A COMPUTATIONAL METHOD TO SOLVE FOR THE HEAT CONDUCTION TEMPERATURE FIELD BASED ON DATA-DRIVEN APPROACH

by

Kun LI, Shiquan SHAN, Qi ZHANG, Xichuan CAI, and Zhijun ZHOU *

State Key Laboratory of Clean Energy Utilization, Zhejiang University, Hangzhou, China

Original scientific paper
https://doi.org/10.2298/TSCI200822165L

In this paper, a computational method for solving for the 1-D heat conduction temperature field is proposed based on a data-driven approach. The traditional numerical solution requires algebraic processing of the heat conduction differential equations, and necessitates the use of a complex mathematical derivation process to solve for the temperature field. In this paper, a temperature field solution model called hidden temperature method is proposed. This model uses an artificial neural network to establish the correspondence relationship of the node temperature values during the iterative process, so as to obtain the "Data to Data" solution. In this work, one example of 1-D steady-state and three examples of 1-D transient state are selected, and the calculated values are compared to those obtained by traditional numerical methods. The mean-absolute error of the steady-state is only 0.2508, and among the three transient cases, the maximum mean-square error is only 2.6875, indicating that the model is highly accurate in both steady-state and transient conditions. This shows that the hidden temperature method simulation can be applied to the solution of the heat conduction temperature field. This study provides a basis for the further optimization of the hidden temperature method algorithm.

Key words: *hidden temperature method, artificial neural network, data-driven, numerical solution, heat conduction*

Introduction

In engineering practice, it is often difficult to obtain analytical solutions to thermal conduction differential equations due to the limitations of mathematical methods. With the development of computer technology, the development of the methods for numerical solution of the heat conduction problem has become a long-standing research hotspot. In the past few decades, classical numerical approaches such as the finite difference method (FDM) [1], finite volume method [2], and boundary element method [3] have been successfully applied to heat conduction problems. While the accuracy of these methods has been verified, the solution of the discrete equations for heat conduction problems in practical applications still involves a large number of complex integral calculations, greatly increasing the difficulty of engineering operations. This has motivated an intense research effort focused on the exploration and development of new numerical solution approaches.

Yang *et al.* [4] proposed a new approach for boundary-domain integral equation based on the use of fundamental solution for isotropic problems, and utilized the radial integration

* Corresponding author, e-mail: zhouzj@zju.edu.cn

method to convert regional integration into the equivalent boundary integration. In the calculation process, only the characteristics of the boundary must be considered and the internal domain can be ignored. This method was shown to be suitable for addressing the problem of transient heat conduction. Chang and Liu [5] introduced virtual time variables and virtual damping viscosity coefficients in the integration solution increase the stability of the numerical integration of the discrete equations, and obtained good results for both the forward and inverse heat conduction problems. These studies have reduced the difficulty of integral operations through mathematical methods and have achieved considerable advances.

The development of meshless methods provides another approach for the calculation of the heat conduction field. Gu *et al.* [6] used the singular boundary method for the 3-D heat conduction problem with variable thermal conductivity. Cheng and Liew [7] established the virtual boundary conditions through the compensation method, and used the meshless reproducing kernel particle method for the 3-D transient heat conduction problem, obtaining good results. In addition, many meshless methods such as smooth particle hydrodynamics [8] and radial basis function [9] have been shown to be efficient in numerical calculation applications. Although the meshless method does not involve the division of discrete domains, it is necessary to set the virtual boundary conditions to ensure the accuracy of the results, increasing the difficulty involved in the solution of the equations.

The development of artificial neural networks has also promoted the development of numerical solutions for the heat conduction temperature fields. Nabian *et al.* [10] studied the calculation accuracy of the deep learning framework for the equation that contains random variables. In this process, the thermal conductivity coefficient was defined as a random variable, and the obtained results demonstrated that this approach can be used to directly solve the thermal conduction differential equations. Deng *et al.* [11, 12] and Hwang *et al*. [13] established the heat conduction control equation in the calculations for the non-heat conduction problem and the topology diagram of the equivalent connection circuit by the continuous-time analogue Hopfield neural network, and then obtained the solution of the control equation. Thus, these studies have made good progress in the use of deep learning for numerical calculations of thermal conductivity.

Although the aforementioned methods have achieved significant progress in reducing the mathematical derivation requirements and the computational cost, they are still based on the approach of obtaining the temperature values by directly solving the physical equation. Therefore, the accuracy and efficiency of these methods are limited by the design of mathematical methods. To overcome this limitation, the problem needs to be considered from a new perspective, and therefore, in this work we investigated a data-driven technique for obtaining the temperature field. Sudheer *et al.* [14] pointed out that the core advantage of neural networks is that there is no need to describe the complexity of the underlying process in a mathematical form, and rather the problem is solved by processing a large amount of data containing the physical information, skipping the tedious mathematical solution process. This approach is known as data-driven technique and has been used for certain applications in the biomedical field [15, 16].

In view of the previously mentioned, this paper proposes a black box model that is suitable for solving thermal conduction differential equations based on the data-driven approach:

$$T\left(1,2,\cdots,i\right) \to HTM \to T\left(k\right) \tag{1}$$

A series of nodes are labeled 1, 2,…, i in a specific order and T(1,2,…, i) represent the temperature values of the nodes. The node for which the temperature value is evaluated is called Point k. Here hidden temperature method (HTM) is a hidden temperature calculation method defined in this article, and the details of its algorithm are described. The model becomes

an important interface between the data and physical model through learning. Subsequently, the accuracy of the method was verified using four examples of steady-state and transient state conditions. The main contributions of this work are:
– A computational method for numerical data-driven calculation of heat conduction is proposed.
– Comparison of the results obtained by the data-driven method to the results obtained by standard methods indicates that this method is suitable for modelling heat conduction in steady-state and transient conditions.
– In transient problems, highly accurate results are obtained for both constant temperature boundary conditions and heat flow boundary conditions.

