A transfer learning modeling framework fusing domain knowledge and its application in district heating network

line 1: 1st Liang Zhang
School of Control Science and Engineering
Shandong University
Jinan, China
202234950@mail.sdu.edu.cn

line 1: 2nd Hui Zhang
School of Control Science and Engineering
Shandong University
Jinan, China
202234949@mail.sdu.edu.cn

line 1: 3rd Sun Bo
School of Control Science and Engineering
Shandong University
Jinan, China
jingchen0608@163.com

4th Jing Chen
School of Intelligent Engineering
Shandong Management University
Jinan, China
jingchen0608@163.com

***Abstract*—District heating networks (DHNs) represent a critical infrastructure for decarbonizing the heating sector. However, as DHN systems grow in scale and complexity, the computational burden associated with numerical simulations increases significantly. To meet both user demand and energy efficiency objectives under emission reduction policies, efficient and accurate simulation modeling approaches are essential for the optimal operation and scheduling of DHNs. This paper proposes a transfer learning–based modeling framework integrated with domain knowledge to enable high-fidelity network modeling. First, the partial differential equation governing the dynamic temperature evolution of fluid in heating pipelines is incorporated into a Physics-Informed Neural Network (PINN) as embedded domain knowledge. Subsequently, high-resolution data generated by Computational Fluid Dynamics (CFD) simulations are employed to construct a pre-trained model for an individual heating pipe. Finally, this pre-trained model is fine-tuned according to the specific initial and boundary conditions of various pipelines within the district heating network, enabling accurate knowledge transfer through integration with sparse measurement data. Experimental results demonstrate that the proposed modeling framework achieves strong agreement with reference numerical solutions reported in the literature and exhibits superior accuracy compared to conventional neural network–based modeling approaches.**

Keywords—District heating networks, physics-dominated neural network,

I. INTRODUCTION

In recent years, the establishment of clean, low-carbon, and efficient energy systems has emerged as a global development trend across countries and regions [1]. District heating networks (DHNs) are recognized as a key solution for decarbonizing the heating sector [2], capable of meeting end-user demands for space heating and domestic hot water through the large-scale integration of renewable heat sources, electric heating, waste heat recovery, and conventional fossil fuel-based heat sources within designated areas [3]. This integration offers well-documented advantages in terms of energy efficiency, greenhouse gas emission reduction, and financial sustainability [4]. Consequently, DHNs have witnessed growing research interest and expanding engineering applications.

As DHNs become increasingly complex, challenges such as temperature fluctuation propagation and significant time delays associated with distributed heat sources can severely impact system regulation, leading to reduced operational efficiency. Accurate prediction of transient temperature variations in heat flow is therefore critical for effective system design and operational optimization [2], necessitating the development of more precise thermal transient models [5].

The spatiotemporal evolution of heat flow temperature in DHNs can be mathematically described by the heat convection–diffusion equation [6]. Direct numerical solutions of partial differential equations (PDEs) represent one of the most efficient and widely adopted approaches for thermal transient modeling. Stevanovic et al. [7] employed third-order Lagrangian interpolation polynomials to solve the thermodynamic model, with simulation results showing strong agreement with measured data. Commercial software packages also rely on numerical methods; for instance, Danielewicz et al. [8] utilized ANSYS and FLUENT based on CFD-constructed databases for numerical modeling of district heating networks, while Lim et al. [9] developed a Simulink-based numerical model in MATLAB to simulate the thermal dynamics of DHNs. Although these methods provide high-fidelity thermal transient information, they are computationally intensive and often impractical for real-time operation of large-scale DHNs.

With advancing research, modeling efforts have evolved to account for the coupling between electrical and thermal networks, giving rise to unified thermo-electric equivalent models. Chen et al. [10] proposed a unified modeling approach for electro-thermal transport based on the consistency of energy flow transmission laws. Schweiger et al. [11] established a dynamic simulation and optimization framework using Modelica for thermo-hydraulic systems, while Yang et al. [12] constructed a joint electro-thermal modeling framework through equation simplification via equivalent representation. While such unified multi-energy flow models offer theoretically comprehensive physical descriptions, their practical application is often limited by mathematical complexity, which can hinder accurate prediction of transient behaviors in DHN systems.

To address the computational burden associated with large-scale thermal transient models, researchers have explored data-driven approaches leveraging machine learning

techniques. Liu et al. [13] introduced a data-driven linearization method using partial least squares (PLS) and Bayesian linear regression (BLR) to mitigate issues of collinearity and overfitting. Tian et al. [14] enhanced PLS regression accuracy through a dual-driving strategy, while Guo et al. [15] improved the backpropagation (BP) neural network-based iterative method to increase computational efficiency in hydrothermal dynamic modeling. These models demonstrate favorable performance in both accuracy and efficiency; however, their effectiveness remains highly dependent on the quantity and quality of available training data.

Research on neural networks for solving PDEs dates back to the 1990s [16]. Recently, Raissi et al. [17] introduced the Physics-Informed Neural Network (PINN) framework, which transforms PDE solving into an optimization problem by minimizing a loss function that incorporates physical constraints. Unlike traditional neural networks requiring extensive labeled datasets, PINN effectively embed prior physical knowledge into the learning process, thereby reducing reliance on large volumes of labeled data and combining the strengths of physics-driven and data-driven methodologies. PINN have demonstrated superior computational accuracy in fluid dynamics applications. For example, Cai et al. [18] applied PINNs to predict temperature and velocity fields in two-phase flows, while Zobeiry and Humfeld [19] used them to solve conduction and convection heat transfer PDEs. Additionally, Liu et al. [20] developed a PINN-based temperature field inversion method accelerated by transfer learning. The aforementioned PINN approaches significantly enhance model accuracy while substantially reducing computational costs. Nevertheless, to date, PINN have not been applied to thermal transient modeling in DHNs.

This study proposes a domain knowledge-integrated transfer learning modeling framework aimed at reducing computational complexity without compromising modeling accuracy. First, this work introduces PINN technology into DHN modeling for the first time, offering a more comprehensive alternative to conventional numerical and data-driven methods. Second, transfer learning is employed to enable accurate modeling under conditions of limited data availability. The proposed framework comprises three stages: (1) generating high-fidelity and low-fidelity dataset pairs using CFD-based network simulations; (2) constructing a high-fidelity surrogate model of heating pipelines using the PINN method; and (3) fine-tuning the pre-trained model via transfer learning according to varying initial and boundary conditions across different pipeline segments, enabling model development under low-fidelity data conditions. In summary, this study makes the following contributions:

1. Application of PINN technology to achieve thermal transient modeling of district heating networks, effectively integrating data-driven and physics-driven paradigms and further enhancing modeling accuracy.

