Tendências em Matemática Aplicada e Computacional, **3**, No. 2 (2002), 189-195.

Estimation of Boundary Conditions in Conduction Heat Transfer by Neural Networks

E.H. SHIGUEMORI, F.P. HARTER, H.F. CAMPOS VELHO, J.D.S. da SILVA[1],
Instituto Nacional de Pesquisas Espaciais, INPE, 12227-010 São José dos Campos, SP, Brasil.

Abstract. Two different artificial neural networks (NN) are used for estimating a time dependent boundary condition ($x = 0$) in a slab: multilayer perceptron (MP) and radial base function (RBF). The input for the NN is the temperature time-series obtained from a probe next to boundary of interest. Our numerical experiments follow the work of Krejsa et al. [4]. The NNs were trainned considering 5 per cent of noise in the experimental data. The training was performed considering 500 similar test-functions and 500 different test-functions. Inversions with trained NNs with different test-functions were better. The RBF-NN presented a slightly better results than MP-NN.

1. Introduction

The solution of a direct (forward) problem consists of finding effects from a complete description of their causes. However, the solution of an inverse problem, unknown causes are determined from observed or desired effects. Different kinds of inverse problems can be found in the literature, based on nature of estimated property [5]: estimation of initial/boundary conditions, properties of the system/material, sources or sink terms, shape, and governing equations. Many texts are available describing (traditional) techniques for solving inverse problems in heat consduction, see [1, 10] for example.

The present work is focused on the first type of inverse problem mentioned above. In recent works [6, 7] initial conditions are estimated by classical inverse techniques, such as Tikhonov regularization, maximum entropy principle, and truncated singular value decomposition. Reference [4] deals (see also [13]) with reconstruction of boundary condition using neural networks: multilayer perpectron (MP), radial base function (RBF), cascade-correlation (CC) and cascade-correlation with genetic algorithm (CCGA). The main conclusion is that NNs are effective tools as alternative techniques for solving inverse problems and they deserve investigation.

The estimation of boundary conditions in inverse heat conduction problem by neural networks is the topic of this paper. However, two new features are studied

[1] Laboratório de Computação e Matemática Aplicada.

here: different training data sets for learning step, considering similar and non-similar data sets; also, several learning strategies for RBF-NN are studied. These purposes distinguish our research from previous ones [4, 13].

Sections 2 and 3 the forward problem and inverse modeling are presented, respectively. Section 4 numerical examples are shown, and in Section 5 final remarks are addressed.

2. Formulation for Forward Problem

The forward problem is the linear one-dimensional unsteady heat conduction process. The mathematical formulation is given by

$$\frac{\partial T(x,t)}{\partial t} = \alpha \frac{\partial^2 T(x,t)}{\partial x^2}, \qquad x \in (0,\ L) \text{ and } t > 0, \tag{2.1}$$

$$k \left.\frac{\partial T}{\partial x}\right|_{x=L} = 0, \qquad -k \left.\frac{\partial T}{\partial x}\right|_{x=0} = q(t), \tag{2.2}$$

$$T(x,0) = f(x), \tag{2.3}$$

where $T(x,t)$ is the temperature, x and t denote space and time variables, and $f(x)$ is the initial condition. The numerical solution is obtained using a finite difference approach, central differences for space, and explicit Euler method for time integration [3, 8, 9].

3. Inverse Problem Solution by Neural Networks

An inverse solution can be understood as an attempt to find out the inverse operator P^{-1} (or an approximation Q for it):

$$P[q(t)] = T(x,t) \quad \Rightarrow \quad q(t) = P^{-1}[T(x,t)] \ . \tag{3.1}$$

A typical approach to compute the unknown $q(t)$ is to formulate the inverse problem as a non-linear optimization problem:

$$\min \ J(q) \ \text{ where : } \ J(q) = \left\| T^{\mathrm{Exp}} - T^{\mathrm{Mod}}(q) \right\|_2^2 + \alpha\Omega[q(t)], \tag{3.2}$$

being T^{Exp} measured quantities, T^{Mod} are computed quantities from a mathematical model, and Ω is a regularization operator [6, 12]. The approach based on the artificial neural networks is to design a non-linear mapping to obtain an approximated inverse solution: $q(t) = Q_{NN}[T(x,t)]$, where $Q_{NN} \sim P^{-1}$.