Problem and method

Problem set-up

Cylindrical heat conduction phenomena are common in engineering practice. For example, in a gasification furnace, molten fly ash particles adhere to the furnace wall to form a slag layer. The geometric form of the slag layer can be regarded as a hollow cylinder [17]. Investigation of the temperature distribution of the slag layer can elucidate whether the slag layer exists in a solid form, so as to better monitor and treat the effect of the slag layer on the heat transfer process of the boiler.

In this paper, we consider the hollow cylinder, as shown in fig. 1. Due to the symmetry of the structure, it is generally believed that the heat transfer process of the cylinder along different radial directions is essentially identical, so that the heat conduction problem of the cylinder can be treated as a 1-D problem, and different boundary conditions can be established to deal with the steady-state problem and the transient state problem.

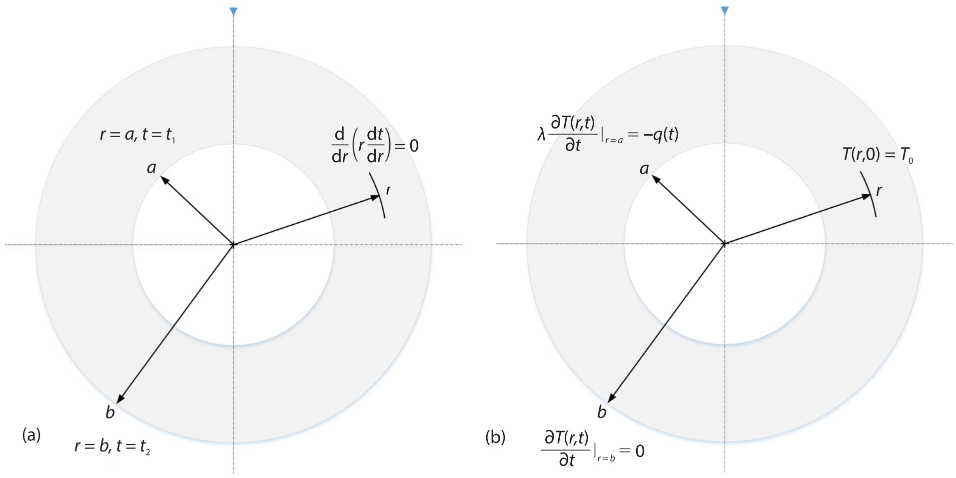

Figure 1. Geometry of the hollow cylinder model; (a) steady-state heat conduction and (b) transient heat conduction

We focus on the correspondence between the node temperatures. To facilitate the exploration of this problem, unified physical and geometric parameters are used in this work [13]. The inner diameter of the hollow cylinder is set to 0.6 m and the outer diameter is 3 m. Other thermal parameters are also set to specific values, namely, the thermal conductivity is 10 W/mK, the constant pressure specific heat capacity is 440 Jk/gK, and the density is 7800 kg/m^3.

Solution methodology

The 1-D heat conduction problems are mainly divided into steady-state problems and transient problems. For a steady-state problem, due to the simplicity of the model, the results of the numerical difference and direct integration methods are equivalent. Generally, the integration method can be used to directly solve for the temperature values. For transient problems, the FDM can generally be used for the calculation of the temperature values. For a very small difference step, the results obtained by the FDM calculations can be considered to be completely equivalent to the real solution [18]. In this work, the integration operation results are selected as the database for steady-state calculations, and the FDM results are selected as the database for transient calculations. These two groups of datasets are used to build two HTM models. Finally, the boundary conditions that were not included in the database are selected to compare the results of the HTM calculations to the results obtained by the traditional methods.

The process used to establish the HTM model is illustrated in fig. 2.

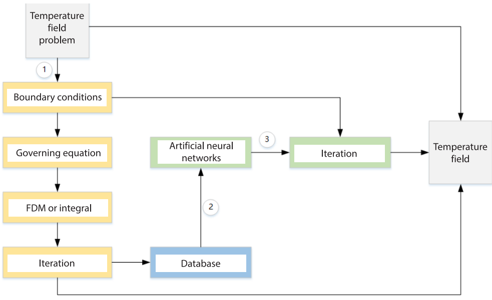

Figure 2. Process used to establish the HTM

In the solution of the heat conduction temperature field, the boundary conditions must be determined first, and the control equation of the temperature field can be expressed according to the boundary conditions. In the 1-D steady-state heat conduction problem, the governing equation can be directly solved by integration. For other cases, the temperature field is discretized, and then the control equation is converted into a difference equation using the FDM, and the temperature values of the nodes are iteratively calculated to obtain the temperature field. This process is the Step 1 in fig. 2 and it is also the traditional numerical solution used to solve for the temperature field. The transformation of the governing equations into difference equations is a classic problem in numerical analysis [19], and will not be described here.

Step 2 is to merge the data values of the multiple sets of the temperature fields to establish a database. After reasonable classification and processing of the data, the data are imported into an artificial neural network model for training, and a black box model with computing power is obtained. Then in Step 3, the black box model is used to perform iterative calculations, and if the obtained temperature field data are consistent with the true values, the model is established successfully. This black box model is defined as HTM, which is consistent with the data-to-data approach. Therefore, establishing the correspondence relationship of the node temperature values is the core of HTM.