2. Development of a domain knowledge-enhanced transfer learning modeling framework, leveraging transfer learning to improve modeling precision under data-limited scenarios.

II. Problem Description

DHN modeling mainly studies the mathematical law of thermal power variation during the transportation of heat flow in the pipeline network. Heat transfer in the heat network is caused by the movement of the working medium carrier and is accomplished through heat exchange between the pipeline network and the environment. Considering a horizontal single pipe, heat transfer can be represented by the variation of water temperature at different positions x with time t, as shown in Fig.1.

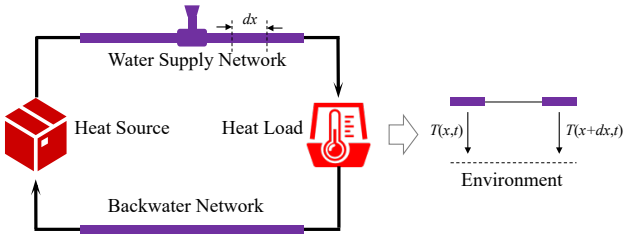


Fig. 1. Thermal flow model of the heating pipeline

The heat transfer in a district heating network (DHN) pipe can be modeled as an incompressible fluid flow, with its energy balance described by the following temperature convection-diffusion equation:

$$c\rho S\frac{\partial T}{\partial t}+cm\frac{\partial T}{\partial x}-\gamma_0\frac{\partial^2 T}{\partial x^2}+\lambda T=0 \qquad (1)$$

Where, $T(x,t)$ is the temperature difference between the water flow in the pipeline and the external environment (generally the soil outside the insulation layer of the pipeline), depending on the time t and the position x along the pipeline; m is the mass discharge of the water stream; c is the specific heat capacity of water; ρ is the density of water; λ is the heat conduction coefficient of the pipe; γ_0 is the radial thermal diffusivity of water flow; S is the cross-sectional area of the heating pipe; T is the hidden solution of the equation.

The first two terms on the left-hand side of equation (1) account for convective heat transfer, the third term represents conductive heat transfer within the fluid, and the fourth term corresponds to thermal losses due to heat dissipation in the pipeline. Since the thermal conductivity of water is extremely low, about 0.59W/mK, its internal static heat conduction is much smaller than the other terms, so the third term can be regarded as 0. The equation can be further simplified as [21] :

$$c\rho S\frac{\partial T}{\partial t}+cm\frac{\partial T}{\partial x}+\lambda T=0 \qquad (2)$$

Equation (2) is the thermal transient model of the DHN pipe. In solving the model, it is necessary to apply the given boundary conditions at the inlet and outlet of the pipe and the initial temperature distribution inside the pipe to the equations.

This study focuses on a typical 32-node low-temperature district heating closed-loop network, as shown in Figure 2, which is heated by three cogeneration units, and nodes 1, 31, and 32 correspond to the three heat sources of the network. This system has been widely studied [14]. It is assumed that the thermal power of the load is known, the initial condition is a steady state, and the water supply temperature of each heat source is 60 ° C. For this network problem, the steady-state mass flow is known and constant for all pipelines, and there is flow mixing only at nodes 5 and 22 in the mainstream lines of the supply network. The same smooth approximate boundary condition is applied at three entrances of the network, namely nodes 1, 31 and 32. This boundary condition is a step function used to describe the process of heat source equipment adjusting the water supply temperature according to the user demand. In addition, two general boundary conditions should be set, that is, the temperature of the flow at infinity distance

from the source approaches the ambient temperature and the temperature of each pipe at the confluence node is the same. The next section gives a brief introduction to physical information neural networks.

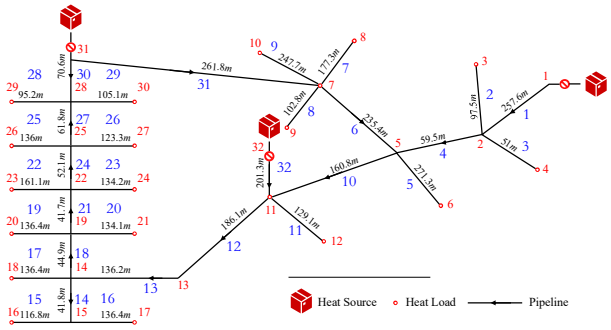


Fig. 2. Schematic diagram of the Bali Island district heating network

III. The Heating Pipe Network Modeling Framework Integrating Domain Knowledge

In this section, the PINN algorithm and theory used to implement the heating pipeline modeling are first introduced. Then, we will use the PINN framework based on transfer learning to solve the problem of modeling the district heating network.

A. Pinn-based Heating Pipeline Modeling

Unlike traditional machine learning approaches that are purely data-driven, the Physics-Informed Neural Network (PINN) is primarily governed by physical laws and is particularly suited for solving problems involving partial differential equations. The framework consists of two key components: a neural network module and a physical information module. Fig.3 illustrates the proposed architecture of the physics-informed neural network.

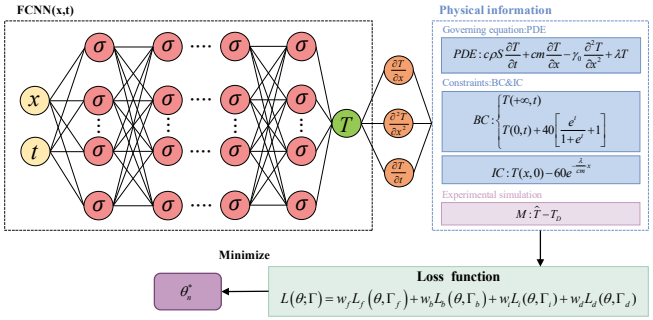


Fig.3. Pipeline surrogate model based on a physics-informed neural network

PINNs commonly employ fully connected neural networks (FCNNs) as surrogate models. Here, an L -layer FCNN, denoted as $FCNN^L(x,t):\mathbb{R}^{d_{in}}\to\mathbb{R}^{d_{out}}$, is first constructed. This L -layer network consists of an input layer, an output layer, and $L-2$ hidden layers, where the l -th layer contains N_l neurons ($N_0=d_{in}, N_L=d_{out}$). Given a nonlinear activation function σ applied layer-wise, the FCNN is recursively defined as follows:

