Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

International Journal of Heat and Mass Transfer 50 (2007) 2089–2100

International Journal of HEAT and MASS TRANSFER

www.elsevier.com/locate/ijhmt

Solution of inverse heat conduction problems using Kalman filter-enhanced Bayesian back propagation neural network data fusion

S. Deng *, Y. Hwang

Department of Weapon System Engineering, Chung Cheng Institute of Technology, National Defense University, No. 190, Sanyuan 1st St., Dashi Jen, Taoyuan 33509, Taiwan, ROC

Received 9 April 2006; received in revised form 8 November 2006
Available online 22 January 2007

Abstract

This paper presents an efficient technique for analyzing inverse heat conduction problems using a Kalman Filter-enhanced Bayesian Back Propagation Neural Network (KF-B^{2}PNN). The training data required for the KF-B^{2}PNN are prepared using the Continuous-time analogue Hopfield Neural Network and the performance of the KF-B^{2}PNN scheme is then examined in a series of numerical simulations. The results show that the proposed method can predict the unknown parameters in the current inverse problems with an acceptable error. The performance of the KF-B^{2}PNN scheme is shown to be better than that of a stand-alone Back Propagation Neural Network trained using the Levenberg–Marquardt algorithm.

Keywords: Inverse heat conduction problem (IHCP); Back propagation neural network (BPNN); Kalman filtering (KF); Data fusion

1. Introduction

In forward heat conduction problems, the internal temperature field is derived from the known heating characteristics, boundary conditions and initial conditions of the body of interest. Conversely, in inverse heat conduction problems (IHCPs), experimental temperature measurements are taken at boundary points or various other points in the interior of the body and then used to estimate the unknown boundary conditions at the external surface. IHCPs are mathematically ill-posed in the sense that the existence, uniqueness and stability of their solutions cannot be assured [1]. Hence, they are generally solved using some form of numerical technique. Classical approaches include space marching [2,3] the single future time step method [1, pp. 115–119], the function specification method [4,1, pp. 119–134], the regularization method [1, pp. 134–145], and the trial function method [1, pp. 134–145]. However, as the capabilities of computer science and technology have advanced, an increasing number of researchers have successfully solved IHCPs using numerical methods such as the Finite Element Method [5], the Finite Difference Method (FDM) [6], or the Boundary Element Method [7]. In a recent development, the emergence of artificial neural network technology has led to an entirely new approach for the solution of IHCPs [8–11].

This study applies the data fusion technique to solve IHCPs. In general, data fusion approaches can be classified as one of two main types, namely parametric or information theoretic [12,13]. A typical parametric method is Kalman filtering (KF), while a typical information theoretic method is that of neural networks. However, both methods have certain limitations. For example, KF requires the system dynamics to be completely known and modeled as a Gauss–Markov stochastic process. Furthermore, the statistics of the system error and the observation error are assumed to be normally distributed. However, many practical systems fail to conform fully to these conditions. Meanwhile, the limitations of the neural network approach include the fact that their successful implementation

* Corresponding author.
E-mail addresses: sgdeng@ccit.edu.tw (S. Deng), g960405@ccit.edu.tw (Y. Hwang).

0017-9310/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijheatmasstransfer.2006.11.019

Nomenclature

C_i	amplifier input capacitance of *i*th neuron
CHNN	Continuous-time analogue Hopfield Neural Network
d_j	desired output of *j*th neuron
$f(\cdot)$	neuron activation function
H	measurement matrix
I_i	external input to *i*th neuron
k	time (discretized)
K	Kalman gain
KF-B^{2}PNN	Kalman filter-enhanced Bayesian back propagation neural network
L	thermal layer thickness
q	heat flux
Q	process noise covariance
R	measurement noise covariance
R_i	resistance of *i*th neuron
t	time (continuous)
T	temperature
$\widehat{T}_i$	difference between temperature of *i*th neuron and its neighbor
ΔT	sampling interval
u_i	internal state of *i*th neuron
w_{ij}	connection strength between neurons *j* and *i*
w_g	process noise vector
$\mathbf{W}$	coefficient matrix
X	state vector
y_j	output of *j*th neuron
Z	observation vector
Γ	input matrix
α	thermal diffusivity
β	steepness of sigmoid function
δt	sampling interval time
δ	direct delta function
v	measurement noise vector
θ_k	bias of neural network
κ	thermal conductivity
o	null matrix
ρ	total shunt capacitance
ϕ	state transition matrix
Subscripts	
i, j	indices
Superscripts	
T	transpose of matrix
$\sim$	dimensionless value

depends on providing them with extensive artificial experience. Furthermore, a stand-alone neural network system is not easily applied to some practical engineering problems, and the attainable estimation accuracy is not always sufficiently precise. Since both the parametric and the information theoretic approaches encounter certain irresolvable implementation problems, an alternative approach must be sought. Chin proposed a hybrid system which combined the KF scheme with a neural network to track a non-maneuvering target [14]. Subsequently, other researchers proposed various neural network-aided Kalman filtering and data fusion schemes for target tracking applications [15,16]. More recently, a KF and neural network-based scheme was proposed for identifying icing in the A340 aircraft [17].

This paper applies the KF to improve the performance of a Bayesian Back Propagation Neural Network (B^{2}PNN) in solving IHCPs. The proposed scheme combines the powerful estimation capabilities of the adaptive KF and the strong learning capabilities of the BPNN. Meanwhile, the Bayesian regularization method is applied to improve the weak generalization capabilities of general Back Propagation (BP) algorithms when applied to nonlinear function approximations and to take account of the uncertain noise inherent in the current IHCPs.

This study commences by applying the CHNN scheme to the forward heat conduction analysis of a one-dimensional cylindrical coordinate system. The results of the forward analysis are then used as training data for a three-layered B^{2}PNN integrated with an adaptive KF scheme (KF-B^{2}PNN). The trained KF-B^{2}PNN scheme is employed to solve various IHCPs with different heat profiles. To demonstrate the performance of the proposed method, the conventional stand-alone BPNN is also applied to solve the same set of IHCPs. The stand-alone BPNN is trained using eight different algorithms. The relative performance of each algorithm is evaluated in terms of the convergence rate obtained by the BPNN and the accuracy of the final solutions. These algorithms include: (1) conjugate gradient back propagation with resilient back propagation (CRB) [18], (2) gradient descent with momentum and adaptive learning rate back propagation (GMB) [19], (3) Levenberg–Marquardt back propagation (LMB) [20], (4) conjugate gradient back propagation with Fletcher–Reeves updates (CBF) [21], (5) scaled conjugate gradient back propagation (SCB) [21], (6) quasi-Newton back propagation (QNB) [20, pp. 242], (7) one-step Secant back propagation (OSB) [22], and (8) conjugate gradient back propagation with Powell–Beale restarts (CBP) [23]. The performance of the stand-alone BPNN trained using the best training algorithm is then compared with that of the proposed KF-B^{2}PNN scheme when applied to solve the same set of IHCPs.