For the 1-D steady-state heat conduction problem, the iterative calculation only involves propagation in the spatial direction, as shown in fig. 3. The nodes with a given temperature value are represented in gray, including the temperature value of the boundary node and the initial temperature value of the central node. The red – 1, yellow – 2, and green – 3 nodes represent the nodes for which the temperature must be calculated during the iterative process. The blue – 4 nodes represent the nodes in the final temperature field after the iterative process. It is important to note that fig. 3 is only an

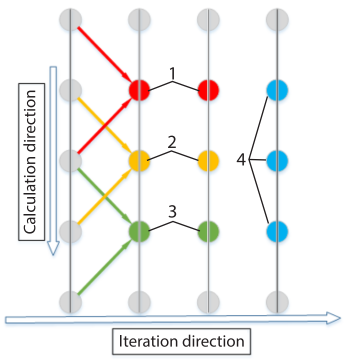

Figure 3. The 1-D steady-state heat conduction node

illustration of the iterative process of the steady-state heat conduction temperature field, and in actual calculations, the number of nodes will be much greater than that shown in the illustration.

In the integration calculation of the 1-D steady-state problem, the temperature value of each node is affected by the temperature values of the two neighbor nodes. Inspired by the aforementioned ideas, we set the temperature of 2 nodes as input, and the temperature of 1 node as output in the calculation of steady-state heat conduction. That is, there is a 2 to 1 relationship between nodes.

Figure 4 describes the updating of the nodes during iteration. In the first iteration, the temperatures of the two gray nodes are used to calculate the temperature of the red – 1 node between them, as shown in fig. 4(a). In the calculation of the temperature of the yellow – 2 node, the temperature value of the node above the yellow node has become a new temperature value, that is, the temperature value of the yellow node must be obtained using the temperature value of the red node and the temperature value of the gray node, as shown in fig. 4(b). Similarly, for the calculation of the temperature of the green – 3 node, the temperature value of the yellow node must be used, as shown in fig. 4(c). Along the heat transfer direction, after all of the nodes have been updated, one iteration of the process ends and the next iteration are performed, as shown in fig. 4(d). When the temperature of the same node changes by less than 10^{-5}, it is generally considered that the iterative process is completed and the final temperature field has been obtained.

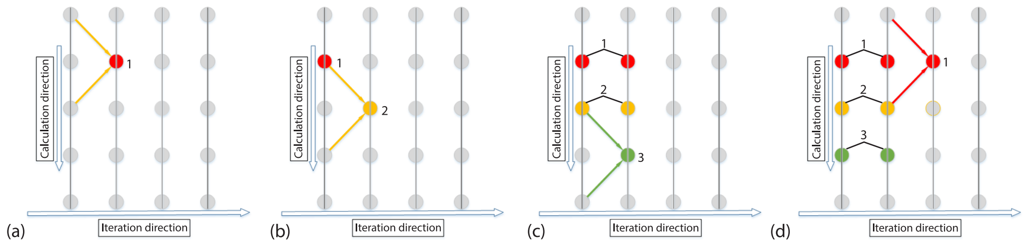

Figure 4. Schematic diagram of the iterative process used to obtain the temperature field

Defining $T_{i,j}$ to represent the temperature value of the i^{th} point at the j^{th} iteration, node correspondence can be expressed:

$$T_{i-1,j} + T_{i+1,j-1} \to T_{i,j} \tag{2}$$

For the 1-D transient heat conduction problem, the iterative calculation process involves both spatial and temporal propagation, as shown in fig. 5. The vertical axis represents the propagation in the spatial direction, and the horizontal axis represents the propagation in the time direction. In the FDM calculations, the temperature value of a node is calculated from the temperature values of the three nodes at the previous time step, namely the node itself and the two neighbor nodes.

Here, node correspondence can be expressed:

$$T_{i-1,j} + T_{i,j} + T_{i+1,j} \to T_{i,j+1} \tag{3}$$

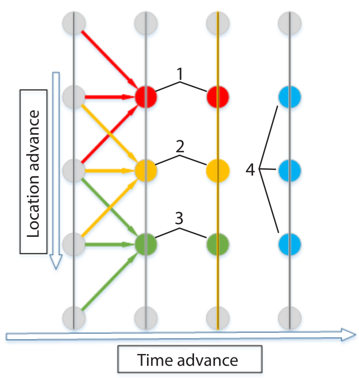

Figure 5. The 1-D transient thermal conduction node correspondence

The actual correspondence is not a direct addition of the temperature of the three nodes, and eq. (3) only symbolically represents the 3 to 1 node relationship. The iterative process is the same as that used for the steady-state case, and is not described again.

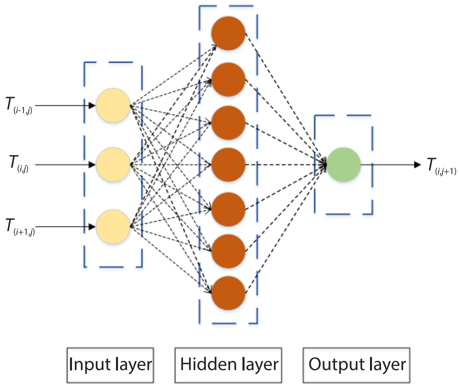

Figure 6. Schematic diagram of the neural network structure

It is assumed that the *data-data* calculation effect can be achieved through the corresponding relationship of the node temperature values. While we refer to the FDM to construct the node relationship, in fact, the correspondence relationship of the nodes does not need to be mathematically derived. We take the data sets in the steady-state and transient problems, respectively, and pre-process the data to make them correspond to the correspondence relationship. The temperature data of the surrounding points before the iteration are taken as the input, and the temperature values obtained by the iterative calculation are taken as the output. We select the back propagation neural network as the test network to construct the neural network. Figure 6 shows a schematic diagram of the neural network of the 1-D transient model. The 1-D steady-state neural network only modifies the input from 3 node values to 2 node values.