Artificial NNs have two stages in their application, which are the learning and activation steps. During the learning step, the weights and bias corresponding to each connection are adjusted to some reference examples. For activation, the output is obtained based on the weights and bias computed in the learning phase. There

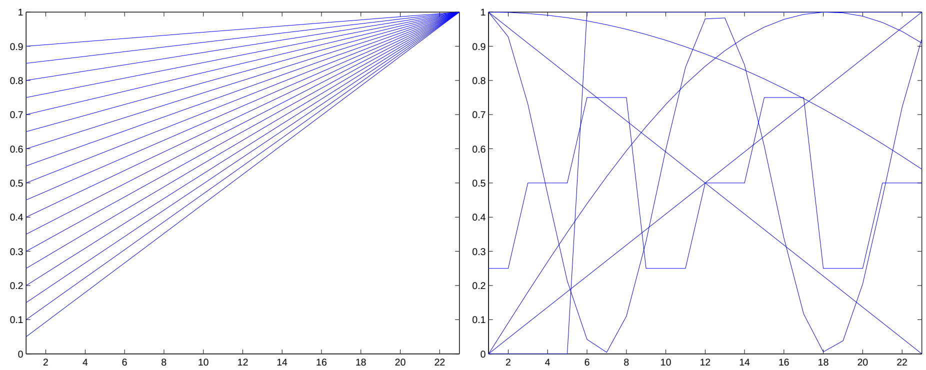

Figure 1: (a) Similar functions, (b) Non-similar functions.

are many possible arrangements and learning strategies. For the present inverse problem, two neural networks are implemented: multilayer perceptron, with back-propagation algorithm, and radial base function. See references [2, 11, 13] for a full description of these NN architectures. A supervised learning was used in both NNs.

The MP-NN has one input layer, one or more hidden layers, and one output layer. It is a feed-forward network and employs a back-propagation algorithm for the learning process. The RBF-NN is composed by only one hidden layer. Differently from the most NNs, there are no weights in the connections between the input layer and the hidden layers, but Gaussian functions.

For both NNs, the training sets are constituted by synthetic data obtained from the forward model, i.e., time-series for a *measure* point close to the boundary. Two different data sets were used. The first data set is the time-series obtained from 500 similar functions (see examples in Figure 1 (a)). The second one is that obtained with 500 non-similar functions (Figure 1 (b)). Similar functions are those belonging to the same class (linear function class, trigonometric function class, such as sine functions with different amplitude and/or phase, and so on). Non-similar functions are those completely different, in which each one belonging to a distinct class.

4. Results and Conclusions

In order to analyze the performance of the MP-NN and RBF-NN in the identification of the boundary condition in heat conduction, two experiments were performed. In the first one noiseless data sets were used. For the second experiment 5% of white Gaussian noise was added to the synthetic data, simulating the real experimental data.

Both NNs were trained with only one hidden layer. The MP-NN and RBF-NN having 20 and 30 neurons, respectively. Table 1 presents the results of training of both NNs, for similar and non-similar noiseless data. The training is carried out until a the minimum value for the error is reached or a maximum number of epoch (epoch denotes the iteration step in the learning phase). The *error* is computed by the following equation

$$\varepsilon(n) = \frac{1}{2N} \sum_{n=1}^{N} \sum_{j=1}^{N_x} \left[T_j^{\mathrm{Exp}}(n) - T_j^{NN}(n) \right]^2 , \tag{4.1}$$

being j the grid point and n an index for training function.

Table 1: Result for training of MP-NN and RBF-NN with similar and non-similar functions – noiseless data.