2. Formulation of forward heat conduction problems

2.1. One-dimensional cylindrical coordinate system

This study commences by constructing the homogeneous differential equation for the one-dimensional heat conduction case. In this case, it is assumed that the one-dimensional hollow cylinder has a thermally insulated surface at $r = b$ and that a time-varying heat flux, $q(t)$, acts at the inner surface, i.e., at $r = a$. The boundary and initial conditions are indicated in the figure. The corresponding system state equations can be expressed in dimensionless form as [24]:

$$\frac{\partial \widetilde{T}}{\partial \tilde{t}} = \frac{\partial^2 \widetilde{T}}{\partial \tilde{r}^2} + \frac{1}{\tilde{r}}\frac{\partial \widetilde{T}}{\partial \tilde{r}}, \quad a \leqslant \tilde{r} \leqslant b, \quad \tilde{t} > 0, \tag{1a}$$

$$\widetilde{T}(\tilde{r}, 0) = c_0, \quad a \leqslant \tilde{r} \leqslant b, \tag{1b}$$

$$\frac{\partial \widetilde{T}}{\partial \tilde{r}} = -\tilde{q}(\tilde{t}), \quad \tilde{r} = a, \quad \tilde{t} > 0, \tag{1c}$$

$$\frac{\partial \widetilde{T}}{\partial \tilde{r}} = 0, \quad \tilde{r} = b, \tag{1d}$$

where:

$$\widetilde{T} = \frac{T - T_0}{q_0 L/\kappa}, \quad \tilde{q} = \frac{q}{q_0}, \quad \tilde{r} = \frac{r}{L}, \quad \tilde{t} = \frac{\alpha t}{L^2},$$
$$L = b - a = 1 \tag{1e}$$

in which c_0 is the uniform initial temperature, α the thermal diffusivity, κ the thermal conductivity, q_0 the nominal value of the surface heat flux and $\tilde{q}(\tilde{t})$ the dimensionless heat flux. Note that for convenience, the superscript ($\tilde{}$) is omitted throughout the remainder of this paper.

2.2. Hopfield Neural Network

The Hopfield continuous-time dynamic neural network constructed from n dynamic neural units is described by the following non-linear differential equations [25,26]:

$$C_i \frac{\mathrm{d}u_i(t)}{\mathrm{d}t} = \sum_{\substack{j=1 \\ i \neq j}}^{n} w_{ij} y_j(t) - \frac{u_i(t)}{R_i} + I_i(t), \quad i = 1, 2, \ldots, n \tag{2a}$$

$$y_i(t) = f(u_i(t)), \quad i = 1, 2, \ldots, n \tag{2b}$$

where u_i is the internal state of the ith neuron, $\frac{1}{R_i} = \sum_{j=1}^{n} w_{ij} + \frac{1}{\rho_i}$, $\frac{1}{R_{ij}} = w_{ij}$, w_{ij} is the strength of the connection between neurons j and i, y_j is the output of the jth neuron, $f(\cdot)$ is the neuron activation function, I_i is the external input to neuron i, and y_i is the output signal from each neuron, including the ith neuron.

The neuron's output, i.e., $y_i = f(u_i)$, is a non-decreasing function of the activation level. Hopfield network implementations generally employ a sigmoid activation function such as the tanh function or a piecewise linear approximation to a sigmoid. The hyperbolic tangent sigmoid function has the form:

$$f(u_i) = \tanh(\beta u_i) = (\mathrm{e}^{\beta u_i} - \mathrm{e}^{-\beta u_i})(\mathrm{e}^{\beta u_i} + \mathrm{e}^{-\beta u_i})^{-1} \tag{3}$$

This sigmoid function is rotationally symmetric about the origin of the coordinate axes, and asymptotically approaches limiting points of $(\infty, 1)$ and $(-\infty, -1)$. The parameter β in the sigmoid function is adjustable according to the non-linearity of the problem. In this paper, a value of $\beta = 1/2$ is used in the simulations.

2.3. Connectivity structure

In this study, the numerical solutions to the forward heat conduction problems are obtained using the CHNN scheme. The detailed derivations of this technique are presented in [27]. This study considers the temperature distributions of heat conduction models with boundary conditions and with some of the connective weight strengths equal to zero. Therefore, the basic CHNN circuit diagram can be modified to the form presented in Fig. 1. In this figure, the function $f(t)$ represents the conversion between the temperature and the corresponding voltage, and the neuron weight strengths, W, are the inverse of the resistance values of the CHNN. In practice, the temperature distribution weight strengths are variable. Therefore, during the simulation process, a voltage converter is used to control the current source in order to introduce non-linear weight strengths throughout the circuit.

3. Formulation for inverse problems

3.1. Back propagation neural network

Fig. 2 illustrates a general BP neural network. As shown, this network is a feed forward, fully-connected hierarchical M-layered network consisting of an input layer, M-2 hidden layers and an output layer. If the kth unit in the Mth layer is denoted by (M, k), the state variable u_k^M for this unit, and its output signal y_k^n to the units in the next layer $(u_k^M + 1, k)$, can be written, respectively, as:

$$u_k^M = \sum_k (w_{k,j}^{M,M-1} u_j^{M-1} + \theta_k^M) \tag{4}$$

$$y_k^M = f_k^M(u_k^M) \tag{5}$$

where $w_{k,j}^{M,M-1}$ is the connection strength between units (M, k) and $(M-1, j)$, and θ_k^M and $f_k^M(\cdot)$ are the bias and activation functions, respectively, of unit (M, k). Note that the output signal y_k^M is transmitted to all units in the next (i.e., $(M+1)$-th) layer.

In the current study, a stand-alone BPNN and a KF-B^{2}PNN scheme are used to solve various IHCPs. In the solution procedure, the input parameters to the neural network, i.e., $u^0 = (u_1^0, u_2^0, \ldots)$, are temperature data at specified points in the interior of the object of interest, while the outputs of the network, i.e., $y^M = (y_1^M, y_2^M, \ldots)$, are parameters relating to the boundary conditions, e.g., the heat flux.

3.2. BPNN training

Using a BPNN to solve IHCPs involves two basic stages, a learning stage, and a recalling stage. The learning

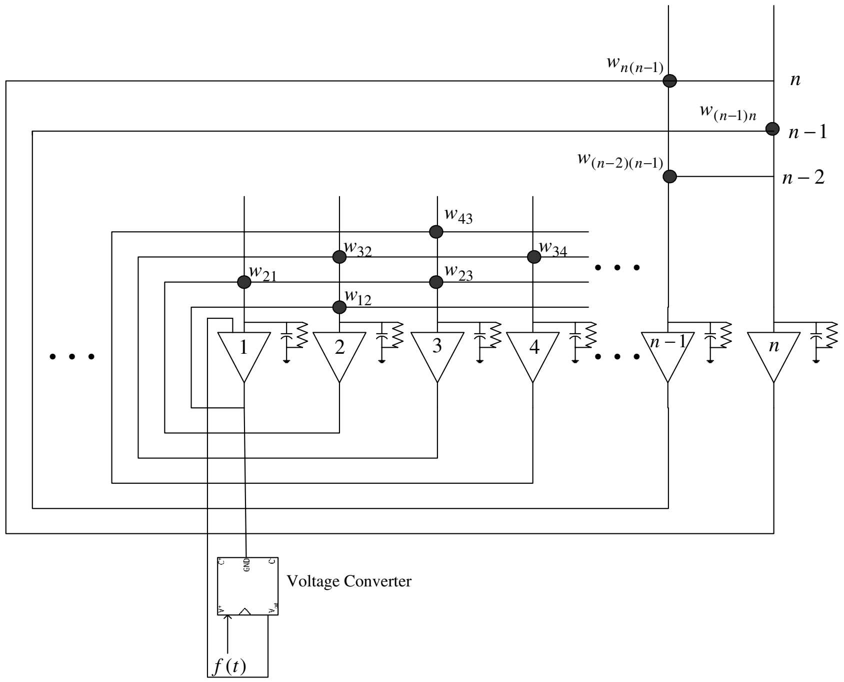

Fig. 1. Modified Hopfield Neural Network connectivity circuit diagram for heat conduction temperature profiles.

