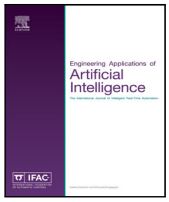
Engineering Applications of Artificial Intelligence 117 (2023) 105516

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Physics-informed convolutional neural networks for temperature field prediction of heat source layout without labeled data

Xiaoyu Zhao, Zhiqiang Gong *, Yunyang Zhang, Wen Yao, Xiaoqian Chen

Defense Innovation Institute, Chinese Academy of Military Science, No. 53, Fengtai East Street, Beijing 100071, China

ARTICLE INFO

Keywords:
Physics-informed convolutional neural networks
Thermal simulation
Heat source layout
Finite difference method
Temperature field prediction

ABSTRACT

Recently, surrogate models based on deep learning have attracted much attention for engineering analysis and optimization. Since constructing data pairs in most engineering problems is time-consuming, data acquisition is becoming the predictive capability bottleneck of most deep surrogate models, which also exist in surrogate for thermal analysis and design. In contrast with data-driven learning, enforcing the physical laws in building surrogates has emerged as a promising alternative to reduce the dependence on annotated data. This paper develops a physics-informed convolutional neural network (CNN) for the thermal simulation surrogate without labeled data. Firstly, we leverage the finite difference method to integrate heat conduction equation and loss function construction, guiding surrogate model training to minimize the violation of physical laws. Since the solution is sensitive to boundary conditions, we properly impose hard constraints by padding in the Dirichlet and Neumann boundaries. The proposed network can learn a mapping from heat source layout to the steady-state temperature field without labeled data, which equals solving an entire family of partial difference equations (PDEs). Moreover, the neural network architecture is well-designed to improve the prediction accuracy of the problem at hand, and pixel-level online hard example mining is proposed to overcome the imbalance of optimization difficulty in the computation domain, which is beneficial to the network training of physics-informed learning. The experiments demonstrate that the proposed method can provide comparable predictions with numerical methods and data-driven deep learning models. We also conduct various ablation studies to investigate the effectiveness of the proposed network components and training methods in this paper. Furthermore, the developed methods can be applied to other design and optimization applications which need to solve parameterized PDEs.

1. Introduction

With the increasing integration of electronic equipment, thermal management becomes an essential part of the electronic system design to ensure the performance and reliability of inner components. One effective way for thermal management is to optimize the heat source layout inside the electronic device (Chen et al., 2016, 2017; Aslan et al., 2018), where physics simulation plays an important role in providing temperature field computations to support layout evaluation in the layout optimization loop. The physics simulation usually involves solving complex partial differential equations (PDEs). It is expensive and time-consuming in many science and engineering problems (Deng et al., 2022b), including thermal analysis and design, as PDEs with different design parameters must be repeatedly solved. Surrogate models, such as Gaussian processes (Lee et al., 2020), kriging (Liu et al., 2017), SVM regression (Moustapha et al., 2018), and neural network (Misiulia et al., 2017), are computationally attractive to replace the costly and repetitive physics simulations by approximating the input–output relation of concerned system. Deep neural network (DNN) has recently made great processes in image recognition (Ronneberger et al., 2015), state estimation (Lu et al., 2021), and fault diagnosis (Deng et al., 2022a). Well-known for its power in solving high-dimensional and great nonlinear problems, more researches have focused on the deep learning-based surrogate model.

This paper studies the deep surrogate model in one thermal management task named temperature field prediction of heat source layout (HSL-TFP). More specifically, we aim to build a low-computational surrogate model using DNNs to replace the time-consuming thermal simulation in heat source layout optimization. There exist some works on the data-driven surrogate model based on DNNs. Chen et al. (2020) modeled the temperature field prediction as an image-to-image regression task, and feature pyramid network is employed as the backbone of DNNs to make predictions. Compared with numerical methods, the DNNs surrogate reduces the computation time and improves the efficiency of layout optimization. For evaluating the performance of

* Corresponding author.
E-mail address: gongzhiqiang13@nudt.edu.cn (Z. Gong).