$$FCNN^1(x,t)=(x,t)\in\mathbb{R}^{d_{in}} \tag{3}$$

$$FCNN^l(x,t)=\sigma\left(W^l FCNN^{l-1}(x,t)+b^l\right)\in\mathbb{R}^{N_l}\quad 1\le l\le L-2 \tag{4}$$

$$FCNN^L(x,t)=W^L FCNN^{L-1}(x,t)+b^L\in\mathbb{R}^{d_{out}} \tag{5}$$

Where FCNN$(\cdot)$ represents the artificial neural network, (x,t) defined on $\Omega\in\mathbb{R}^d$ is the input variable of the FCNN, and $W^l\in\mathbb{R}^{N_l\times N_{l-1}}$ and $b^l\in\mathbb{R}^{N_l}$ denote the weight matrix and bias matrix of the l -th layer, respectively. An advantage of PINN is that we can compute derivatives with respect to the inputs x and t by applying the chain rule through automatic differentiation techniques. This functionality is conveniently integrated within machine learning frameworks such as TensorFlow and PyTorch.

For a single heating pipeline, assuming steady hydraulic conditions, the dynamic thermal process of heat flow transported in the pipeline is described by the temperature advection diffusion equation, the boundary condition equation and the initial condition equation (IC). In order to integrate the physical information into the FCNN framework flexitively, the various physical laws in the process are integrated into the following residual function:

$$PDE: F\left(\hat{T},\frac{\partial\hat{T}}{\partial x},\frac{\partial\hat{T}}{\partial t},\frac{\partial^2\hat{T}}{\partial x^2},\cdots,\theta\right)=0,(x,t)\in\Gamma_f \tag{6}$$

$$BC: B\left(\hat{T},T(0,t),T(\infty,t),...,\theta\right)=0,(x,t)\in\Gamma_b \tag{7}$$

$$IC: I\left(\hat{T},T(x,0),...,\theta\right)=0,(x,t)\in\Gamma_i \tag{8}$$

Where x is the spatial variable, t is the temporal variable, $\hat{T}$ denotes the estimated temperature from the physics-informed neural network, $F(\cdot)$ is the residual form of the temperature advection-diffusion equation, and $B(\cdot)$ and $I(\cdot)$ are the residual forms of the boundary and initial conditions, respectively. By definition, all residual terms are evaluated at a set of scattered points (e.g., randomly distributed points or clustered points within the domain), constituting the training data set Γ of size $|\Gamma|$. Specifically, $\Gamma=\{x_1,x_2,\cdots,x_\Gamma\}$ and comprises three subsets: Γ_f , Γ_b , and Γ_i , which correspond to the computational domains of the three residual functions, respectively. θ represents the neural network parameters, consisting of weights W^l and biases b^l . Furthermore, to eliminate the discrepancy between the theoretical values obtained from the neural network solution and the actual temperature data, CFD software (Fluent 18.0) is used to establish a simulation model of a single pipe segment, generating high-fidelity temperature data. Data from selected measurement points are used to simulate sparse measurement data:

$$Data: M\left(\hat{T},T_D\right)=0 \tag{8}$$

Where T_D represents the temperature data generated by the CFD software, and $M(\cdot)$ is the residual form of the measurement network output. Subsequently, the aforementioned physical information and the high-fidelity temperature data are incorporated into the loss function. The loss function consists of four components: the partial differential equation loss, the boundary loss, the initial loss, and the data-driven loss:

$$L(\theta;\Gamma)=\lambda_f L_f(\theta;\Gamma_f)+\lambda_b L_b(\theta;\Gamma_b)+\lambda_i L_i(\theta;\Gamma_i)+\lambda_d L_d(\theta;\Gamma_d) \tag{9}$$

$$L_f(\theta;\Gamma_f)=\frac{1}{|\Gamma_f|}\sum_{x\in\Gamma_f}\left\|F\left(T,\frac{\partial T}{\partial x},\frac{\partial T}{\partial t},\frac{\partial^2 T}{\partial x^2},\cdots,\theta\right)\right\|^2 \tag{10}$$

$$L_b(\theta;\Gamma_b)=\frac{1}{|\Gamma_b|}\sum_{x\in\Gamma_b}\left\|B\left(\hat{T},T(0,t),T(\infty,t),...,\theta\right)\right\|^2 \tag{11}$$

$$L_i(\theta;\Gamma_i)=\frac{1}{|\Gamma_i|}\sum_{x\in\Gamma_i}\left\|I\left(\hat{T},T(x,0),...,\theta\right)\right\|^2 \tag{12}$$

$$L_d(\theta;\Gamma_d)=\frac{1}{|\Gamma_d|}\sum_{x\in\Gamma_d}\left\|M\left(\hat{T},T_D\right)\right\|^2 \tag{13}$$

Where $L_f(\theta;\Gamma_f)$, $L_b(\theta;\Gamma_b)$, $L_i(\theta;\Gamma_i)$, and $L_d(\theta;\Gamma_d)$ represent the PDE constraint loss, boundary constraint loss, initial constraint loss, and data loss, respectively. λ_f, λ_b, λ_i, and λ_d are the task weights for the four loss terms; $\|\cdot\|^2$ denotes the L^2-norm operator, which is the general form of the mean squared error used for training the FCNN; $|\Gamma_f|$, $|\Gamma_b|$, and $|\Gamma_i|$ denote the number of training points selected from the computational domains of Γ_f, Γ_b, and Γ_i, respectively; and $|\Gamma_d|$ represents the number of high-fidelity temperature data points.

Finally, the optimal network parameters θ^* are obtained by minimizing the loss function $L(\theta;\Gamma)$. When the loss function is minimized, violations of physical laws are correspondingly optimized. Algorithm 1 presents the overall framework for heating pipe modeling using PINN.

Algorithm 1: Single Heating Pipe Modeling

Input: Time t, position x, initial water temperature T in the pipe at position x and time t.

