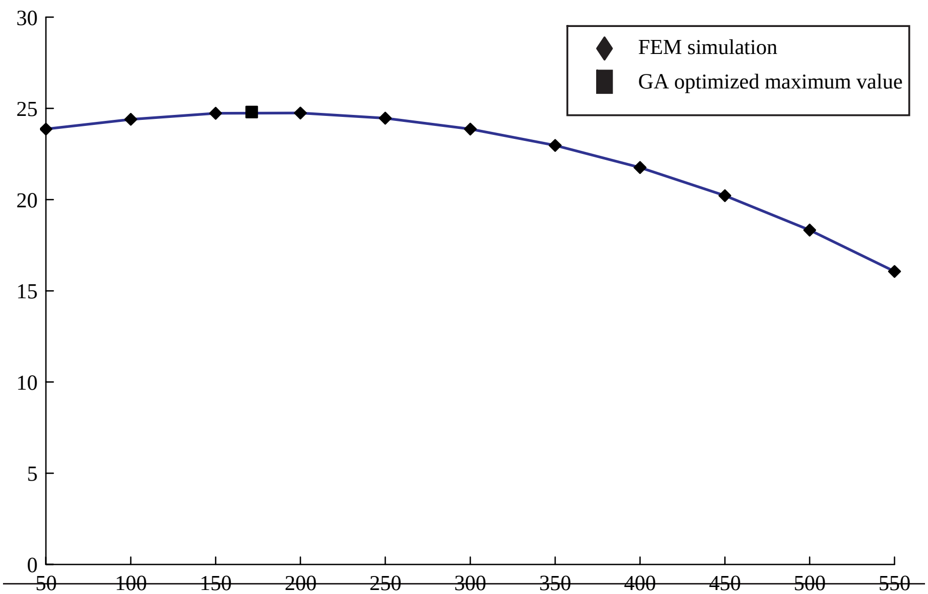


Figure 10. Maximum effectiveness obtained using GA for variation in w with capacity ratio fixed at 0.5

w	Effectiveness (percent)
171.00	24.770
171.27	*24.776*
172.00	24.771

Table VII. Optimized results using GA for variation in w with capacity ratio fixed at 0.5

Parameters	Value
W (μm)	300
H (μm)	997
y (μm)	410
L (μm)	1,992
a (μm)	73
b (μm)	139
c (μm)	1,185
Capacity ratio	0.22
Effectiveness (percent)	59.19

Table VIII. **Note:** Optimized design parameters of MHE obtained using GA

necessity of remodeling the time consuming finite element models can be eliminated. The results also establish that the combination of ANN and GA forms a superior tool for optimization.

Conclusions

The performance of the MHE has been determined using FEM for a limited number of parameters. Using this limited data, an ANN is trained to generate additional data for other parameters. ANN used in the analysis accurately predicts the effectiveness of the MHE due to the fact that there is no scattered input data. By introducing the neuro-genetic optimization methodology, ANN is combined with GA to find the maximum effectiveness of the MHE achievable in the given range of input parameters. GA improves the performance of the MHE by proposing a new set of design parameters. The maximum effectiveness obtained through GA is 59.191 percent as opposed to the effectiveness of 24.776 percent obtained from FEM simulation thus yielding an improvement of almost 35 percent. This showed that the combination of ANN and GA forms a superior tool for the optimization of the MHE.

References

Baratti, R., Cannas, B., Fanni, A., Pintus, M., Sechi, G.M. and Toreno, N. (2003), "River flow forecast for reservoir management through neural networks", *Neurocomputing*, Vol. 55, pp. 421-37.

Ferreira, P.M., Faria, E.A. and Ruano, A.E.B. (2001), "Neural network models in greenhouse air temperature prediction", *Neurocomputing*, Vol. 43, pp. 51-75.

Goldberg, D.E. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Boston, MA.

Harris, C., Despa, M. and Kelly, K. (2000), "Design, and fabrication of a cross flow micro heat exchanger", *Journal of Microelectromech. Syst.*, Vol. 9, pp. 502-8.

Harris, C., Kelly, K., Wang, T., McCandless, A. and Motakef, S. (2002), "Fabrication, modeling, and testing of micro-cross-flow heat exchangers", *Journal of Microeletromech. Syst.*, Vol. 11, pp. 726-35.