https://doi.org/10.1016/j.engappai.2022.105516
Received 20 April 2022; Received in revised form 28 September 2022; Accepted 6 October 2022
Available online 21 October 2022
0952-1976/

various methods, Chen et al. (2021) summarized the HSL-TFP task benchmark from data generation, baseline models, and evaluation metrics. The data-driven deep surrogate model are also investigated in structural mechanics (Kallioras et al., 2020; Ates and Gorguluarslan, 2021), fluid mechanics (Zhu et al., 2019; Li et al., 2020a), and materials physics (Teichert and Garikipati, 2019). As deep learning methods operate in the big data regime, amounts of training data are usually necessary to achieve reliable predictive performance. However, preparing sufficient simulation data for training is similarly expensive and time-consuming in such engineering and scientific fields.

Incorporating the physical knowledge leveraged in numerical simulation into DNN training is promising to reduce the cost of preparing labeled data. Some similar researches exist called physics-informed neural networks (PINN) or physics-constrained neural networks that assist or replace numerical simulation to solve governing equations of physical systems. Classical PINN primarily focuses on solving one PDE with specific parameters by fully-connected neural networks (FC-NNs). For example, Raissi et al. (2019) exploited PINN by developing deep learning techniques. It leverages the capability of the deep neural network as universal function approximators and utilizes the techniques of automatic differentiation (Baydin et al., 2018). The physics-informed loss function of PINN is constructed based on governing PDEs and is used to regularize the output for minimizing the violation of PDEs. This representative PINN has been studied in many scientific problems to solve various PDEs, including high-speed flows (Mao et al., 2020), subsurface flows (Wang et al., 2020), metamaterial design (Liu and Wang, 2019), power system (Misyris et al., 2020), and heat transfer (Zobeiry and Humfeld, 2021). These studies have shown the remarkable potential of using physics-informed learning to reduce the dependence on training data. For the HSL-TFP task at hand, the heat conduction equation that governs thermal system is known that can assist to build surrogate model with the physics-informed neural network technology.

In this paper, we develop a physics-informed convolutional neural network (CNN) for surrogate modeling of thermal simulation. The training of surrogate model is data-free because the physical knowledge is joined into the construction of loss function to guide the network to minimize the violation of heat conduction equation. the heat source layouts and temperature fields are the inputs and outputs of the network, corresponding to parameters and solution functions in heat conduction equation, which are discretely represented by image-like grid data. CNN architecture is chosen as the backbone rather than FC-NNs to learn the mapping from layout to steady-state temperature field, considering the efficiency of parameter sharing and local connection in handling large-scale and high-dimensional problems. From the perspective of solving PDEs, the proposed surrogate model is to learn the mapping from parametric dependence in parameter space to the corresponding PDE solution. In contrast with PINN works based on FC-NNs to solve one instance of PDE, the proposed method is a surrogate to learn an entire family of PDEs. Similarly, some physics-informed FC-NN methods (Sun et al., 2020) have explored the parameterized spaces, including initial conditions, boundary conditions, and equation parameters, but FC-NN takes scalability problems reason of high-dimensional input spaces.

In addition, we propose some methods to improve training efficiency of the proposed physical-informed surrogate model. the prediction and its differential items are required for loss computation to optimize the network parameters during training. In the CNN architecture, the differential items are calculated by finite difference method that differs from the automatic differentiation employed in physics-informed FC-NN works. Besides, treatment of boundary conditions affects training efficiency and solution accuracy (Zhu et al., 2019; Gao et al., 2021), and even causes the ill-posed solution. To address this issue, we impose a hard constraint on the boundary by padding instead of the soft constraint to add a penalty term (Raissi et al., 2019) in loss function. It is well-known that physical-informed neural network is usually hard to train. For the task at hand, this paper proposes pixel-level online hard example mining to weigh each position adaptively, considering different optimization difficulties in the computation domain. It can drive network optimizers to make more efforts in the hard-optimized domain. The experiments conducted on two HSL-TFP cases show that our label-free method can achieve competitive results compared with data-driven and numerical methods.