1: Establish a CFD simulation model for a single heating pipe, run the simulation, and obtain high-fidelity temperature data (T_D);

2: Generate training points $(x,t)\in\Gamma_f$, boundary points $(x,t)\in\Gamma_b$, and initial points $(x,t)\in\Gamma_i$ within the computational domain;

3: Construct a physics-informed neural network $\mathrm{N}^L(x)$ for the single heating pipe;

4: Initialize neural network parameters and task weights λ_f, λ_b, λ_i, λ_d;

5: **for** epoch=1 to end **do**

6: Calculate the residual functions according to Eqs. (6)–(8);

7: Compute $L_f(\theta;\Gamma_f)$, $L_b(\theta;\Gamma_b)$, $L_i(\theta;\Gamma_i)$, and $L_d(\theta;\Gamma_d)$ based on Eqs. (10)–(12);

8: Minimize the loss function $L(\theta;\Gamma)$ using Eq. (9) and update the network parameters θ;

9: Results: The temperature at the corresponding spatiotemporal coordinates inside the heating pipe;

10: **end for**

11: Save the optimal network parameters θ^* and task weights λ_f, λ_b, λ_i, λ_d.

Output: Physics-informed neural network parameters.

B. Transfer Learning based heating network modeling framework

Currently, numerical methods can accurately simulate pipeline temperature distributions given boundary and initial conditions. However, in practical district heating networks, boundary conditions are often imprecise, and available temperature measurement data are limited. Additionally, modeling each pipe in the network individually would lead to prohibitively high training costs. Therefore, this paper proposes a transfer learning-based modeling framework to achieve district heating network modeling, as illustrated in Fig. 4.

The transfer learning-based physics-informed neural network modeling framework consists of three stages: (i) CFD simulation data collection, (ii) pipeline surrogate model training, and (iii) pipeline discrepancy model training and network modeling. First, in the data collection stage, a simulation model of the district heating network described in Section 2.1 is established using CFD software. The simulation is run to collect high-fidelity network data T_{ND}. Next, a pipeline surrogate model is constructed using a physics-informed neural network, which is trained by incorporating physical laws and the high-fidelity CFD simulation data. Finally, based on the model parameters of the pipeline surrogate model, multiple pipeline discrepancy models are built. These models are trained using sparse measurement data and are then combined into a nested structure to form the complete district heating network model. The surrogate model learns the temperature evolution trend of a single pipe section, while the discrepancy model minimizes the difference between numerical simulations and actual measured data.

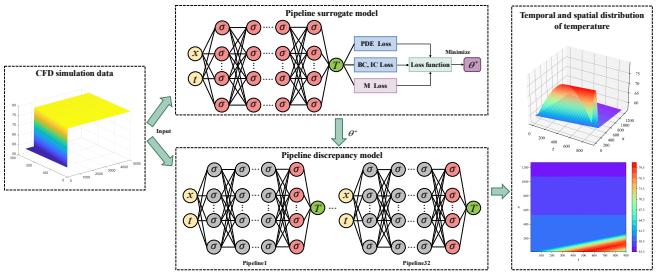


Fig.4. Transfer Learning based heating network modeling framework

The construction process of the pipeline surrogate model is consistent with the single-pipe modeling procedure described earlier. Trained on high-fidelity data, the model can accurately simulate the spatiotemporal temperature distribution inside the pipe. It takes the spatiotemporal coordinates of a typical heating pipe in the district heating network as input and outputs the temperature at the corresponding spatiotemporal coordinates within the pipe.

To reduce computational costs and minimize the discrepancy between the surrogate model and the real environment, high-fidelity sparse measurement data are simulated based on key points in the heating network. These data are used to train the discrepancy models for the 32 pipeline sections. The pipeline discrepancy models are trained using transfer learning. First, the network parameters θ^* and weight parameters λ_f, λ_b, λ_i, and λ_d of the pipeline surrogate model are copied to the pipeline discrepancy models. At this stage, the pipeline discrepancy models adopt the same architecture as the surrogate model, taking spatiotemporal coordinates as input and outputting temperature. Subsequently, during training, only the weights and biases of the last layer of the network are fine-tuned based on the sparse measurement data, while the parameters of the remaining layers are frozen to the pre-trained values of the surrogate model. Specifically, only the weight matrix W^{L-1} and bias matrix b^{L-1} of the $L-1$ hidden layer are updated. This approach minimizes the discrepancy between the pipeline surrogate model and the real measurement environment while reducing computational costs.

Since the studied district heating network consists of 32 interacting pipe sections, 32 independent neural networks need to be organized into a nested structure and trained simultaneously to solve this network problem. In the heating network, only pipes 1, 30, 31, and 32 have known initial and

Algorithm 2: Transfer Learning-Based District Heating Network Modeling Framework.

Input: Time t, position x, temperature of the water flow in the pipe at position x and time t.

1: Establish a district heating network model, run the simulation, and obtain temperature data T_{ND}.
2: According to Algorithm 1, construct a pipe surrogate model and initialize the model parameters and task weights.
3: Minimize the loss function to compute the optimal parameters θ^*.
4: Save the optimal network parameters θ^* and weight parameters λ_f, λ_b, λ_i, and λ_d of the surrogate model;
5: **for** k =1,2,…,32 do
6: **if** the initial and boundary conditions of the pipe are known;
7: Select measurement points according to the district heating network to obtain sparse measurement data $T^*_{ND,k}$;
8: Generate monitoring points $(x,t)\in\Gamma_{f,k}$, boundary points $(x,t)\in\Gamma_{b,k}$, and initial points $(x,t)\in\Gamma_{i,k}$ in the computational domain.
9: Construct a pipe discrepancy model based on the optimal network parameters θ^* and weight parameters λ_f, λ_b, λ_i, and λ_d;
10: Fine-tune the parameters of the last network layer and freeze the parameters of the other layers;
11: Minimize the loss function to obtain the optimal network parameters θ_k^*;
12: Results: Output the temperature at the corresponding spatiotemporal coordinates inside the heating pipe;
13: else if the initial and boundary conditions of the pipe depend on one connected pipe $p1$, and the length of pipe $p1$ is X;
14: Select measurement points according to the district heating network to obtain sparse measurement data $T^*_{ND,k}$;
15: Construct a pipe discrepancy model based on the optimal network parameters θ^* and weight parameters λ_f, λ_b, λ_i, and λ_d;
16: Use the temperature $\widehat{T}_{p1}(X,t)$ at the outlet of pipe $p1$ at time $t=0$ as the boundary and initial conditions in the pipe discrepancy model;
17: Fine-tune the parameters of the last network layer and freeze the parameters of the other layers;
18: Minimize the loss function to obtain the optimal network parameters θ_k^*;
19: Results: Output the temperature at the corresponding spatiotemporal coordinates inside the heating pipe;
20: **else** the initial and boundary conditions of the pipe depend on connected pipes $p1$ and $p2$;
22: Select measurement points according to the district heating network to obtain sparse measurement data $T^*_{ND,k}$;
22: Construct a pipe discrepancy model based on the optimal network parameters θ^* and weight parameters λ_f, λ_b, λ_i, and λ_d;
23: According to equations (16)-(17), calculate the boundary condition $T_k(0,t)$ and initial condition $T_k(x,0)$ of the pipe and input them into the pipe discrepancy model;
24: Fine-tune the parameters of the last network layer and freeze the parameters of the other layers;
25: Minimize the loss function to obtain the optimal network parameters θ_k^*;
26: **Results:** Output the temperature at the corresponding spatiotemporal coordinates inside the heating pipe;
27: Save the best network parameters θ^* and task weights λ_f, λ_b, λ_i, and λ_d;
end if
end for