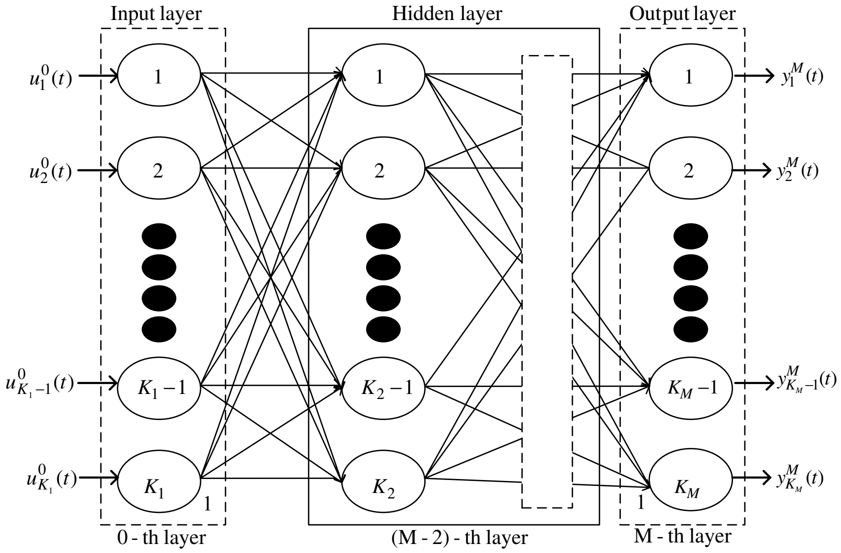

Fig. 2. Structure of multilayered neural network.

stage is essentially a supervised learning process conducted using a training model and a set of target output values chosen from the problem domain. The goal of the training process is to modify the connection strengths, $w_{k,j}^{M,M-1}$, and biases, θ_k^M, of the BPNN iteratively until the actual output vector $y^M = (y_1^M, y_2^M, \ldots)$ lies within an acceptable error of the target output vector, $d^M = (d_1^M, d_2^M, \ldots)$. If the training data set fully covers the problem space, and training is successfully completed, the neural network will be capable of providing accurate outputs for any arbitrary unknown input in the subsequent recalling stage by applying the connection strengths and biases established during the learning stage.

Conventionally, a BP algorithm adjusts the connection weights iteratively using the steepest descent technique. However, the resulting convergence is inherently slow and the solution may become trapped at local minima. Since the introduction of the original BP learning algorithm, extensive research has been conducted to improve the convergence rate. This research has broadly followed one of two different directions, namely (1) the development of ad hoc techniques [18], such as varying the learning rate or using momentum and rescaling variables, and (2) the application of standard numerical optimization techniques [19–23]. All of the proposed techniques improve the performance of the traditional BPNN to a greater or lesser extent, and ultimately, the choice of the most appropriate algorithm depends on the complexity of the target problem, the volume of training data available, the size of the network, and the acceptable error tolerance. This study overcomes the weak generalization capabilities of BPN networks when applied to the solution of non-linear function approximations by employing the Bayesian regularization algorithm. The details of this algorithm are well documented in the literature [28,29], and hence are not repeated here.

3.3. KF procedure

The one-dimensional hollow cylinder system process equation is given by:

$$Z(t) = T(r,t) + v(t), \quad r = b, \quad t > 0 \tag{6}$$

where $Z(t)$ is the noise-corrupted temperature measurement, in which $v(t)$ is the measurement noise with an assumed zero mean and white Gaussian distribution. The state equation discretized over time intervals of length ΔT has the form:

$$\mathbf{X}(k+1) = \phi\mathbf{X}(k) + \Gamma\lfloor \mathbf{q}(k) + \mathbf{w_g}(k)\rfloor \tag{7}$$

where

$$\mathbf{X}(k) = [\mathbf{T}_1(t)\mathbf{T}_2(t)\cdots\mathbf{T}_{M\text{-}1}(t)\mathbf{T}_M(t)]^{\mathrm{T}} \tag{8a}$$

$$\phi = \mathrm{e}^{\mathbf{W}\Delta T} \tag{8b}$$

$$\Gamma = \int_{k\Delta T}^{(k+1)\Delta T} \exp\{\mathbf{W}[(k+1)\Delta T - \tau]\}\Psi \mathrm{d}\tau \tag{8c}$$

In the above, $\mathbf{X}(k)$ represents the state vector, ϕ is the state transition matrix, Γ is the input matrix, $q(k)$ is the sequence of deterministic inputs, and $\mathbf{w_g}(k)$ is the process noise vector, which is assumed to have a zero mean and white Gaussian distribution with a variance of $E\{\mathbf{w_g}(k)\mathbf{w_g}^{\mathrm{T}}(j)\} = Q\delta_{kj}$, where δ_{kj} is a Dirac delta function.

The noise-corrupted temperature measurement equation given in Eq. (6) can be expressed in matrix form as:

$$\mathbf{Z}(k) = \mathbf{H}\mathbf{X}(k) + v(k) \tag{9}$$

where $\mathbf{Z}(k)$ is the observation vector at time k, $\mathbf{H} = [0 \quad 0 \quad \ldots \quad 1]$ is the measurement matrix, and $v(k)$ is the measurement noise vector, which is assumed to have a zero mean and a white Gaussian distribution with a variance of $E\{v(k)v^{\mathrm{T}}(j)\} = R\delta_{kj}$.

3.4. KF-B^{2}PNN data fusion scheme

The KF-B^{2}PNN data fusion scheme proposed in this study is designed to obtain highly accurate solutions to IHCPs. The three parameters known to have a direct influence on the solution errors are taken as inputs [14–17], i.e., the Kalman gain $K(k)$, the difference between the measured temperature $Z(k)$ and the estimated temperature $\widehat{X}(k)$, and the difference between the predicted temperature $\widehat{X}(k \mid k-1)$ and the estimated temperature. Since the exact state $X(k)$ is known (as far as simulation is concerned), the supervised learning algorithm can use the signal $Error = X(k) - \widehat{X}(k)$ (i.e., the difference between the known state and the estimated state) to train the network. During operation, the output of the neural network (*Error*) is used to correct the state estimate.

4. Framework of forward and inverse techniques

The test cases presented in this study consider boundary heat fluxes with a variety of both simple and complex waveforms. The solutions of the forward heat conduction problems obtained using the CHNN method are then used in the subsequent IHCP analysis performed using a stand-alone BPNN and the proposed KF-B^{2}PNN scheme to estimate the boundary conditions. Initially, this study considers three conventional heat conduction problems, namely a "triangular" initial temperature profile in a bar, a time-varying heat flux with a triangular profile, and a surface subjected to a constant flux. In the first two cases, the CHNN results are compared to the exact solutions in order to verify the accuracy of the proposed CHNN scheme. In the third case, the CHNN solutions are compared with those of the Finite Difference Method. Having confirmed the accuracy of the CHNN method in solving conventional forward heat conduction problems, the scheme is applied to solve three rather more complicated time-varying boundary heat flux problems.

In the current forward and inverse neural network analysis procedure, the B^{2}PNN algorithm used to solve the current IHCPs is designed with 6 nodes in the input layer, 15 nodes in the hidden layer, and 6 nodes in the output layer. The nodes in the hidden layer are assigned a sigmoid activation function, while those in the output layer have a linear activation function.

In the current analysis procedure, the solutions obtained by the CHNN scheme for the forward heat conduction problems provide temperature data corresponding to a number of specified points inside the domain. These temperature data are then used as the input to the inverse KF-B^{2}PNN procedure used to estimate the boundary conditions. To reflect actual engineering problems, a measurement noise of 10E–4 is added to the results of the temperature data to use as inputs to the BPN. Fig. 3 presents a flowchart of the forward and inverse neural network analysis procedure carried out in this study. As shown, respective pairs of the boundary conditions and the temperature data calculated at specified points in the domain by the CHNN are used to train the KF-B^{2}PNN using various training methods. The forward heat conduction problems are solved for arbitrarily assigned boundary conditions to obtain temperature distributions in the domain of interest. The boundary conditions for a given set of calculated temperature data are then predicted using the trained neural network. Finally, the predicted boundary conditions are compared with the arbitrarily assigned boundary conditions in the previous step to confirm the accuracy of the proposed approach.