In fact, the CNN have been applied in various physics simulation field, such as discovering underlying PDEs discovery (Long et al., 2018, 2019), PDEs solving (Yao et al., 2020; Zhu et al., 2019), and surrogate model construction (Zhu et al., 2019). Yao et al. (2020) presented that finite element method (FEA) models for PDEs is a special CNN, and FEA-Net was developed to predict the response of materials and structures. Kim et al. (2019) discretized and parameterized fluid simulation velocity fields, and a CNN-based generative model is provided to synthesize fluid simulations. Compared with previous studies, our work involves PDEs solving and surrogate model constructing studies, and the proposed physics-informed CNN is utilized as surrogate model in thermal simulation tasks. Furthermore, the impact of different network components on prediction performance is thoroughly discussed in our paper, which is helpful for the research of network structure design in this field. In conclusion, the novel contributions of our work are as follows.

(1) We develop a physics-informed loss function for HSL-TFP based on heat conduction equation and finite difference method. The physics-informed loss can train the network without labeled data.
(2) We introduce to impose hard constraints on the Dirichlet and Neumann boundary, which will benefit the stable and fast convergence of network training.
(3) A UNet-like architecture is proposed, and we emphasize discussing the performance impact of different network components.
(4) To balance the optimization difficulty in the whole computation domain, we introduce pixel-level online hard example mining for HSL-TFP. The experiments demonstrate that the proposed method can present comparable predictions with numerical and data-driven methods.

The rest of this paper is organized as follows. The description of the HSL-TFP problem is introduced in Section 2. Section 3 presents the construction of physics-informed loss function, the hard constraints on boundary conditions, the CNN-based architecture, and the pixel-level hard example mining. The results of two cases with varying difficulty are presented and discussed in Section 4. Finally, we conclude the paper in Section 5.

2. Temperature field prediction of heat source layout (HSL-TFP)

Assume that there is a two-dimensional rectangular heat-conducting plate with several electronic components. The plate makes heat exchange with the external environment by isothermal boundary. The components generate heat when working, dissipating heat by conduction. Each component can be simplified as a heat source for this problem, and the heat source layout will affect the steady temperature field distribution in the whole domain. This paper expects to employ DNN to rapidly make temperature field predictions for heat source layout.

Similar to problems stated by Chen et al. (2016, 2017), Aslan et al. (2018) and Chen et al. (2020, 2021), the two-dimensional conducting domain with several rectangle heat source is set as shown in Fig. 1. An isothermal boundary of length δ and temperature T_0 in the bottom exchanges heat with external circumstances. Other boundaries are adiabatic. The steady temperature field of the domain satisfies Poisson's equation, which is expressed as follows:

$$\frac{\partial}{\partial x}\left(\lambda\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda\frac{\partial T}{\partial y}\right)+\phi(x,y)=0,\ (x,y)\in\Omega, \tag{1}$$

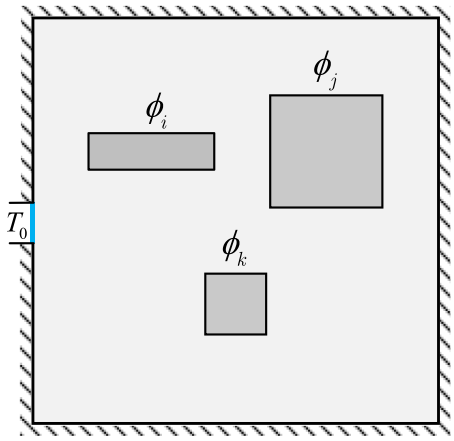


Fig. 1. Schematic of the square domain with several heat sources. The heat sink is located in the middle of left boundary with constant temperature T_0, and the rest is Neumann boundary. ϕ_i is the intensity of the ith heat source.

where constant λ is the thermal conductivity of the domain. Ω is the computation domain. $\phi(x, y)$ is the intensity distribution function, which correlates with the heat source layout. Assume that ϕ_i and Γ_i respectively represent the intensity and distribution area of the heat source i, $\phi(x, y)$ can be modeled as follows:

$$\phi(x,y) = \begin{cases} \phi_i, & (x,y) \in \Gamma_i \quad i \in 1,\dots,n \\ 0, & \text{others} \end{cases}. \tag{2}$$