boundary conditions. The output of any other pipe k depends on the output of one or several connected pipes based on boundary and initial conditions. When the output of pipe k depends on the output of a single connected pipe according to boundary and initial conditions—for example, the temperature of pipe 3 at node 2 of the district heating network depends on the temperature of pipe 1 estimated by PINN—let $\widehat{T}_1(257.6,t)$ denote the temperature estimated by PINN for pipe 1 at node 2, where 257.6 corresponds to the length of pipe 1. The temperature at the inlet of pipe 2 is then:

$$T_2(0,t)=\widehat{T}_1(257.6,t) \tag{14}$$

Where $T_2(0,t)$ denotes the temperature of pipe 2 at node 2. This serves as the boundary condition at the inlet of pipe 2, and the initial condition of the pipe is:

$$T_2(x,0)=\widehat{T}_1(0,0)f(x) \tag{15}$$

where $f(x)$ is an exponential decay function that simulates heat loss during the heat transfer process in the pipe. When the output of pipe k depends on the outputs of multiple connected pipes according to the boundary and initial conditions—for example, the temperatures of pipes 5 and 10 at node 5 of the district heating network depend on the temperatures of pipes 4 and 6 estimated by PINN—let $\widehat{T}_4(59.5,t)$ and $\widehat{T}_6(235.4,t)$ represent the temperatures estimated by PINN for pipes 4 and 6 at node 5, respectively, where 59.5 and 235.4 correspond to the lengths of pipes 4 and 6. Let m_4 and m_6 denote the mass flow rates of pipes 4 and 6, respectively. According to NBC2, the temperatures of pipes 5 and 10 at node 5 are identical and can be solved as follows:

$$T_5(0,t)=T_{10}(0,t)=\frac{\widehat{T}_4(59.5,t)\cdot m_4+\widehat{T}_6(235.4,t)\cdot m_6}{m_4+m_6} \tag{16}$$

Where, $T_5(0,t)$ and $T_{10}(0,t)$ represent the temperatures of pipe 5 and pipe 10 at node 5, respectively, serving as the boundary conditions at the inlets of pipe 5 and pipe 10. The initial conditions of the pipes are:

$$T_5(x,0)=T_{10}(x,0)=\widehat{T}_5(0,0)f(x)=\widehat{T_{10}}(0,0)f(x) \tag{17}$$

$$\widehat{T}_5(0,0)=\widehat{T_{10}}(0,0)=\frac{\widehat{T}_4(59.5,0)\cdot m_4+\widehat{T}_6(235.4,0)\cdot m_6}{m_4+m_6} \tag{18}$$

Where $T_5(x,0)$ and $T_{10}(x,0)$ denote the initial conditions of pipe 5 and pipe 10, respectively, and $\widehat{T}_5(0,0)$ and $\widehat{T_{10}}(0,0)$ represent the temperatures at the inlets of pipe 5 and pipe 10 at time $t=0$, respectively. Algorithm 2 presents the overall framework for district heating network modeling based on transfer learning.

IV. Experimental Analyses

A. Heating Pipeline Modeling Experiment

Accurate modeling of a single pipe section serves as the foundation for subsequent heating network modeling. To further validate the correctness of the proposed heating pipe modeling method based on physics-informed neural networks (PINN), the method proposed in this study is compared with a data-driven approach. The partial differential equations are solved using Fluent 18.0 software to obtain numerical solutions, i.e., PDE numerical solutions, which serve as the first type of reference numerical solution for comparison with the model proposed in this study. Additionally, to demonstrate the ability of the proposed model to reflect the dynamic characteristics of heat transfer, the calculation results from a unified modeling method are introduced as the second type of reference numerical solution. In the first experiment, numerical validation is performed using a pipe with a length of 200 meters. The specific parameters of the pipe, including the steady-state mass flow rate, cross-sectional area, and thermal conductivity, are listed in Table 1.

TABLE I. HEATING PIPE PARAMETERS

Parameters	Value	Physical unit
c	4182	$J/(kg\cdot°C)$
m	3	$kg\cdot s^{-1}$
ρ	1000	kg/m^3
S	0.01	m^2
λ	0.6	$W\cdot mK^{-1}$

For the original partial differential equation, the initial and boundary conditions are set as:

$$T(x,0)=60e^{-\frac{\lambda}{cm}x} \tag{19}$$

$$\begin{cases} T(0,t)=60+20\sin\left(\dfrac{t}{300}\right) \\ T(+\infty,t)=0 \end{cases} \tag{20}$$

Here, the boundary condition $T(+\infty,t)=0$ indicates that the relative temperature of the water flow at infinity is zero, i.e., consistent with the ambient temperature. The sinusoidal function $60+20\sin\left(\dfrac{t}{300}\right)$ represents the initial temperature variation function of the pipe, simulating the fluctuating temperature changes within the pipe. Additionally, to quantify the accuracy of the proposed model, we use Root Mean Square Error ($RMSE$), Mean Absolute Error (MAE), Mean Absolute Percentage Error ($MAPE$), and R^2 for evaluation.