The neuron model includes the input layer, the output layer and the hidden layer [20], where the hidden layer can be a single layer or multiple layers. For a single neuron, we assume the input as $x_1, x_2,\ldots, x_i,\ldots, x_n$, the corresponding neuron connection weights as $\omega_1, \omega_2,\ldots, \omega_i,\ldots, \omega_n$, the threshold for neurons as θ, and then the output value can be expressed:

$$y = f\left(\sum_{i=1}^{n} w_i x_i - \theta\right) \tag{4}$$

As mentioned previously, the HTM model can be obtained by taking the temperature values to be iterated as the node output and the initial temperature values before the iteration as the node input and substituting the neural network for training. The advantage of the HTM model is that it avoids complex integration operations, greatly reduces the difficulty of derivation in numerical calculations, and more efficiently exploits the advantages of the computational approach.

Cases and results

In this paper, two sets of neural networks are built to solve the four typical cases of steady-state and transient problems.

Steady-state problem

Case 1 is used to explore the 1-D steady-state heat conduction problem, as shown in the fig. 1(a), and the most commonly used fixed wall temperature condition is selected as the boundary condition.

First, the cylindrical surface is meshed by setting up a grid node at the intervals of 0.1 m for a total of 25 nodes in 1-D. For the steady-state heat conduction problem of a 1-D cylinder, the accurate distribution of the temperature field can be obtained through integral calculation. Given 100 sets of fixed wall temperature boundary conditions, the corresponding temperature fields are obtained through integral solution that is then used as the initial database for neural network learning.

As mentioned previously, the nodes have a 2-to-1 correspondence during 1-D steady-state heat conduction. We process the data in the database according to the correspondence relationship to obtain the training set. Both the size of the data set and the number of the nodes in the hidden layer of the neural network will affect the final calculation result, and currently no mature theoretical guidance for the selection of these two parameters is available. This paper determines the final parameter values by changing these two parameters to compare the test set error during the training process. The error of the neural network is generally expressed by mean-square error (MSE). The R value measures the correlation between the expected data and the actual data. A correlation of 1 indicates that the two are completely consistent, and a correlation of 0 indicates that the data is completely random [21]:

$$MSE = \frac{1}{m}\sum_{i=1}^{m}\left(y_i - \hat{y}_i\right)^2 \tag{5}$$

The obtained error values are shown in tab. 1.

Table 1. The MSE of different neural networks in steady-state heat conduction

Number of nodes / Amount of data	10	20	30
1000	0.132	0.102	0.0852
1500	0.0904	0.0725	0.0474
2000	0.0919	0.0648	0.0457
2300	0.0794	0.0599	0.0684

Although the error value of each training is not the same for the same data set and the same number of nodes, it is generally within a certain range. However, in this example, the changes in the amount of data and the number of nodes have no significant effect on the error. The neural network with 20 hidden layer nodes and 2300 sets got the best training result of tab. 1.

The MSE and R values of the neural network are shown in tab. 2. The error values of the training set and the test set are on the order of 10^{-2}. A schematic diagram of the R value is shown in fig. 7. It is observed that the accuracy of the neural network meets the calculation requirements.

Table 2. The 1-D steady-state model neural network training error

	Samples	MSE	R
Training	1610	0.0760	9.999*10-1
Validation	345	0.0570	9.999*10-1
Testing	345	0.0599	9.999*10-1

Case 1. Constant temperature on both sides: In this case, the temperature of the inner wall surface is 1185 K and the temperature of the outer wall surface is 100 K (this set of operating conditions does not appear in the learning data set). The neural network is used for the calculation of the temperature field and the results are compared with the results of integral calculations as shown in fig. 8.

Here, the red –1 dots represent the HTM calculation results, and the black – 2 lines describe the integral calculation results. It is clear that the two sets of results are in excellent agreement.