For the problem in this paper, the equation should satisfy the Dirichlet Boundary and Neumann Boundary, which are formulated as follows:

$$T(x,y) = T_0, (x,y) \ \in \partial\Omega_D, \tag{3}$$

$$\lambda \frac{\partial T(x,y)}{\partial \boldsymbol{n}} = 0, (x,y) \ \in \partial\Omega_N, \tag{4}$$

where Ω_D and Ω_N respectively denote the Dirichlet and Neumann Boundary. The equation can be solved by traditional numerical methods, such as finite difference method (FDM), finite element method (FEM), and finite volume method (FVM). Consequently, traditional numerical solvers sometimes are inefficient when calling the solver frequently. In this paper, we investigate to replace the numerical solver with neural networks to make real-time temperature field predictions.

3. Physics-informed convolutional neural networks for HSL-TFP

In this paper, each heat source layout corresponds to one PDE parameterized by the intensity distribution function $\phi(x, y)$. For the trained CNN, we expect that the neural network takes $\phi(x, y)$ as input and predicts the two-dimensional temperature field T. Therefore, the network is essentially learning operators, mapping intensity distribution function to the PDE solution.

In addition, some works have shown the power of deep networks in learning operators. Motivated by the success of deep learning, Bhattacharya et al. (2021) developed a data-driven framework to describe the mapping between infinite-dimensional spaces and applied it to the elliptic PDE. Patel et al. (2021) introduced a regression framework to discover operators from molecular simulation data. Anandkumar et al. (2020) and Li et al. (2020b) proposed a graph kernel network to describe the mapping between different finite-dimensional approximations for infinite-dimensional spaces. By introducing Fourier transform, Li et al. (2020a) presented the Fourier neural operator, a resolution-invariant solution operator for the Navier–Stokes equation. From the universal approximation theorem, Lu et al. (2021) proposed a network named deep operator network (DeepONet), which can make small generalization errors for deterministic and stochastic PDEs.

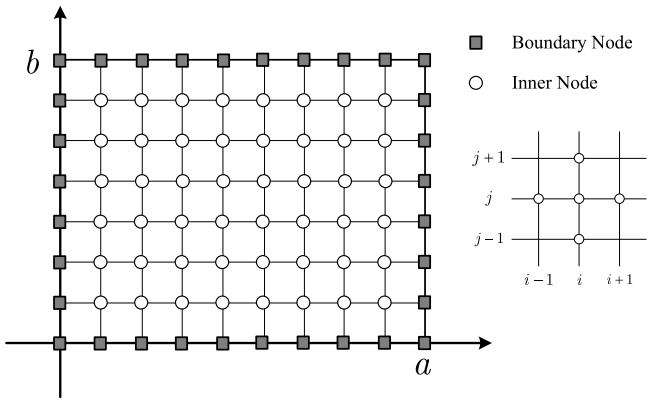


Fig. 2. Illustration of mesh division in rectangle domain.

In contrast with these methods, the learning of deep operators is driven by physics equations in our work. For the HSL-TFP task, discrete representations of intensity distribution and solution functions are obtained by dividing mesh. These representations can be viewed as images, and CNN is employed for the image-to-image regression problem in finite dimension space. Guided by physics knowledge, CNN can learn to solve an entire family of heat conduction equations.

This section will first provide the construction of the physics-informed loss function for HSL-TFP, and the convolution layer can handily compute the loss. Then, we introduce hard enforcement in prediction to satisfy the boundary condition. Finally, the CNN architecture based on UNet is described.