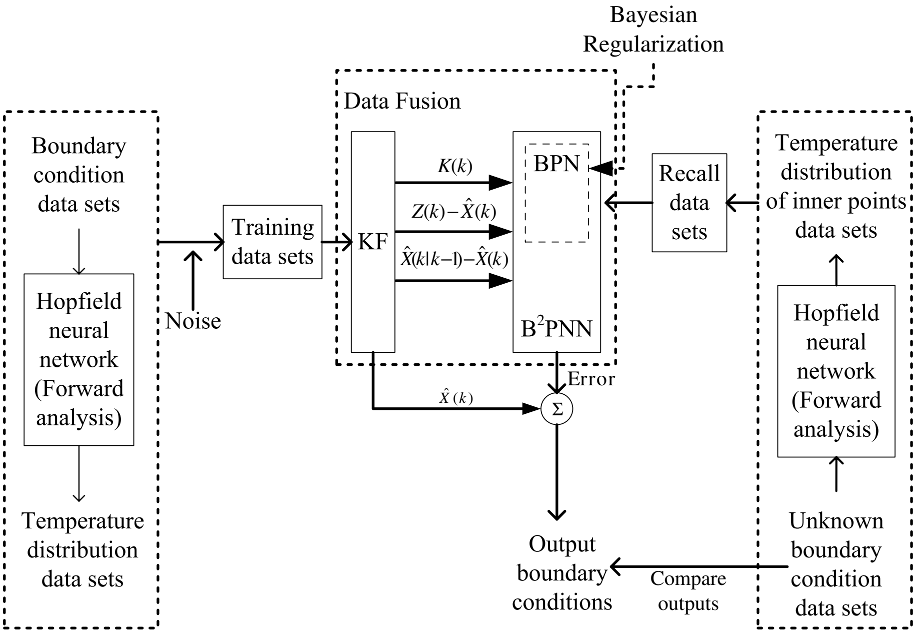
Bayesian
Regularization
Data Fusion
Boundary condition data sets
Hopfield neural network (Forward analysis)
Temperature distribution data sets
Noise
Training data sets
KF
$K(k)$
$Z(k)-\hat{X}(k)$
$\hat{X}(k|k-1)-\hat{X}(k)$
BPN
B^2PNN
Recall data sets
Temperature distribution of inner points data sets
Hopfield neural network (Forward analysis)
Unknown boundary condition data sets
Error
$\hat{X}(k)$
Σ
Output boundary conditions
Compare outputs

Fig. 3. Flow chart of forward and inverse neural network analysis.

In evaluating the performance of the neural network, the accuracy of the outputs obtained under each of the different training methods was evaluated using the root mean square error (MSE) indicator, i.e.,

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i^{\mathrm{mea}} - d_i^{\mathrm{exa}})^2 \tag{10}$$

where y^{mea} is the neural network solution at time t, d^{exa} is the analytical solution or desired output for the same input data at time t, and n is the number of measurements.

5. Results and discussion

5.1. Forward problems

This study considered a total of six different heat flux profiles. In the first two cases, the CHNN results were compared to the exact solutions to verify the accuracy of the proposed CHNN scheme. In the third case, the solutions of the CHNN scheme were compared to those of the FDM method. Having confirmed its performance in solving conventional heat conduction problems, the CHNN method was then applied to three more complicated time-varying boundary heat flux problems. Again, the solutions of the CHNN scheme and the FDM method were compared. The six heat flux profiles were as follows:

(a) *Case 1*: A triangular temperature profile within a laterally insulated bar of length L, whose ends are maintained at zero temperature. The initial temperature is given by:

$$f(x) = \begin{cases} x & 0 < x < L/2 \\ L - x & 2/L < x < L \end{cases} \tag{11a}$$

The exact solution is [30]:

$$T(x,t) = \frac{4L}{\pi^2}\left\{\sin\frac{\pi x}{L}\exp\left[-\left(\frac{c\pi}{L}\right)^2 t\right] - \frac{1}{9}\sin\frac{3\pi x}{L} \times \exp\left[-\left(\frac{3c\pi}{L}\right)^2 t\right] + - \cdots\right\} \tag{11b}$$

(b) *Case 2*: A time-varying heat flux with a triangular profile. Prior to $t^+ = 0$, the heat flux is zero. When t^+ lies between zero and 0.6, the surface flux, q, increases linearly with time. For $t^+ > 0.6$, the flux decreases linearly to zero at $t^+ > 1.2$ and remains at zero thereafter. The exact solutions for the temperature at $x^+ = 0$ and $x^+ = 1$ for the linear heat flux are given by [1, pp. 169–171]:

$$\phi^+(0,t^+) = \frac{1}{2}(t^+)^2 + \frac{1}{3}t^+ - \frac{1}{45} + \frac{2}{\pi^4}\sum_{n=1}^{\infty}\frac{1}{n^4}\exp(-\pi^2 n^2 t^+) \tag{12a}$$

$$\phi^+(1,t^+) = \frac{1}{2}(t^+)^2 - \frac{1}{6}t^+ + \frac{7}{360} + \frac{2}{\pi^4}\sum_{n=1}^{\infty}\frac{(-1)^n}{n^4}\exp(-\pi^2 n^2 t^+) \tag{12b}$$

$$T^+(x,t^+) = \begin{cases} \phi^+(x^+,t^+) \quad 0 < t^+ \leqslant 0.6 \\ \phi^+(x^+,t^+) - 2\phi^+(x^+,t^+ - 0.6) \\ \quad 0.6 < t^+ \leqslant 1.2 \\ \phi^+(x^+,t^+) - 2\phi^+(x^+,t^+ - 0.6) \\ \quad +\phi^+(x^+,t^+ - 1.2) \quad t^+ > 1.2 \end{cases} \tag{12c}$$

(c) *Case 3*: The initial temperature of a finite solid cylinder is zero and its surface is subjected to a constant flux. The temperature distribution in the cylinder is given by [31,1, pp. 17–18]:

$$T_a^+(r^+,t_a^+) = 2t_a^+ + \frac{1}{2}(r^+)^2 - \frac{1}{4} - 2\sum_{n=1}^{\infty} \mathrm{e}^{-\beta_n^2 t_a^+} \frac{J_0(r^+\beta_n)}{\beta_n^2 J_0(\beta_n)} \tag{13a}$$

where β_n, $n = 1, 2, \ldots$, are the positive roots of the Bessel function,

$$J_1(\beta_n) = 0 \tag{13b}$$

a is the cylinder radius, and

$$T_a^+(r^+,t_a^+) \equiv \frac{[T(r,t) - T_0]k}{q_c a}, \quad t_a^+ \equiv \frac{\alpha t}{a^2},\ r^+ \equiv \frac{r}{a} \tag{13c}$$

(d) *Case 4*: The time-varying flux has a step profile followed by a ramp profile (step-ramp profile) from $t = 0$ to $t = 6$ with a time interval of 0.01, i.e.,

$$q(t) = \begin{cases} 0 & 0 < t \leqslant 1 \\ 1 & 1 < t \leqslant 3 \\ \frac{1}{750}t - 0.2 & 3 < t \leqslant 6 \end{cases} \tag{14}$$

(e) *Case 5*: The time-varying flux has a triangular profile followed by a sine profile (triangular-sine profile) from $t = 0$ to $t = 6$ with a time interval of 0.01, i.e.,

$$q(t) = \begin{cases} 0.3 & 0.0 < t \leqslant 1.0 \\ 0.007t - 0.4 & 1.0 < t \leqslant 2.0 \\ -0.005t + 2 & 2.0 < t \leqslant 3.0 \\ 0.5 & 3.0 < t \leqslant 3.3 \\ 0.5\left(1 + \sin\left(\frac{\pi}{150}t\right)\right) & 3.3 < t \leqslant 5.8 \\ 0.3 & 5.8 < t \leqslant 6.0 \end{cases} \tag{15}$$

(f) *Case 6*: The time-varying flux has a sine profile followed by a second sine profile (sine–sine profile) from $t = 0$ to $t = 6$ with a time interval of 0.01, i.e.,

$$q(t) = \begin{cases} \sin(2\pi t/250) & 0 < t \leqslant 3 \\ 0.5\sin(2\pi t/250) + 0.5\sin(2\pi t/25) & 3 < t \leqslant 6 \end{cases} \tag{16}$$

Note that for reasons of heat flux rationality, only values greater than zero are applied.