Neural network	Data set	Num. of epoch	Error
MP	Non-similar	10000	0.0371
MP	Similar	10000	0.0147
RBF	Non-similar	10000	0.0278
RBF	Similar	5348	0.0115

For RBF-NN there are several strategies to find the center or means of the associated Gaussian activation function. Three strategies were tested here: fixed centers randomly selected, self-organized selection of centers, and supervised selection of centers. The best results were obtained with the first strategy. Here after, all results are related with this strategy of training.

During the training phase Table 1 shows that the error obtained to RBF-NN is smaller than MP-NN, for both training sets. For both NNs the error using similar functions was smaller than that obtained with non-similar functions.

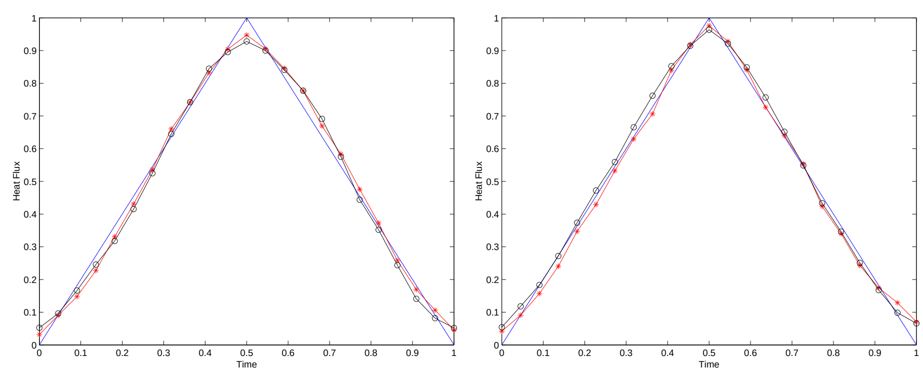

Figure 2: (a) Inversion with MP, (b) Inversion with RBF. Solid line represents the exact output, solid line with circles represents similar functions and solid line with stars represents non-similar functions.

Table 2: Results of training of MP-NN and RBF-NN with similar and non-similar functions – 5% of noise.

Neural network	Data set	Num. of epoch	Error
MP	Non-similar	10000	0.0538
MP	Similar	10000	0.0205
RBF	Non-similar	10000	0.0386
RBF	Similar	10000	0.0139

Reconstructions of boundary conditions with noiseless data are presented below. In the jargon of the neural network field, this phase is denominated *activation*. Figure 2 (a) shows the results related to MP-NN estimation, for similar and non-similar functions. Figure 2 (b) shows the same experiment, but using RBF-NN. For both experiments the activation is done with functions that do not belong to the training data set. The error for MP-NN was 0.00932 and 0.01333 respectively for network trained with similar and non-similar functions. The RBF-NN has 0.00798 and 0.00842 errors for the same sets, respectively.

The previous experiments were re-run adding 5% of noise in the training data sets. Results are shown in Table 2 and Figures 3 (a) - 3 (b).

Again, the RBF-NN error obtained during the training phase is smaller than the error for MP-NN. The error is smaller using similar functions than using non-similar functions for MP-NN and RBF-NN. The same boundary conditions estimated in Figure 2 are done here, but with 5% of noise in the data. The inversion is plotted in Figures 3 (a) - 3(b). The error for reconstruction with RBF-NN is smaller than MP-NN estimation.

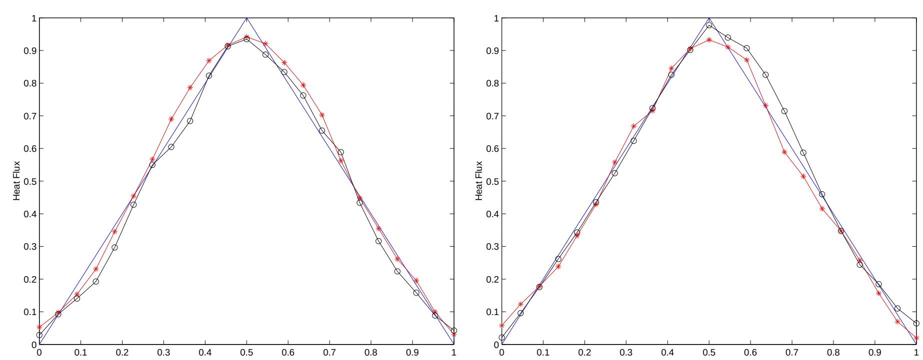

Figure 3: (a) Inversion with MP, (b) Inversion with RBF. Solid line represents the exact output, solid line with circles represents similar functions and solid line with stars represents non-similar functions.