Jain, L. and Vemuri, V. (1999), "Industrial applications of neural networks", *Computational Intelligence*, CRC Press, Boca Raton, FL.

Leong, C.H., Azid, I.A. and Seetharamu, K.N. (2002), "Thermal performance prediction of plastics ball grid array (PBGA) using artificial neural network", *ASEAN Journal of Science and Technology for Development*, Vol. 19 No. 1, pp. 29-38.

Lewis, R.W., Nithiarasu, P. and Seetharamu, K.N. (2004), *Fundamentals of Finite Element Method in Heat and Fluid Flow*, Wiley, New York, NY.

Lewis, R.W., Morgan, K., Thomas, H.R. and Seetharamu, K.N. (1996), *The Finite Element Method in Heat Transfer Analysis*, Wiley, New York, NY.

Mitchell, M. (1996), *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, MA.

Ranganayakulu, Ch., Seetharamu, K.N. and Sreevatsan, K.V. (1995), "Finite element analysis of compact plate-fin and tube-fin heat exchanger", PhD dissertation, School of Mechanical Engineering, University of Science Malaysia, Penang.

Segerlind, L.J. (1984), *Applied Finite Element Analysis*, Wiley, New York, NY.

About the authors

H.W. Lee received the BEng (Mechanical Engineering) in 2004 from University of Science (USM), Malaysia. He graduated with first class honours and ranked second in his class. He is currently a Master Student in School of Mechanical Engineering, USM, Malaysia. His research interests include thermal management of MEMS and application of ANN and GA for optimization. E-mail: hingwah717@yahoo.com; hing_wahl2@asme.org

Y.J. Teng received the BEng (Mechanical Engineering) in 2004 from University of Science (USM), Malaysia. He graduated with first class honours and ranked top in his class. He is currently a Researcher at SIRIM. His research interests include thermal management of heat exchangers and solder joint reliability. E-mail: jengteng@yahoo.com

I.A. Azid received the BSc degree in Mechanical Engineering from Clarkson University, Potsdam, NY, in 1992, the MSc degree from University of Wales, Swansea, UK, in 1995, and the PhD from University of Wales, Cardiff, UK, in 1999. He is currently an Associate Professor at the School of Mechanical Engineering, Universiti Sains Malaysia, Penang, Malaysia. He has published more than 40 papers in journals and conferences. His research interests include application of artificial intelligence in electronic packaging, thermal management in MEMS and structural optimization. He is the Vice-Chairman of IEEE-CPMT Malaysia Chapter. I.A. Azid is the corresponding author and can be contacted at: ishak@eng.usm.my

K.N. Seetharamu received the BE degree from Mysore University, Mysore, India, in 1960; the ME degree from the Indian Institute of Science, Bangalore, in 1962, and the PhD degree from Indian Institute of Technology (IIT), Madras, in 1973. He was a Professor of Mechanical Engineering at IIT, Madras, for almost three decades before joining Universiti Sains Malaysia, Malaysia where he served as a Professor in the School of Mechanical Engineering from 1996 to 2005. He has published more than 126 papers in international journals and 200 papers in international and national conferences. He co-authored *Finite Element Method in Heat Transfer Analysis* in 1996 and *Fundamentals of Finite Element Method in Heat and Fluid Flow* (New York, NY: Wiley, 2004), authored *Engineering Fluid Mechanics* (New Delhi: Narosa, 2004), contributed a chapter on "Fundamentals of thermal management" in *Fundamentals of Microsystem Packaging* (New York, NY: McGraw-Hill). His research interest is in the areas of fluid flow, heat transfer, stress analysis, electronic packaging, biomechanics and FEM simulation. He has engaged actively in the area of electronic packaging including MEMS. He is also engaged in the application of ANN and GA to the engineering problems. Professor Seetharamu was the Founder President of IEEE CPMT Chapter of Malaysia and the Secretary/Treasure of IMAPS Malaysia Chapter. He is responsible for starting both the IEEE-CPMT and IMAPS chapters in Malaysia. E-mail: knseetharamu@yahoo.com