5.2. Results of forward heat conduction problems

Figs. 4 and 5 compare the exact solutions and the CHNN solutions for Cases 1 and 2, respectively. The MSE for the CHNN results of Case 1 is 1.61E-06. For Case 2, the MSE at $x = 0$ is 1.32E-08 while at $x = 1$, the MSE is 1E-09. Eq. (12c) provides the numerical values of $T(0,t^+)$ for the case of a triangular time-varying heat flux. The corresponding results are presented in Fig. 5 for $\delta t^+ = 0.06$.

In Case 3, the initial temperature of the finite solid cylinder is zero and the surface is subjected to a constant flux. The resulting temperature distribution is solved using both the CHNN scheme and the FDM numerical technique. The MSE results for the two methods at $r = 0$ are found to be 5.23E-08 for the CHNN method and 4.27E-07 for the FDM method.

The results of Cases 1 to 3 indicate that the CHNN method provides accurate solutions for the temperature distribution in forward heat conduction problems, irrespective of the initial temperature or the nature of the heat flux

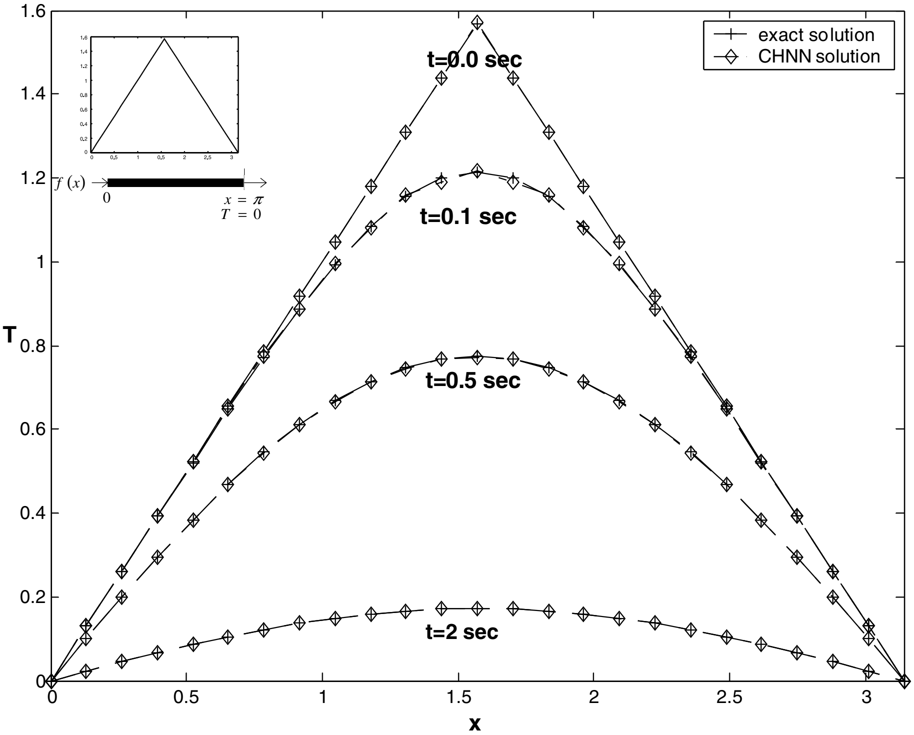

Fig. 4. Comparison of exact and CHNN solutions for initial triangular temperature distribution in laterally insulated bar of length π and $c = 1$.

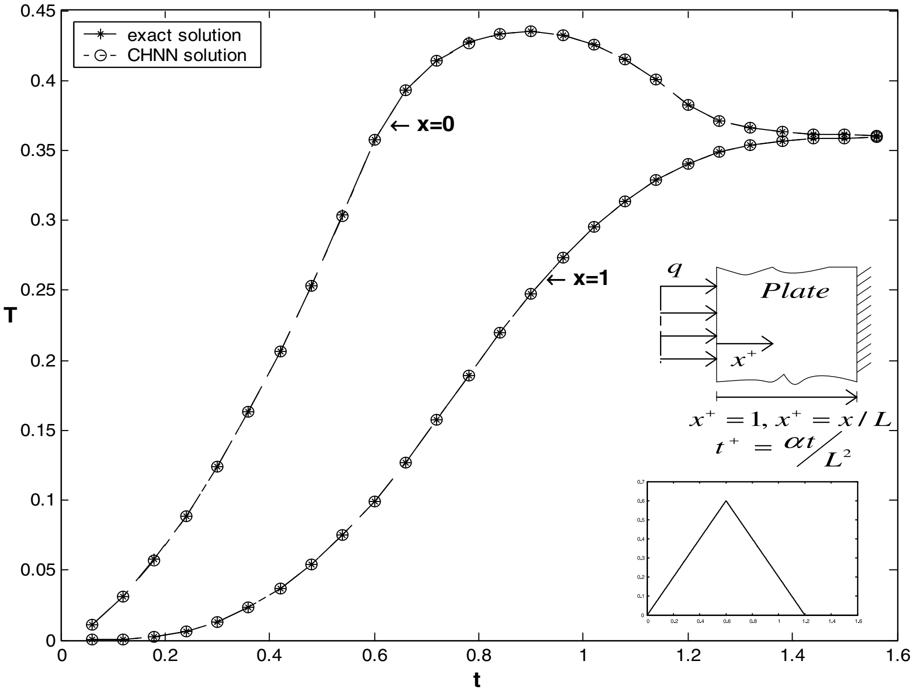

Fig. 5. Comparison of exact and CHNN solutions for temperature at insulated surface of finite plate heated with triangular time-varying heat flux.

variation at the boundary over time. The results of Cases 1 and 2 show that the solutions of the CHNN scheme are in good agreement with the exact results. Significantly, the results of Case 3 indicate that the performance of the CHNN method is slightly better than that of the FDM approach. Having confirmed the capability of the CHNN method in solving the standard heat conduction problems of Cases 1 to 3, the CHNN scheme was then applied to a series of heat conduction problems with more complicated heat flux profiles (Cases 4–6).