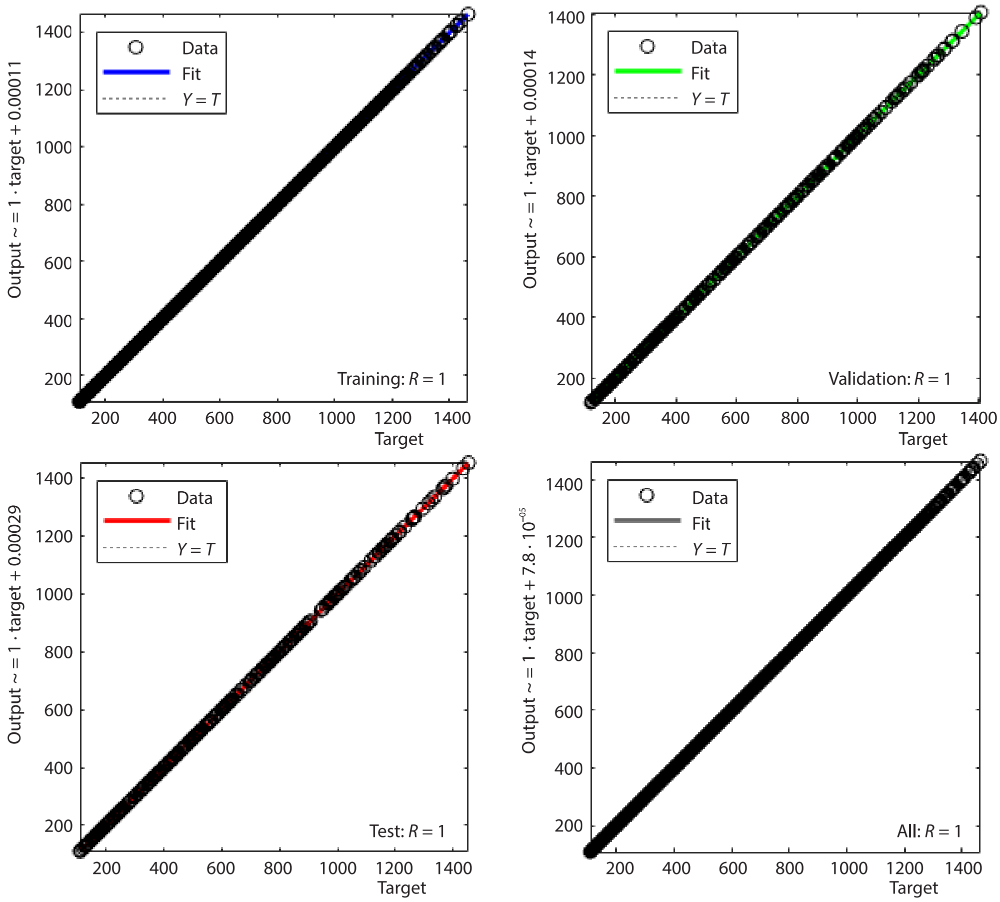

Figure 7. The 1-D steady-state model neural network

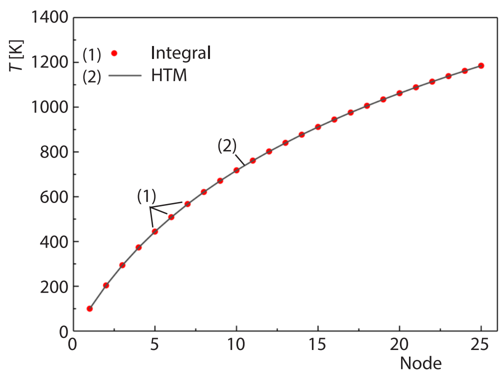

Figure 8. The 1-D steady-state model calculation results

We measure the calculation error by the mean-absolute error (MAE):

$$MAE = \frac{1}{n}\sum\left(\left|t_{\mathrm{HTM}} - t_{\mathrm{int}}\right|\right) \tag{6}$$

For this case we obtain $MAE = 0.2508$. The calculation error indicates that the calculation results are highly accurate.

Transient thermal problems

In Cases 2-4 the problem of 1-D transient heat conduction is explored by changing the heat flow, as shown in the fig. 1(b). In engineering practice, regardless of whether a combustion reaction or a flow of hot fluid occurs inside the cylinder, heat is exchanged on the inner wall surface. In Cases 2-4 the inner wall surface is selected as the heat flow wall surface, and the outer wall surface is the insulated wall surface. Then, the boundary conditions can be expressed:

$$\lambda \frac{\partial T(r,t)}{\partial t}\Big|_{r=a} = -q(t) \tag{7}$$

$$\lambda \frac{\partial T(r,t)}{\partial t}\Big|_{r=b} = 0 \tag{8}$$

Prior to building the neural network, 20 different heat flow conditions were selected, and the temperature field under transient conditions was solved using FDM for these 20 conditions. The obtained data were processed into a 3 to 1 node temperature correspondence relationship, and were imported into the neural network for training. The determination of the number of hidden layer nodes is consistent with the previous method. The error values are shown in tab. 3.

Table 3. The MSE of different neural networks in transient heat conduction

Amount of data \ Number of nodes	10	15	20	25	30
2000	$4.467\cdot10^{-5}$	$3.275\cdot10^{-5}$	$1.114\cdot10^{-5}$	$4.803\cdot10^{-5}$	$3.691\cdot10^{-5}$
4000	$1.571\cdot10^{-5}$	$1.220\cdot10^{-5}$	$5.982\cdot10^{-6}$	$4.074\cdot10^{-6}$	$3.596\cdot10^{-6}$
6000	$2.417\cdot10^{-5}$	$1.725\cdot10^{-5}$	$4.322\cdot10^{-6}$	$3.061\cdot10^{-6}$	$2.695\cdot10^{-6}$
8000	$1.661\cdot10^{-5}$	$9.897\cdot10^{-5}$	$8.651\cdot10^{-6}$	$4.415\cdot10^{-6}$	$2.967\cdot10^{-6}$
10.000	$1.861\cdot10^{-5}$	$1.642\cdot10^{-5}$	$4.208\cdot10^{-6}$	$2.488\cdot10^{-6}$	$1.266\cdot10^{-7}$

An examination of the data presented in tab. 3 shows that the error gradually decreases with increasing number of layers and nodes. Considering the computational cost and the calculation requirements comprehensively, for the three cases we select 10000 groups of nodes as the initial database, and the number of hidden layer nodes of the neural network is set to 30. The training error values of the neural network are shown in tab. 4 and it is observed that the MSE values are on the order of 10^{-7}. The schematic diagram of the R values is shown in fig. 9, demonstrating that the training is effective and meets the calculation requirements of this example.