$$RMSE=\sqrt{\frac{1}{N_{pred}}\sum_{i=1}^{N_{pred}}\left(T_{real}(t,x_i)-\hat{T}(t,x_i)\right)^2} \tag{21}$$

$$MAE=\frac{1}{N_{pred}}\sum_{i=1}^{N_{pred}}\left|T_{real}(t,x_i)-\hat{T}(t,x_i)\right| \tag{22}$$

$$MAPE=\frac{100\%}{N_{pred}}\sum_{i=1}^{N_{pred}}\left|\frac{T_{real}(t,x_i)-\hat{T}(t,x_i)}{T_{real}(t,x_i)}\right| \tag{23}$$

$$R^2=1-\frac{\sum_{i=1}^{N_{pred}}\left(T_{real}(t,x_i)-\hat{T}(t,x_i)\right)^2}{\sum_{i=1}^{N_{pred}}\left(T_{real}(t,x_i)-\hat{T}(t,x_i)\right)^2} \tag{24}$$

Where $T_{real}(t,x_i)$ denotes the ground truth temperature from numerical simulation, $\hat{T}(t,x_i)$ is the temperature estimated by the proposed physics-informed neural network, x_i represents the spatial coordinate of the i-th sampling position, and N_{pred} is the number of sampling points.

The pipeline model uses 8 fully connected hidden layers, each with 60 neurons. The hyperbolic tangent function is employed as the activation function. The Adam optimizer is utilized to accelerate convergence, offering better performance compared to the traditional stochastic gradient descent method. Additionally, to capture the nonlinear characteristics of physical laws, a decay process is introduced to adjust the learning rate. Specifically, during the first 1000 iterations, the learning rate is set to 0.01, and it is reduced by 10% every 3000 iterations thereafter. For the Adam optimizer, the maximum number of iterations is set to 50000.

Fig 5 presents the spatiotemporal temperature distributions of the pipeline calculated by the unified mathematical modeling method, the neural network modeling method, the PDE numerical solution, and the proposed physics-informed neural network modeling method, respectively. Fig 5(a–d) displays the numerical experimental results for a heating pipeline with a length of 200 m. Visually, the proposed knowledge-integrated neural network modeling method accurately predicts the axial temperature gradient of the pipeline, and the predicted spatiotemporal temperature distribution is consistent with the solution results of the original thermal-hydraulic equation. Figure 6 provides a more detailed comparative analysis of the four modeling methods. Fig 6(a–d) presents the temperature simulation results at pipeline positions of 0 m, 50 m, 100 m, and 150 m. Clearly, compared to the unified mathematical modeling method, the proposed method better reflects the dynamic characteristics of heat transfer in the pipeline, and it aligns more closely with the PDE numerical solution than the neural network modeling method.

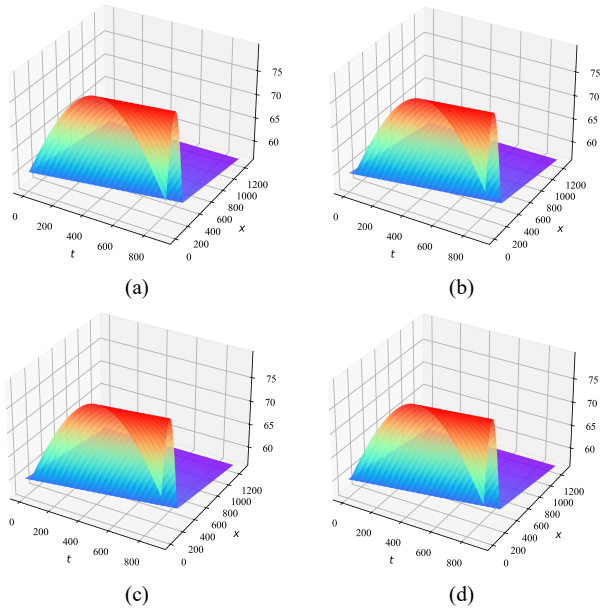


Fig 5: Numerical experimental results for a pipe with a length of 200 m. (a) The unified mathematical modeling method. (b) the PDE numerical solution (c) The physics-informed neural network modeling method (d) The neural network modeling method.

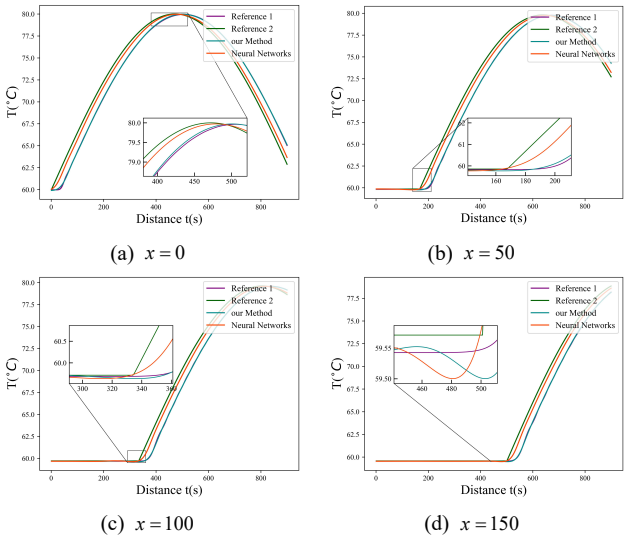


Fig.6. A more detailed comparative analysis.

Fig 7 presents the estimation errors of the pipeline temperature by the neural network modeling method and the proposed physics-informed neural network modeling method. In Fig. 7(b), the estimation errors are relatively large across

both the temporal and spatial domains. It is evident that the overall error of the proposed method, as illustrated in Fig. 7(c), is the smallest compared to the traditional methods. The results of *RMSE* , *MAE* , *MAPE* , and R^2 are listed in Table 2. The proposed method achieves an RMSE of 1.2234, representing an error reduction of 47.85% compared to the traditional neural network modeling method.