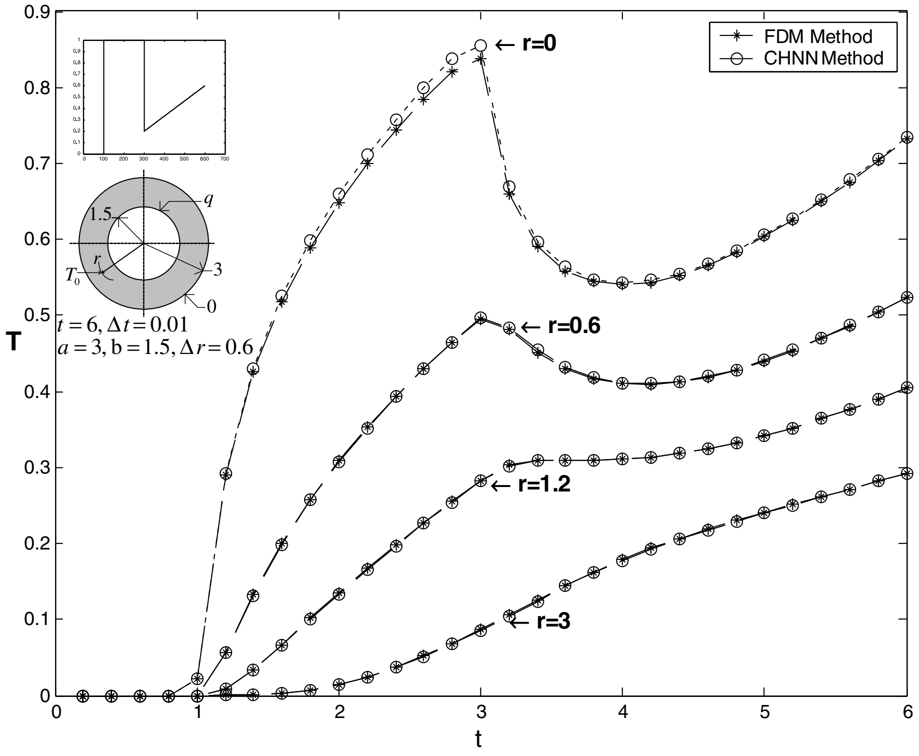

Fig. 6. Comparison of CHNN and FDM results for temperature with step-ramp heat flux.

5.3. Inverse heat conduction problems

In this study, the training data for the stand-alone BPNN were prepared by using the CHNN method to solve forward heat problems with boundary conditions of an arbitrarily assigned heat flux, q. Initially, the CHNN Eqs. (1a)–(1e) were discretized. In the discretization procedure, the calculation domain was divided into 6 nodal points,

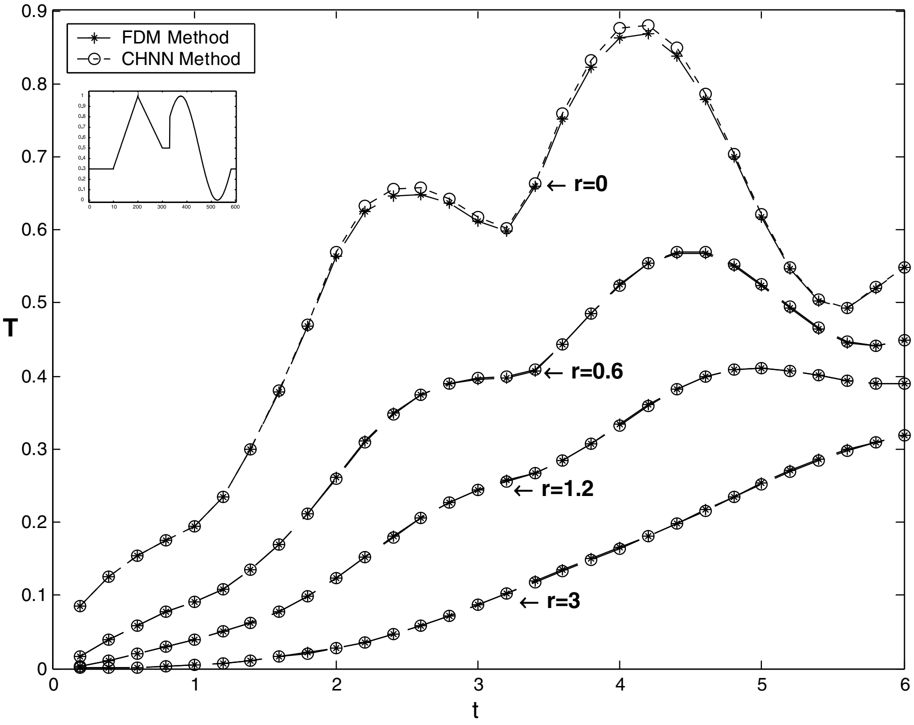

Fig. 7. Comparison of CHNN and FDM results for temperature with triangular-sine heat flux.

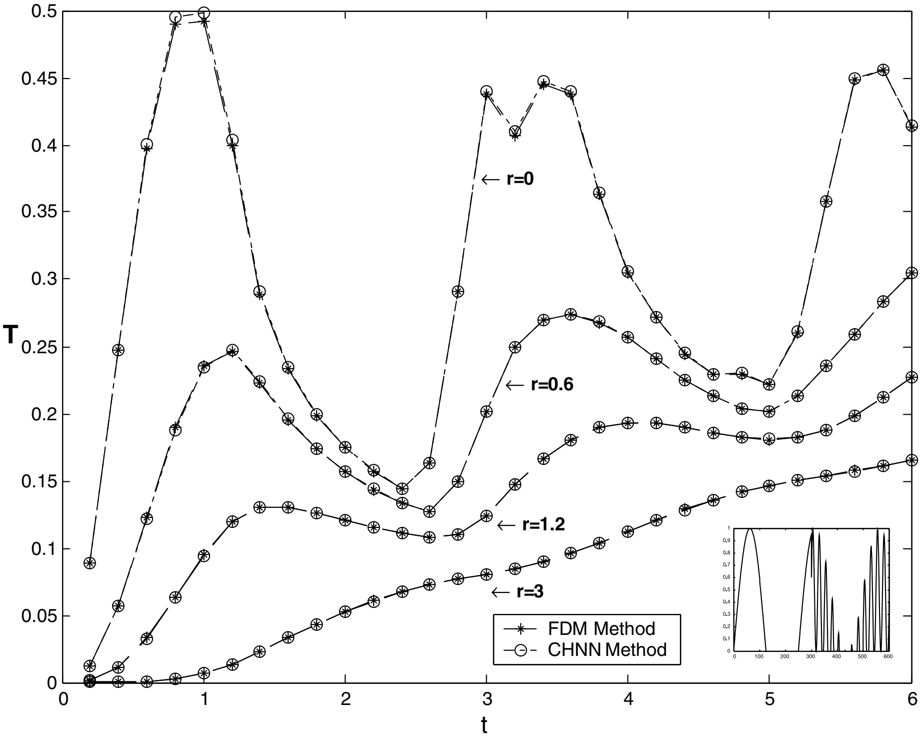

Fig. 8. Comparison of CHNN and FDM results for temperature with sine–sine heat flux.

values of the heat flux, q, and the temperature distribution, T, were prepared, and the value of the time constant was generated in the range 0–6 s. A total of 601 sets of time-history temperature data were obtained for each of the heat fluxes in Cases 4–6, shown in Figs. 6 and 7 (top left) and Fig. 8 (bottom right), respectively. These data were used as the boundary conditions for the forward problems. For each of the heat flux cases, a total of 2404 sets of temperature history data (corresponding to 4 different measurement positions) were calculated by the CHNN method, as indicated by the curves plotted in Figs. 6–8 by the circular symbols. These temperature data were then used to train the stand-alone BPNN and the KF- B^{2}PNN scheme.

The stand-alone BPNN assumed the following parameters: 15 units in the hidden layer, a learning rate of 0.0001, a momentum rate of 0.09, and 3000 convergence iterations. Network training was performed using 8 different algorithms, i.e., QNB, LMB, CRB, SCB, CBP, OSB, GMB, CBF. Table 1 compares the convergence characteristics of the 8 training algorithms when applied to Cases 4, 5 and 6 in the stand-alone BPNN environment. Note that the target error was specified as 0.001. For Cases 4 and 5, it can be seen that the QNB and LMB algorithms have a better convergence performance than the other algorithms and that the LMB scheme achieves the minimum MSE. For Case 6, the LMB algorithm again provides the best convergence performance. However, it not only fails to achieve the target error of 0.001, but also requires a large number of iterations.