3.1. Physics-informed loss function

The Physics-informed CNN aims to learn the steady-state solutions of parametric PDEs described in Eq. (1). Let G be an operator taking the intensity distribution function ϕ, and $G(\phi)$ is solution function of the corresponding PDE. For each point (x, y) in the solution $G(\phi)$, $G(\phi)(x, y)$ outputs the temperature, which is a real number. We consider the two-dimensional heat conduction problem for HSL-TFP, and the computation domain is set to the rectangle. Without loss of generality, the rectangle domain and boundary are described as follows:

$$D_I = \{(x,y) | 0 < x < a, 0 < y < b\}, \tag{5}$$

$$\partial D = \{(x,y) | 0 \le x \le a, y = 0 \text{ or } b; x = 0 \text{ or } a, 0 \le y \le b\}. \tag{6}$$

To represent the function ϕ and $G(\phi)$, we discretize the function domain at finite amounts of location, and the function ϕ and $G(\phi)$ are treated as an image. Considering the great success of CNN in image processing, we expect that CNN can well learn the operator G. As shown in Fig. 2, the computation domain D bounded by ∂D is divided into $m \times n$ rectangle mesh, and the step along the x-aixs and y-axis are $h = a/n$ and $k = b/m$. Inspired by the finite difference method, the derivative terms of the solution function $G(\phi)$ could be obtained by differential approximation. Therefore, we can construct a loss function to drive the solution function $G(\phi)$ to satisfy the governing equation. For inner-domain points D_I, the five-point finite difference is built by Taylor series expansion:

$$\begin{aligned} &\frac{1}{h^2}\left[T(x_i + h, y_j) - 2T(x_i, y_j) + T(x_i - h, y_j)\right] \\ &= \frac{\partial^2 T}{\partial x^2} + \frac{h^2}{24}\left[\frac{\partial^4}{\partial x^4} T\left(\xi_1, y_j\right) + \frac{\partial^4}{\partial x^4} T\left(\xi_2, y_j\right)\right], \end{aligned} \tag{7}$$

where $x_{i-1} \le \xi_1, \xi_2 \le x_{i+1}$. similarly, the difference form of $\partial^2 T/\partial y^2$ is presented as follows:

$$\begin{aligned} &\frac{1}{k^2}\left[T(x_i, y_j + k) - 2T(x_i, y_j) + T(x_i, y_j - k)\right] \\ &= \frac{\partial^2 T}{\partial y^2} + \frac{k^2}{24}\left[\frac{\partial^4}{\partial y^4} T\left(x_i, \eta_1\right) + \frac{\partial^4}{\partial y^4} T\left(x_i, \eta_2\right)\right], \end{aligned} \tag{8}$$

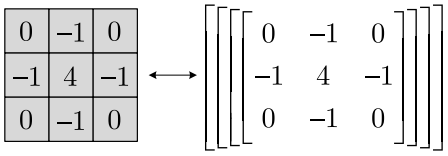


Fig. 3. Laplace convolution kernel for second difference.

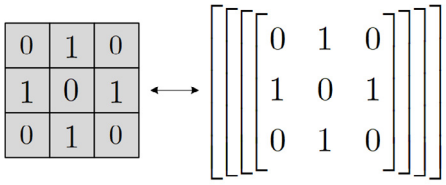


Fig. 4. Convolution kernel for computing $T(x_i, y_j)'$.

where $y_{i-1} \le \eta_1, \eta_2 \le y_{i+1}$. According to Eq. (7) and (8), the difference equation of Eq. (1) is obtained as follows:

$$\begin{aligned}&\frac{\partial}{\partial x}\left(\lambda\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda\frac{\partial T}{\partial y}\right)+\phi(x,y)\\&=\phi(x,y)+\frac{\lambda}{h^2}\left[T(x_i+h,y_j)-2T(x_i,y_j)+T(x_i-h,y_j)\right]\\&\quad+\frac{\lambda}{k^2}\left[T(x_i,y_j+k)-2T(x_i,y_j)+T(x_i,y_j-k)\right]+O(h^2+k^2).\end{aligned}\tag{9}$$

The truncation error of the five-point finite difference is $O(h^2+k^2)$. For convenience of discussion, we use x_{i-1}, x_{i+1} to represent $(x_i - h)$ and $(x_i + h)$, and the same is for y-axis. For each point $T(x_i, y_j)$ of mesh in D_I, it should satisfy the differential equation as follows:

$$\begin{aligned}&\frac{T\left(x_{i+1},y_j\right)-2T\left(x_i,y_j\right)+T(x_{i-1},y_j)}{h^2}\\&+\frac{T\left(x_i,y_{j+1}\right)-2T\left(x_i,y_j\right)+T(x_i,y_{j-1})}{k^2}+\frac{\phi(x,y)}{\lambda}=0.\end{aligned}\tag{10}$$