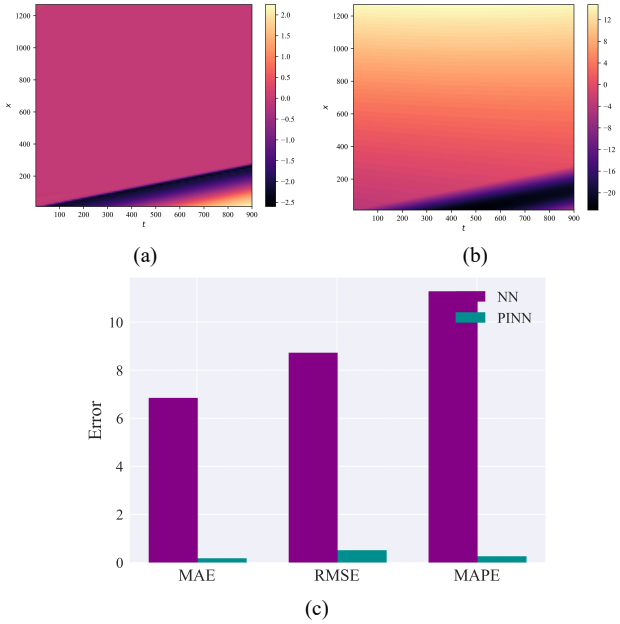


Fig.7. Error analysis of the two comparative methods relative to Reference Method 1. (a) The physics-informed neural network modeling method. (b) The neural network modeling method. (c) Error Comparison

TABLE II. ERROR INDEX STATISTICS

Index	PINN	NNs
RMSE	1.2234	2.3460
MAE	0.3748	1.3748
MAPE	10.3273	20.4785
R^2	0.9674	0.8264

B. District Heating Network Modeling Experiment

The second experiment uses the district heating network with 32 nodes from the Bali case study as an example. Table 4 presents the known parameters of the 32 heating pipes in the network, including thermal conductivity, length, cross-sectional area, and steady-state mass flow rate.

For pipes 1, 30, 31, and 32, the initial and boundary conditions are set as follows:

$$T(x,0) = 60e^{-\frac{\lambda}{cm}x} \tag{25}$$

$$\begin{cases} T(0,t) = 40\left[\dfrac{e^t}{(1+e^t)}+1\right] \\ T(+\infty,t) = 0 \end{cases} \tag{26}$$

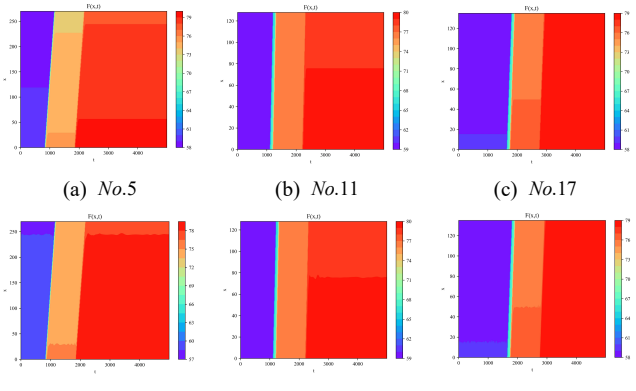
Where the step function $40\left[\dfrac{e^t}{(1+e^t)}+1\right]$ represents the initial temperature variation function of the pipe, used to simulate the stepwise temperature change within the pipe.

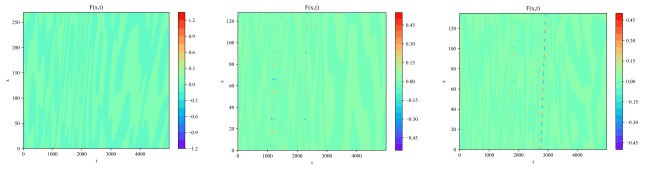
The model for each pipe in the heating network has the same structure as the pipe model in Experiment 1, consisting of 8 fully connected hidden layers with 60 neurons per layer.

TABLE III. DISTRICT HEATING NETWORK PIPELINE PARAMETERS

Pipe.no	L (m)	A(m^2)	HTC(W/mK)	MFR(Kg/s)
1	257.6	0.01227	0.321	4.7977
2	97.5	0.00126	0.210	0.7635
3	51.0	0.00126	0.210	0.7635
4	59.5	0.00785	0.327	3.2708
5	271.3	0.00080	0.189	0.6665
6	235.4	0.00332	0.236	0.8805
7	177.3	0.00126	0.210	0.8734
8	102.8	0.00126	0.210	0.8734
9	247.7	0.00126	0.210	0.8734
10	160.8	0.00785	0.327	3.4848
11	129.1	0.00126	0.210	1.5622
12	186.1	0.00785	0.327	4.1924
13	136.2	0.00503	0.278	4.1924
14	41.8	0.00196	0.219	1.3341
15	116.8	0.00080	0.189	0.6671
16	136.4	0.00080	0.189	0.6671
17	136.4	0.00080	0.189	0.6671
18	44.9	0.00503	0.278	2.1913
19	136.4	0.00080	0.189	0.5031
20	134.1	0.00080	0.189	0.5031
21	41.7	0.00332	0.236	1.1852
22	161.1	0.00080	0.189	0.6690
23	134.2	0.00080	0.189	0.6690
24	52.1	0.00332	0.236	0.1527
25	136.0	0.00080	0.189	0.6524
26	123.3	0.00080	0.189	0.6524
27	61.8	0.00126	0.210	1.4575
28	95.2	0.00080	0.189	0.6483
29	105.1	0.00080	0.189	0.6483
30	70.6	0.01227	0.321	2.7541
31	261.8	0.01227	0.321	3.5008
32	201.3	0.01227	0.321	2.2698

Fig 8 (a-c) presents the spatiotemporal temperature distribution results for pipes 5, 11, 17, and 23 of the network, obtained through the transfer learning-based heating network modeling framework. Fig 6 (d-f) shows the reference numerical solutions for pipes 5, 11, 17, and 23 of the network. Fig 6 (g-i) displays the error between the spatiotemporal temperature distribution results and the reference numerical solutions for pipes 5, 11, 17, and 23. It is evident that the error is relatively large when step changes in temperature occur within the pipes, but its maximum value does not exceed 0.15 K. The overall error of the heating network modeling framework is very small.





(g) *No.*5 (h) *No.*11 (i) *No.*17

Fig 8: Reference results, results from the proposed model, and errors of the spatiotemporal temperature distribution for pipes 5, 11, and 17

Fig 9 provides a comparison between the results of the other 28 pipeline sections in the network and the reference numerical solution at x=0 meters. It is evident that the obtained results are nearly identical to the reference numerical solution, further demonstrating the advantage of the transfer learning-based heating network modeling framework in terms of modeling accuracy.