Having completed the training stage, the LMB-trained stand-alone BPNN and the KF-B^{2}PNN scheme were applied to inversely solve Cases 4, 5 and 6. In the IHCP (recalling) stage for these three cases, a total of 300 sets

Table 1
Backpropagation neural networks to perform of eight training algorithms in terms of iterations and MSE for one-dimensional cases

Type		Case 4		Case 5		Case 6	
		Iterations	MSE	Iterations	MSE	Iterations	MSE
Algorithm	QNB	211	0.000981	216	0.001000	3000	0.003661
	LMB	56	0.000816	35	0.000972	3000	0.002882
	CRB	3000	0.001020	3000	0.001880	3000	0.034049
	SCB	964	0.000997	3000	0.000998	3000	0.024123
	CBP	500	0.000999	3000	0.001119	355	0.038943
	OSB	3000	0.001283	2105	0.000999	3000	0.035751
	GMB	3000	0.008931	3000	0.005648	3000	0.059163
	CBF	3000	0.001798	442	0.000999	472	0.040512

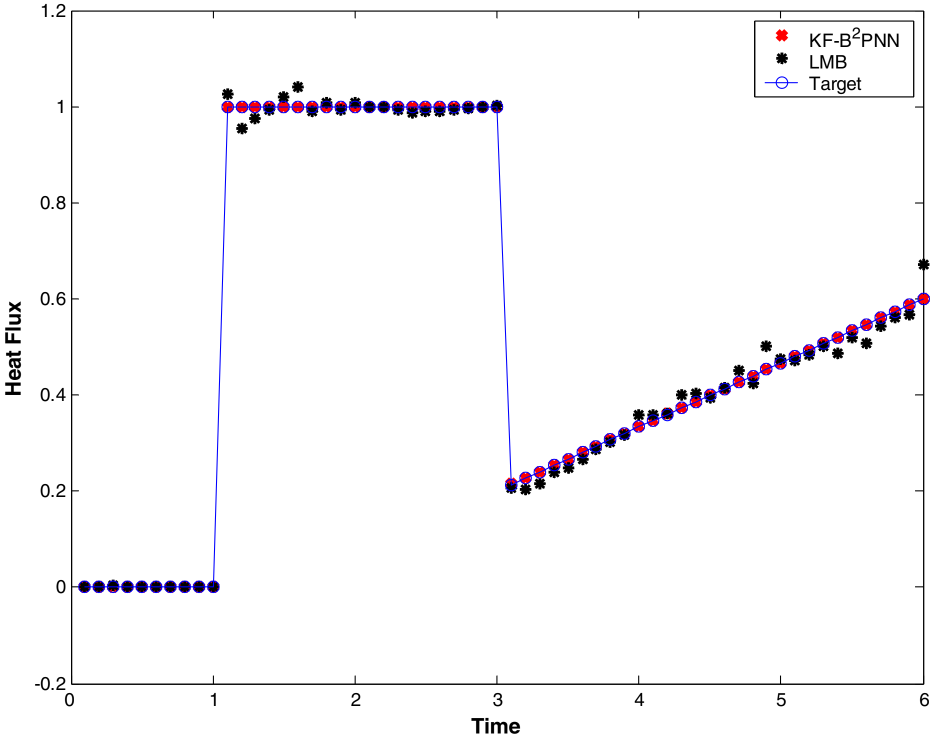

Fig. 9. Target and predicted heat fluxes of one-dimensional IHCP in Case 4.

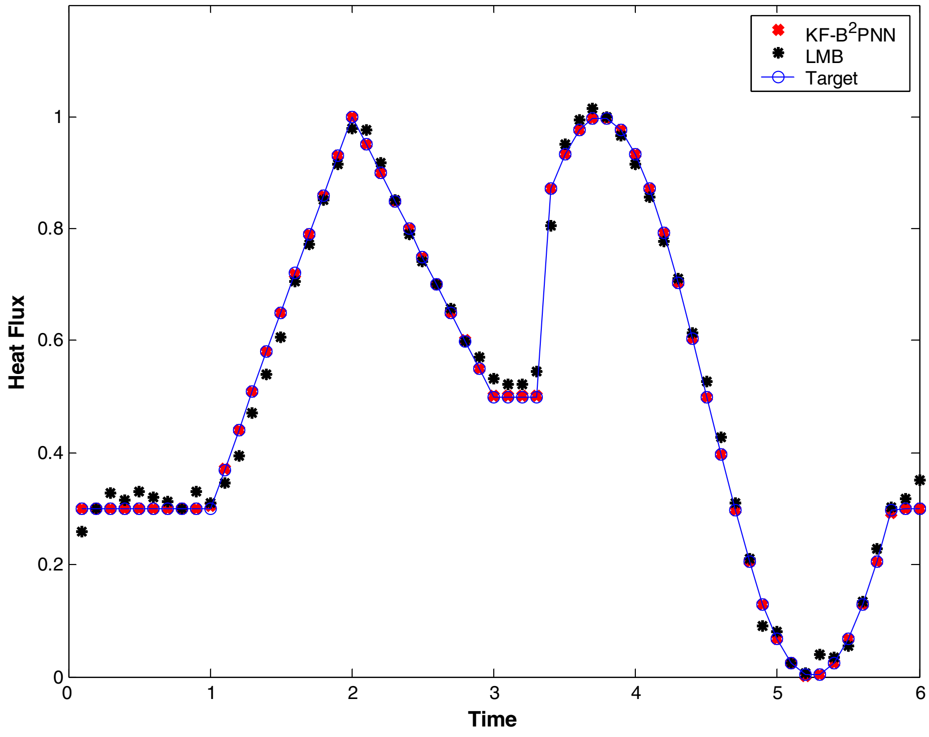

Fig. 10. Target and predicted heat fluxes of one-dimensional IHCP in Case 5.

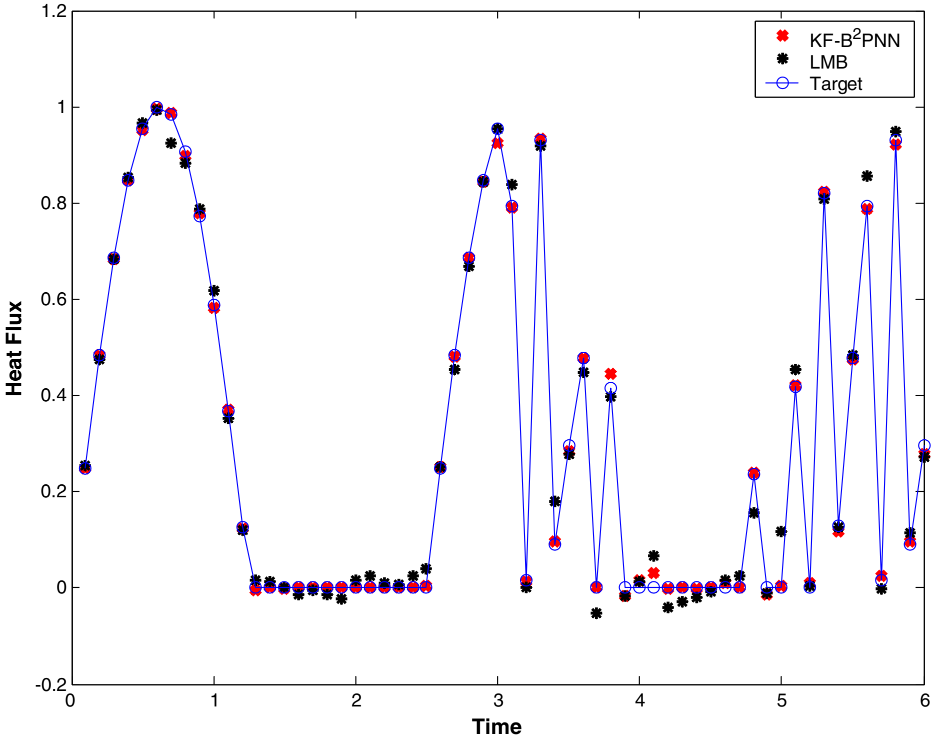

Fig. 11. Target and predicted heat fluxes of one-dimensional IHCP in Case 6.

of temperature history data (corresponding to 4 different measurement positions) were calculated by the CHNN method, and the target error was reduced to 0.00001. The MSE values of the outputs provided by the KF-B^{2}PNN scheme for Cases 4–6 were found to be 1.223E-11, 1.116E-11, and 3.190E-11, respectively, while those obtained from the LMB-trained network for the same cases were 8.811E-06, 1.243E-06, and 2.432E-06, respectively.