Table 4. The 1-D transient model neural network training error

	Samples	MSE	R
Training	7000	$1.460\cdot10^{-6}$	$9.999\cdot10^{-1}$
Validation	1500	$1.543\cdot10^{-6}$	$9.999\cdot10^{-1}$
Testing	1500	$1.486\cdot10^{-6}$	$9.999\cdot10^{-1}$

In the temperature field calculations, many kinds of heat flow conditions such as the change of the heat flow with a triangular cross-section may be encountered and the heat flow changes in a stepwise manner. In this work, three cases are selected, including a variety of heat flow conditions (the heat flow conditions included in the example are not in the training set), and the results obtained by the neural network and FDM are compared. In the three sets of calculation examples, the iteration time step is 0.01 s and 600 steps are used in the iterative process.

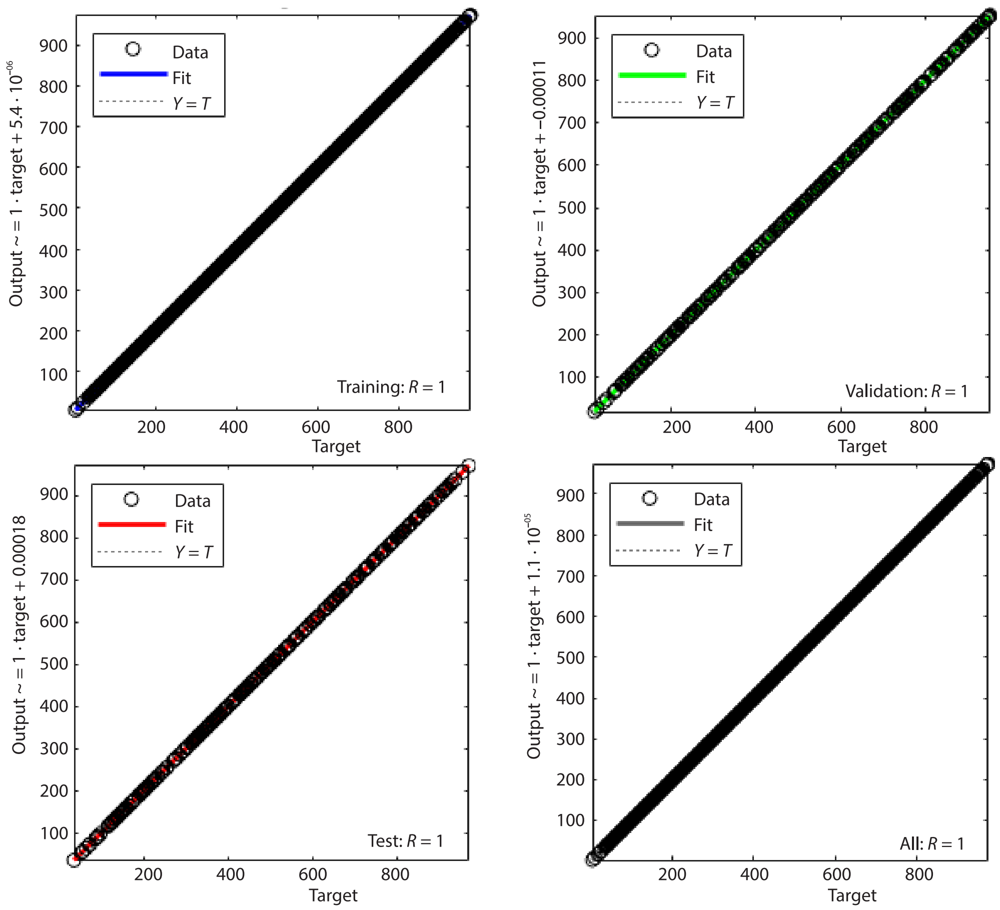

Figure 9. The 1-D transient problem training *R* value

Case 2. The heat flow conditions are given:

$$q(t)=\begin{cases} 0, 0<t\le 1 \\ 1, 1<t\le 3 \\ \dfrac{2}{15}t-0.2, 3<t\le 6 \end{cases} \tag{9}$$

The heat flow function is shown in fig. 10(a). The heat flow contains a step and a triangular profile change. The calculation results obtained by the neural network and FDM methods are compared in fig. 10(b).

Case 3. The heat flow conditions are given:

$$q(t) = \begin{cases} 0.3, 0.0 < t \le 1.0 \\ 0.7t - 0.4, 1.0 < t \le 2.0 \\ -0.5t + 2, 2.0 < t \le 3.0 \\ 0.5, 3.0 < t \le 3.3 \\ 0.5\left[1 + \sin\left(\dfrac{2\pi}{3}t\right)\right], 3.3 < t \le 5.8 \\ 0.3, 5.8 < t \le 6.0 \end{cases} \tag{10}$$

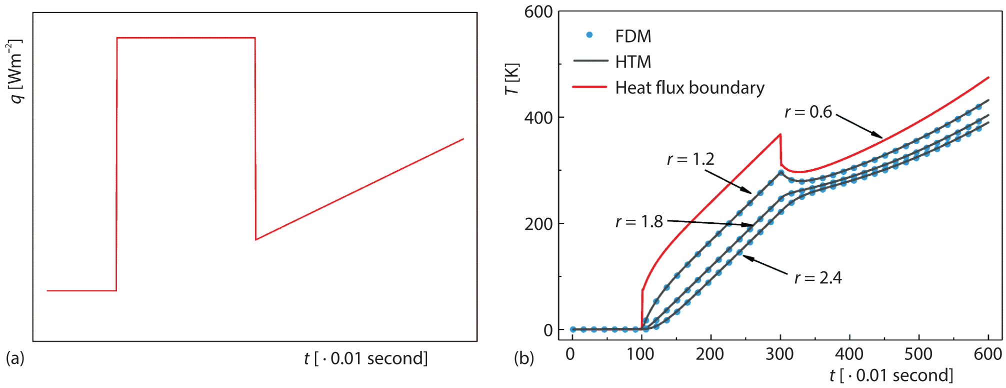

Figure 10. Heat flow and calculation results of *Case 2*

The heat flow function is shown in fig. 11(a). The change in the heat flow first passes through a triangular cross-section and then a sinusoidal curve. The results obtained by the neural network and FDM are compared in fig. 11(b).