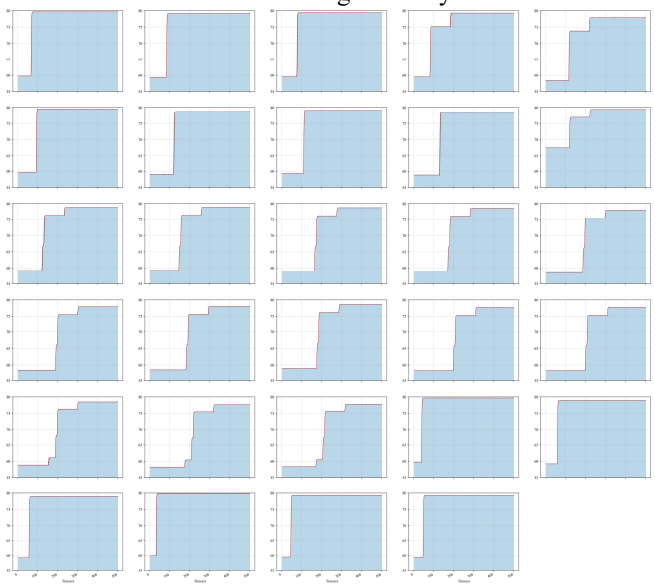


Fig 9: Comparison of temperature variation results and reference numerical solutions for the other 29 network pipelines at x = 0 m

V. References

Accurate modeling of heating pipelines (and pipeline networks) directly affects the optimal scheduling of centralized heating areas, impacting both heating capacity and environmental comfort. This study presents a knowledge-guided, data-driven modeling method for heating pipelines, which helps address the issues of sparse measurement data and missing physical information. First, the thermal diffusion partial differential equation based on computational fluid dynamics is embedded as prior knowledge into the neural network, adhering to fundamental physical laws. Subsequently, a data-driven model for heating pipelines is constructed based on given sparse temperature measurement data, revealing the patterns of dynamic temperature field variations. Finally, a transfer learning-based approach is employed for the implementation of heating network modeling. After pre-training a standard model, fine-tuning is performed based on different pipeline attributes to construct the heating network. Experimentally, a real network composed of 32 heating pipeline sections is used for numerical validation, achieving the first practical application and validating the accuracy of PINN for heating network problems.

References

[1] X. Zheng et al., "Digital twin modeling for district heating network based on hydraulic resistance identification and heat load prediction," Energy, vol. 288, p. 129726, 2024.

[2] S. Kuntuarova et al., "Design and simulation of district heating networks: A review of modeling approaches and tools," Energy, vol. 305, p. 132189, 2024.

[3] H. Lund et al., "The role of district heating in future renewable energy systems," Energy, vol. 35, no. 3, pp. 1381–1390, 2010.

[4] Z. Pan, Q. Guo, and H. Sun, "Interactions of district electricity and heating systems considering time-scale characteristics based on quasi-steady multi-energy flow," Applied Energy, vol. 167, pp. 230–243, 2016.

[5] A. Benonysson, B. Bøhm, and H. F. Ravn, "Operational optimization in a district heating system," Energy Convers. Manag., vol. 36, no. 5, pp. 297–314, 1995.

[6] M. Chertkov and N. N. Novitsky, "Thermal transients in district heating systems," Energy, vol. 184, pp. 22–33, 2019.

[7] V. D. Stevanović et al., "Prediction of thermal transients in district heating systems," Energy Convers. Manag., vol. 50, no. 9, pp. 2167–2173, 2009.

[8] J. Danielewicz et al., "Three-dimensional numerical model of heat losses from district heating network pre-insulated pipes buried in the ground," Energy, vol. 108, pp. 172–184, 2016.

[9] S. Lim, S. Park, H. Chung, M. Kim, Y.-J. Baik, and S. Shin, "Dynamic modeling of building heat network system using Simulink," Appl. Therm. Eng., vol. 84, pp. 375–389, 2015.

[10] Q. Chen, J. Hao, L. Chen, F. Dai, and Y. Xu, "Integral transport model for energy of electric-thermal integrated energy system," Autom. Electr. Power Syst., vol. 41, no. 13, pp. 7–13, 2017

[11] G. Schweiger, P.-O. Larsson, F. Magnusson, P. Lauenburg, and S. Velut, "District heating and cooling systems – Framework for Modelica-based simulation and dynamic optimization," Energy, vol. 137, pp. 566–578, 2017.

[12] J. Yang, N. Zhang, A. Botterud, and C. Kang, "On an equivalent representation of the dynamics in district heating networks for combined electricity-heat operation," IEEE Trans. Power Syst., vol. 35, no. 1, pp. 560–570, 2020.

[13] Y. Liu, N. Zhang, Y. Wang, J. Yang, and C. Kang, "Data-driven power flow linearization: A regression approach," IEEE Trans. Smart Grid, vol. 10, no. 3, pp. 2569–2580, 2019.

[14] H. Tian, H. Zhao, C. Liu, J. Chen, Q. Wu, and V. Terzija, "A dual-driven linear modeling approach for multiple energy flow calculation in electricity–heat system," Appl. Energy, vol. 314, p. 118872, 2022.

[15] S. Guo, W. Ji, C. Wang, T. Song, and J. Wang, "Hydraulic-thermal coupling dynamic models based on mechanism and data-driven methods of the heating networks in integrated energy systems," Energy Convers. Manag., vol. 292, p. 117353, 2023.

[16] R. Yentis and M. E. Zaghloul, "VLSI implementation of locally connected neural network for solving partial differential equations," IEEE Trans. Circuits Syst. I, Fundam. Theory Appl., vol. 43, no. 8, pp. 687–690, 1996.

[17] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," J. Comput. Phys., vol. 378, pp. 686–707, 2019.

[18] S. Cai, Z. Wang, S. Wang, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks for heat transfer problems," J. Heat Transfer, vol. 143, no. 6, p. 060801, 2021.

[19] N. Zobeiry and K. D. Humfeld, "A physics-informed machine learning approach for solving heat transfer equation in advanced manufacturing and engineering applications," Eng. Appl. Artif. Intell., vol. 101, p. 104232, 2021.

[20] X. Liu, W. Peng, Z. Gong, W. Zhou, and W. Yao, "Temperature field inversion of heat-source systems via physics-informed neural networks," Eng. Appl. Artif. Intell., vol. 113, p. 104902, 2022.

[21] K. C. B. Steer, A. Wirth, and S. K. Halgamuge, "Control period selection for improved operating performance in district heating networks," Energy Build., vol. 43, no. 2, pp. 605–613, 2011.