The loss function can be constructed based on the above difference equation. This paper makes steps h and k equal in mesh generation for easily calculating by a convolution kernel. Under this setting, Eq. (10) can be converted to the following form:

$$\begin{aligned}&4\cdot T\left(x_i,y_j\right)-T\left(x_{i+1},y_j\right)-T(x_{i-1},y_j)-T\left(x_i,y_{j+1}\right)-T(x_i,y_{j-1})\\&=\frac{h^2\phi(x_i,y_j)}{\lambda}.\end{aligned}\tag{11}$$

The temperature field T is the solution $G(\phi)$ predicted by CNN. We expect each inner-domain point of T to satisfy the above difference equation, and the second difference can be easily computed by the Laplace convolution kernel shown in Fig. 3. However, we discover that it is hard to train the neural network when roughly employing the Laplace convolution. Training is more efficient when the physics-informed loss function is constructed in the following form:

$$\mathcal{L}=\frac{1}{|D_I|}\sum_{(x,y)\in D_I}\left\|T\left(x_i,y_j\right)-\frac{1}{4}T\left(x_i,y_j\right)'\right\|,\tag{12}$$

$$\begin{aligned}T(x_i,y_j)'&=T(x_{i+1},y_j)+T(x_{i-1},y_j)+T(x_i,y_{j+1})\\&\quad+T(x_i,y_{j-1})+\frac{h^2\phi(x,y)}{\lambda},\end{aligned}\tag{13}$$

where $T(x_i, y_j)'$ is only the intermediate variable. It can also be computed by the convolution kernel, which is shown in Fig. 4. In particular, when computing $T(x_i, y_j)'$, the computation graph among $T(x_{i+1}, y_j)$, $T(x_{i-1}, y_j)$, $T(x_i, y_{j+1})$, and $T(x_i, y_{j-1})$ need not be constructed, which avoids that the error of $T(x_i, y_j)'$ affects neighboring points in back propagation. Although it is essentially derived from the difference equation of Eq. (10), this loss function formulation can significantly improve network training efficiency. In addition, the loss function can also be constructed by another difference scheme, such as the nine-point difference.

Why can the physics-informed loss function in this paper guide the neural network to learn the operator for solving an entire family of PDEs? It may be explained with the aid of the iteration method. Iterative numerical methods are iterative algorithms to determine the solutions for the linear equations system. It is from an initial solution and produces the approximation solution sequence based on an iteration format. Jacobi method is a common iteration method, and the update rule for Poisson's equation is denoted in the following form:

$$\begin{aligned}T\left(x_i,y_j\right)'&=\frac{1}{4}(T(x_{i+1},y_j)+T(x_{i-1},y_j)+T(x_i,y_{j+1})\\&\quad+T(x_i,y_{j-1}))+\frac{h^2}{4\lambda}\phi(x_i,y_j).\end{aligned}\tag{14}$$

The form of the loss function is consistent with Jacobi iteration. In each back propagation, the loss function guides the network to predict the temperature field approximating the value of the next iteration. Though the network learns to solve an entire family of PDEs, not one PDE, the iteration information of other PDEs is also helpful for training because of the similarity among parametric PDEs.

3.2. Hard-constrained boundary condition

The boundary conditions uniquely determine the solution of the steady-state problem. The PDE solving becomes ill-posed if constraints of boundary conditions (BCs) are not properly imposed on the computation domain. One way is to apply soft constraints, which add penalty terms in the loss function to impel the neural network to predict specific values in the boundary. Since the solution is sensitive to the boundary values, the soft constraint will negatively affect the early training when the boundary values do not fit exactly. Sun et al. (2020) introduced that soft boundary constraint will slow the convergence of PINN and devised a DNN structure to enforce the initial and boundary conditions. For CNN-based architecture, Gao et al. (2021) also studied the hard boundary condition enforcement for solving parameterized PDE. To improve the stability and convergence speed, we also perform hard-constraint on the Dirichlet and Neumann boundary conditions for the HSL-TFP.