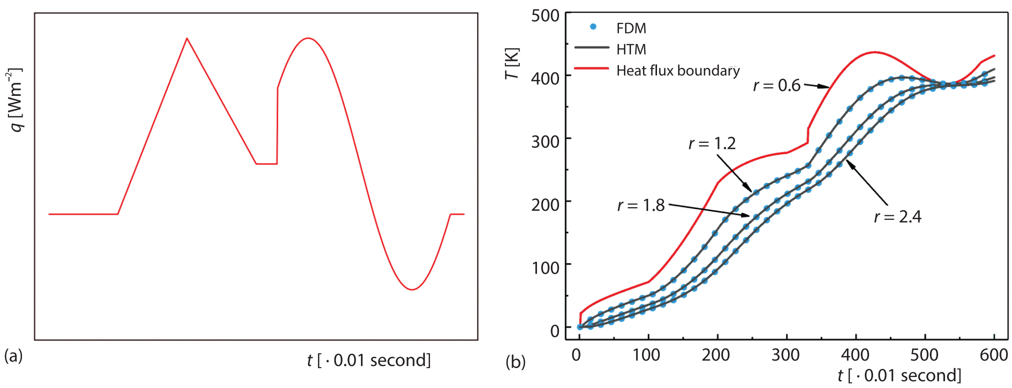

Figure 11. Heat flow and calculation results of *Case 3*

Case 4. The heat flow conditions are given:

$$q(t)=\begin{cases}\sin\left(\dfrac{4\pi t}{5}\right), 0<t\leq 3\\[2ex] 0.5\sin\left(\dfrac{4\pi t}{5}\right)+0.5\sin(8\pi), 3<t\leq 6\end{cases} \tag{11}$$

The heat flow function is illustrated in fig. 12(a). The heat flow curve contains two sets of sinusoids with different frequencies. The results obtained by the neural network and FDM are compared in fig. 12(b).

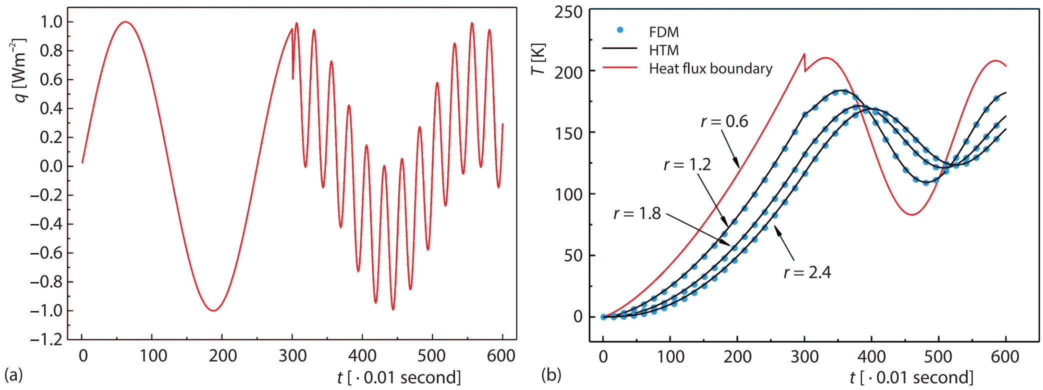

Figure 12. Heat flow and calculation results of *Case 4*

In the comparison of the three cases, the horizontal axis represents the calculation step, and the vertical axis represents the temperature value. The changes of the temperature values for the nodes located at r = 1.2 m, r = 1.8 m, and r = 2.4 m are plotted for 600 steps. In figs. 10(b), 11(b), and 12(b), the FDM calculation results are shown in blue, and the HTM calculation results are represented by black lines. An examination of these figures shows that the two sets of results are in excellent agreement and no obvious error is observed at the position where the heat flow was abrupt.

The calculated error values for the three cases are shown in tab. 5. An examination of these values shows that the maximum error value of the three sets of results is only 2.6875, demonstrating that the HTM calculation results are highly consistent with the FDM results. Thus, it can be concluded that HTM can solve the 1-D transient heat conduction problem well.

Table 5. The 1-D transient thermal calculation error

	Case 2	Case 3	Case 4
MSE	2.6875	0.5184	1.3875

Conclusions

To summarize, this paper proposes a computational method to solve for the heat conduction temperature field. The HTM is constructed by the relationship between the node temperature values, and then is used to calculate the temperature field. Based on the cylindrical heat conduction model, four examples of the calculation of the 1-D heat conduction problem prove the accuracy of the method.

In the 1-D steady-state problem, a 2 to 1 node correspondence is established, and typical constant temperature boundary conditions are selected to reconstruct the temperature field. Compared to the results of integral calculation, the MAE of the steady-state is only 0.2508, confirming the high accuracy of the HTM calculation results.

For the 1-D transient problem, we established a 3 to 1 node correspondence and selected 3 sets of heat flow changes. The maximum MSE error of the three cases is only 2.6875, demonstrating that the HTM model is also accurate for the calculation of the temperature field in the transient problem involving time terms.