Figs. 9–11 show the heat flux profiles of Cases 4–6 (denoted by circular symbols) used as the boundary conditions for the forward problems solved by the CHNN scheme. Meanwhile, the outputs from the trained KF-B^2PNN scheme are indicated by the curves plotted using cross symbols, and those obtained from the LMB-trained network are indicated by the curves plotted using asterisk symbols. It is clear that the results of the KF-B^2PNN scheme are in better agreement with the target heat flux profiles than those of the LMB-trained network.

6. Conclusion

This paper has proposed a KF-B^2PNN scheme for the solution of IHCPs. The scheme integrates the standard Kalman filter with a Bayesian Back Propagation Neural Network. Initially, a Continuous-time analogue Hopfield Neural Network was applied to solve a number of forward heat conduction problems. The resulting temperature data were then used to train a conventional stand-alone BPNN and the KF-B^2PNN scheme. Through simulation, the performance of the proposed KF-B^2PNN scheme was evaluated and compared with that of the conventional Back Propagation Neural Network. The results indicated that the KF-B^2PNN scheme successfully achieves the target rms error of 0.00001 for the current IHCPs. Therefore, the proposed method is capable of predicting the unknown parameters in IHCPs with an acceptance error tolerance. Furthermore, its performance is superior to that of the conventional BPNN approach.

References

[1] J.V. Beck, B. Blackwell, C.R. St. Clair, Inverse Heat Conduction Ill-posed Problems, Wiley, New York, 1985, pp. 1–8.

[2] N. Al-Khalidy, Analysis of boundary inverse heat conduction problems using space marching with Savitzky–Gollay digital filter, Int. Commun. Heat Mass Transfer 26 (2) (1999) 199–208.

[3] M.S. Shin, J.W. Lee, Prediction of the inner wall shape of an eroded furnace by the nonlinear inverse heat conduction technique, JSME Int. J. B 43 (4) (2000) 544–549.

[4] D. Maillet, A. Degiovanni, R. Pasquetti, Inverse heat conduction applied to the measurement of heat transfer coefficient on a cylinder. Comparison between an analytical and a boundary element technique, J. Heat Transfer, Trans. ASME 113 (3) (1991) 549–557.

[5] T.C. Chen, P.C. Tuan, Input estimation method including finite-element scheme for solving inverse heat conduction problems, Num. Heat Transfer, Part B: Fundamentals 47 (3) (2005) 277–290.

[6] K.J. Dowding, J.V. Beck, A sequential gradient method for the inverse heat conduction problem (IHCP), J. Heat Transfer, Trans. ASME 121 (1999) 300–306.

[7] D. Lesnic, L. Elliott, D.B. Ingham, Application of the boundary element method to inverse heat conduction problems, Int. J. Heat Mass Transfer 39 (7) (1996) 1503–1517.

[8] V. Dumek, M. Druckmuller, M. Raudensky, K.A. Woodburg, Novel approaches to the IHCP: neural networks and expert systems, in: Proc 1 Int. Conf. Inverse Probl. Eng. Theory Pract., ASME, New York, 1993, pp. 275–282.

[9] M. Raudensky, J. Horsky, J. Krejsa, Usage of neural network for coupled parameter and function specification inverse heat conduction problem, Int. Commun. Heat Mass Transfer 22 (5) (1995) 661–670.

[10] J. Krejsa, K.A. Woodbury, J.D. Ratliff, M. Raudensky, Assessment of strategies and potential for neural networks in the inverse heat conduction problem, Inverse Probl. Eng. 7 (1999) 197–213.

[11] H. Shiguemori Elcio, S. Da Silva José DemÍSio, F. De Campos Velho Haroldo, Estimation of initial condition in heat conduction by neural network, Inverse Probl. Sci. Eng. 12 (3) (2004) 317–328.

[12] L.A. Klein, Sensor and Data Fusion Concepts and Applications, second ed., SPIE-International Society for Optical Engine, Bellingham, WA, 1999, pp. 47–81.

[13] R.C. Luo, M.G. Kay, Multisensor integration and fusion in intelligent systems, IEEE Trans. Syst., Man Cybernet. 19 (5) (1989) 901–931.

[14] L. Chin, Application of neural networks in target tracking data fusion, IEEE Trans. Aerospace Electron. Syst. 30 (1) (1994) 281–287.

[15] V. Vaidehi, N. Chitra, C.N. Krishnan, M. Chokkalingam, Neural network aided Kalman filtering for multitarget tracking applications, in: IEEE Proceedings of the Radar Conference, Waltham, MA, 1999, pp. 160–165.

[16] Y.C. Wong and M.K. Sundareshan, Data fusion and tracking of complex target maneuvers with a simplex-trained neural network-based architecture, in: IEEE World Congress on Computational Intelligence, Anchorage, AK, vol. 2, 1998, pp. 1024–1029.

[17] R. Aykan, C. Hajiyev, F. Caliskan, Kalman filter and neural network-based icing identification applied to A340 aircraft dynamics, Aircraft Eng. Aerospace Technol.: An Int. J. 77 (1) (2005) 23–33.

[18] M. Riedmiller, H. Braun, A direct adaptive method for faster backpropagation learning: the RPROP algorithm, in: Proceedings of the IEEE International Conference on Neural Networks, San Francisco, 1993, pp. 586–591.

[19] M.T. Hagan, B. Howard, M.H. Demuth, M.H. Beale, Neural Network Design, PWS Publishing Company, Boston, 2002 (Chapter 9 (pp. 15–19)).

[20] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall, New Jersey, 1998, p. 239.

[21] M.F. Moller, A scaled conjugate gradient algorithm for fast supervised learning, Neural Networks 6 (1993) 525–533.

[22] R. Battiti, First and second order methods for learning: between steepest descent and Newton's method, Neural Comput. 4 (2) (1992) 141–166.

[23] M.J.D. Powell, Restart procedures for the conjugate gradient method, Math. Program. 12 (1977) 241–254.

[24] P.C. Tuan, C.C. Ji, L.W. Fong, W.T. Huang, An input estimation approach to on-line two dimensional inverse heat conduction problems, Num. Heat Transfer B 29 (1996) 345–363.

[25] J.J. Hopfield, Neurons with graded response have collective computational properties like those of two-state neurons, Proc. Natl. Acad. Sci. USA 81 (1984) 3088–3092.

[26] J.J. Hopfield, D.W. Tank, Computing with neural circuits: a model, Science 233 (1986) 625–633.

[27] S. Deng, Y. Hwang, Applying neural networks to the solution of forward and inverse heat conduction problems, Int. J. Heat Mass Transfer 49 (25–26) (2006) 4732–4750.

[28] D.J.C. MacKay, Bayesian interpolation, Neural Comput. 4 (1992) 415–447.

[29] F.D. Foresee, M.T. Hagan, Gauss–Newton approximation to Bayesian learning, in: Proceedings of the 1997 International Joint Conference on Neural Networks , Houston, 1997, pp. 1930–1935.

[30] E. Kreyszig, Advanced Engineering Mathematics, seventh ed., Wiley, New York, 1993, pp. 648–649.

[31] H.S. Carslaw, J.C. Jaeger, Conduction of Heat in Solids, second ed., Oxford University Press, New York, 1959, pp. 203, 328–329.