For the Dirichlet boundary condition defined in Eq. (3), the values in Dirichlet boundary are directly replaced by the constant value T_0, which enforces that these values do not vary during training. For the Neumann boundary condition (Eq. (4)), it is assumed that there is extra mesh $\partial\Omega'$ surrounding the computation domain $\Omega \cup \partial\Omega$. According to the central difference formula, it should satisfy the following formulation:

$$\begin{cases}\lambda\dfrac{T(x_{n+1},y_j)-T(x_{n-1},y_j)}{2h}=0, & 0\le j\le m,\\ \lambda\dfrac{T(x_{-1},y_j)-T(x_1,y_j)}{2h}=0, & 0\le j\le m,\\ \lambda\dfrac{T(x_i,y_{m+1})-T(x_i,y_{m-1})}{2h}=0, & -1\le i\le n+1,\\ \lambda\dfrac{T(x_i,y_{-1})-T(x_i,y_1)}{2h}=0, & -1\le i\le n+1.\end{cases}\tag{15}$$

Therefore, we can enforce the padding values equal to the corresponding inner-nodes as follows:

$$\begin{cases}T(x_{n+1},y_j)=T(x_{n-1},y_j), & 0\le j\le m,\\ T(x_{-1},y_j)=T(x_1,y_j), & 0\le j\le m,\\ T(x_i,y_{m+1})=T(x_i,y_{m-1}), & -1\le i\le n+1,\\ T(x_i,y_{-1})=T(x_i,y_1), & -1\le i\le n+1.\end{cases}\tag{16}$$

The schematic of enforcing hard-constrain in Dirichlet and Neumann boundary is shown in Fig. 5. Before computing the loss for

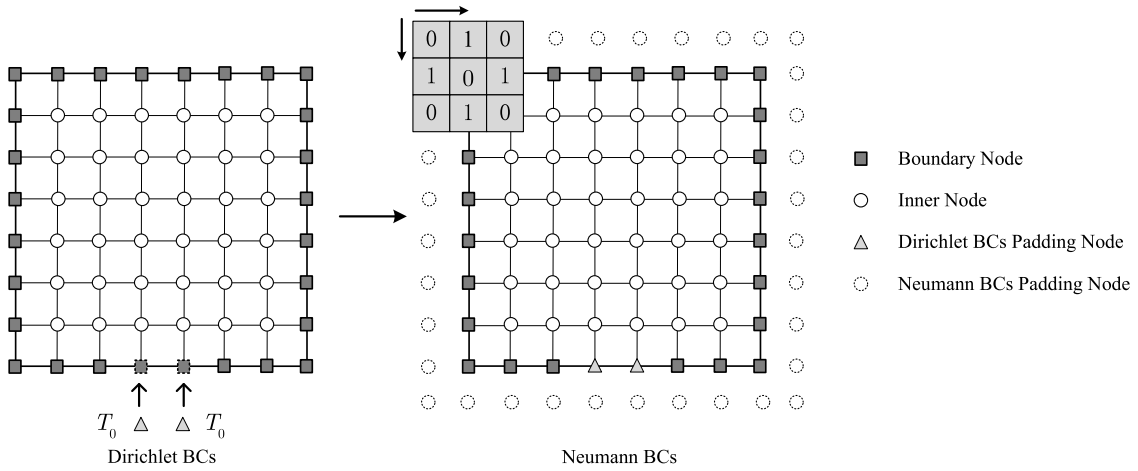


Fig. 5. Illustration of enforcing hard constraint in Dirichlet and Neumann boundary.