It should be noted that in the steady-state and transient problems, the correspondence between nodes is not the same, so that two HTM models must be established. However, in the same node correspondence, HTM shows a strong universality, and shows good performance characteristics under different boundary conditions of the 1-D transient problem.

The results obtained in this work show that HTM can be used for the iterative calculation of 1-D heat conduction, and acts as a highly accurate black box model in the calculation of heat transfer problems.

As mentioned previously, the learning process of the neural network does not require rigorous mathematical derivation of the difference equation. Rather, it only requires the re-integration of the data with the existing temperature field to realize the *data-data* approach. The advantages of HTM will gradually emerge with increasing number of dimensions or increasingly complexity of the calculation model. In future work, we will continue to explore the application of HTM models in complex temperature fields.

Acknowledgment

This work was supported by the Ningxia Science and Technology Department (No. 2018BCE01004) and the National Nature Science Foundation of China (No. 51976186).

References

[1] Mosayebidorcheh, S., *et al.*, Transient Thermal Analysis of Longitudinal Fins with Internal Heat Generation Considering Temperature-Dependent Properties and Different Fin Profiles, *Energy Conversion & Management*, *86* (2014), Oct., pp. 365-370

[2] Mohammed, H. A., *et al.*, Numerical Simulation of Heat Transfer Enhancement in Wavy Micro-Channel Heat Sink, *International Communications in Heat & Mass Transfer, 38* (2011), 1, pp. 63-68

[3] Yang, K., *et al.*, Using Analytical Expressions in Radial Integration BEM for Variable Coefficient Heat Conduction Problems, *Engineering Analysis with Boundary Elements 35* (2011), 10, pp. 1085-1089

[4] Yang, K., *et al.*, Radial Integration BEM for Transient Heat Conduction Problems, *Engineering Analysis with Boundary Elements, 34* (2010), 6, pp. 557-563

[5] Chang, C. W., Liu, C. S., A New Algorithm for Direct and Backward Problems of Heat Conduction Equation, *International Journal of Heat and Mass Transfer, 53* (2010), 23-24, pp. 5552-5569

[6] Gu, Y., *et al.*, Singular Boundary Method for Steady-State Heat Conduction in 3-D General Anisotropic Media, *International Journal of Heat and Mass Transfer, 55* (2012), 17-18, pp. 4837-4848

[7] Cheng, R. J., Liew, K. M., A Meshless Analysis of 3-D Transient Heat Conduction Problems, *Engineering Analysis with Boundary Elements, 36* (2012), 2, pp. 203-210

[8] Rook, R., *et al.*, Modelling Transient Heat Transfer Using SPH and Implicit Time Integration, *Numerical Heat Transfer, Part B: Fundamentals, 51* (2007), 1, pp. 1-23

[9] Waters, J., *et al.*, Global *vs.* Localized RBF Meshless Methods for Solving Incompressible Fluid-Flow with Heat Transfer, *Numerical Heat Transfer, Part B: Fundamentals, 68* (2015), 3, pp. 185-203

[10] Nabian, M. A., *et al.*, A Deep Neural Network Surrogate for High-Dimensional Random Partial Differential Equations, *Probabilistic Engineering Mechanics, 57* (2019), Aug., pp. 14-25

[11] Deng, S., *et al.*, Solving the Temperature Distribution Field in Non-Linear Heat Conduction Problems Using the Hopfield Neural Network, *Numerical Heat Transfer, Part B: Fundamentals, 51* (2007), 4, pp. 375-389

[12] Deng, S., *et al.*, Applying neural networks to the solution of forward and inverse heat conduction problems, *International Journal of Heat and Mass Transfer*, *49* (2006), 25-26, pp. 4732-4750
[13] Hwang, Y., *et al.*, Applying Neural Networks to the Solution of the Inverse Heat Conduction Problem in a Gun Barrel, *Journal of pressure vessel technology, 130* (2008), 3, 031203
[14] Sudheer, K., *et al.*, A Data Driven Algorithm for Constructing Artificial Neural Network Rainfall Runoff Models, *Hydrological processes, 16* (2002), 6, pp. 1325-1330
[15] Bar-Sinai, Y., *et al.,* Learning Data-Driven Discretizations for Partial Differential Equations, *Proceedings of the National Academy of Sciences*, *116* (2019), 31, pp. 15344-15349
[16] Kirchdoerfer, T., Ortiz, M., Data Driven Computing in Dynamics, *International Journal for Numerical Methods in Engineering*, *113* (2018), 11, pp. 1697-1710
[17] Ye, I., *et al.*, Numerical Modelling of Slag Flow and Heat Transfer on the Wall of an Entrained Coal Gasifier, *Fuel, 150* (2015), Feb, pp. 64-74
[18] Wong, H., *et al.*, Improved Finite-Difference Methods Based on a Critical Evaluation of the Approximation Errors, *Numerical Heat Transfer, Part A: Applications, 2* (1979), 2, pp. 139-163
[19] Chung, K., *et al.*, A Generalized Finite-Difference Method for Heat Transfer Problems of Irregular Geometries, *Numerical Heat Transfer, 4* (1981), 3, pp. 345-357
[20] Rafiq, M., *et al.*, Neural Network Design for Engineering Applications, *Computers & Structures, 79* (2001), 17, pp.1541-1552
[21] Zhao, Y., *et al*., An Improved Retrieval Method of Atmospheric Parameter Profiles Based on the BP Neural Network, *Atmospheric Research*, *213* (2018), June, pp. 389-397

Paper submitted: August 22, 2020
Paper revised: February 27, 2021
Paper accepted: March 8, 2021