Good estimations for boundary conditions are obtained even with 5% of noise in the data. The MP-NN presented errors 0.01273 and 0.01331 for similar and non-similar functions, respectively. The RBF-NN presented error respectively 0.01013 and 0.01166 for the same set.

The paper presents a solution of the inverse problem in heat conduction by artificial neural network. The boundary condition is retrieved by two neural networks, even with 5% of noise in the observational data. The reconstruction of the boundary conditions are comparable with those obtained with regularization methods [6], even for data containing noise. However, the NN do not remove the inherent ill-posedness of the inverse problem. The performance of two neural networks are studied using two different data set in the training – similar and non-similar functions. In the activation phase, it is observed that better results are obtained when non-similar data were used for training in both the NNs.

As mentioned previously, processing with NNs is a two step process: training and activation. After the training phase, the inversion with NNs is much faster than the regularization methods, and the NNs do not need a mathematical model to simulate the forward model. In addition, NNs is an intrinsicly parallel algorithm. Finally, NNs can be implemented in hardware devices, the neurocomputers, becoming the inversion processing faster than NNs emulated by software.

References

[1] J. V. Beck, B. Blackwell, C. R. St. Clair, "Inverse Heat Conduction: Ill-Posed Problems", John Wiley & Sons, 1985.

[2] S. Haykin, "Neural Networks: A Comprehensive Foundation", Mcmillan, 1994.

[3] J. D. Hoffman, "Numerical Methods for Engineers and Scientists", McGraw-Hill, Inc., New York, USA, 1993.

[4] 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 Problems in Engineering*, **7** (1999), 197-213.

[5] N.J. McCormick, Inverse problems: methods and applications, em "Seleta do XXIII CNMAC" (E.X.L. Andrade et al., eds), Tendências em Matemática Aplicada e Computacional, Vol. 2, pp.1-12, SBMAC, 2001.

[6] W.B. Muniz, F.M. Ramos, H.F. Campos Velho, Entropy- and Tikhonov- based regularization techniques applied to the backwards heat equation, *Computers & Mathematics with Applications*, **40** (2000), 1071-1084.

[7] W.B. Muniz, H.F. Campos Velho, F.M. Ramos, A Comparison of some inverse methods for estimating the initial condition of the heat equation, *Journal of Computational and Applied Mathematics*, **101** (1999), 153-171.

[8] M.N. Ozisik, "Heat Conduction", John Wiley & Sons, 1980.

[9] M.N. Ozisik, "Finite Difference Methods in Heat Transfer", CRC Press, 1994.

[10] M.N. Ozisik, H.R.B. Orlande, "Inverse Heat Transfer: Fundamentals and Applications", Taylor & Francis, 2000.

[11] J.D.S. Silva, E.H. Shiguemori, H.F. Campos Velho, Neural network systems for estimating the initial condition in a heat conduction problem, em "International Joint Conference on Neural Networks — World Conference on Computational Intelligence", 2002, Honolulu, USA (Proceedings in CD Rom).

[12] A.N. Tikhonov, V.Y. Arsenin, "Solutions of Ill-Posed Problems", Winston and Sons, 1977.

[13] K.A. Woodbury, Neural networks and genetic algorithms in the solution of inverse Pproblems, *Bulletim of Brazilian Society for Computing and Applied Mathematics*, available in the site: http://www.sbmac.org.br/a_sbmac/publicacoes/frames.htm .