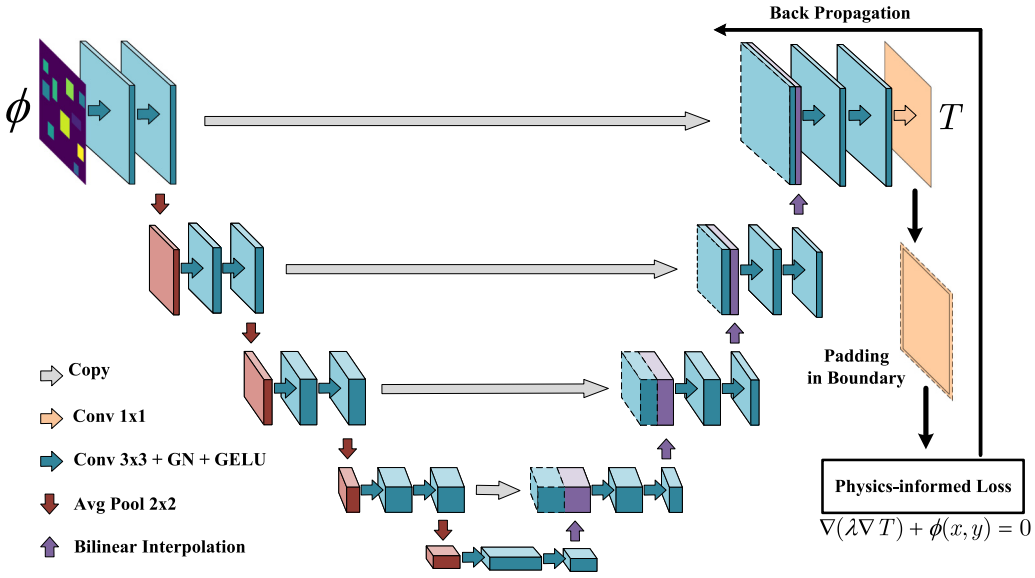


Fig. 6. Pipeline of physics-informed CNN for HSL-TFP. The CNN architecture is modified based on UNet for this specific task.

gradient-based optimization, the prediction is padded with constant boundary values for Dirichlet BCs. Then the surrounding domain is expanded with inner-domain nodes for Neumann BCs. Finally, the processing map is applied to obtain the T' according to Eq. (12), and the loss is calculated without the values in Dirichlet BCs.

$$\mathcal{L} = \frac{1}{|D_I \cup D_N|} \sum_{(x_i,y_j)\in(D_I\cup D_N)} \left\| T(x_i, y_j) - T(x_i, y_j)' \right\| \tag{17}$$

3.3. CNN architecture and framework for HSL-TFP

The inputs and outputs are both gridding data when applying the finite difference method for the two-dimensional steady-state prediction problem. When flattening the data, FC-NN will suffer from scalability problems reason of the high-dimensional data space. Compared with FC-NN, CNN is superior in processing image-style data. In this paper, the UNet (Ronneberger et al., 2015) is employed as the base model, which is one excellent image-to-image CNN architecture. UNet is an encoder–decoder framework consisting of a contracting path to capture global information and an asymmetric expanding path to make high-resolution predictions. In particular, there is a skip connection in the same level of feature map for feature fusion from the encoder to decoder. Due to these special structures, UNet has achieved great success in many computer vision tasks.

Unlike classification problems, the task at hand is an image-to-image regression problem. We modified UNet architecture, shown in Fig. 6. The inputs and outputs are the discretized representation of the source term $\phi(x, y)$ and T in Eq. (1). The blue arrow is the block with the 3×3 convolution layer, Group Normalization (GN), and GELU (Ronneberger et al., 2015) activation function. We choose GN (Wu and He, 2018) because it is more stable than Batch Normalization (BN) in this task. Since the intensity values without heat source are zeros, the intensity distribution map is sparse. It changes drastically along the edge of the heat source, which is significantly different from visual images. The statistics of BN would not be exact and will harm training. In addition, GELU activation is expected to increase the ability in nonlinear fitting. The red downward arrow is the downsampling block using average pool with 2×2 filters and stride 2. Symmetrically, the upward arrow is the $2 \times$ upsampling block with bilinear interpolation rather than the transpose convolution. Since the architecture is symmetrical, the padding of the 3×3 convolution layer is modified to 1 in contrast with original UNet, which makes the sizes of input and output under the same grid division equal. Besides, it is worth noticing that the padding mode of all convolution layers is all set to 'reflect'.

When conducting network training, we discovered large differences in the optimization difficulty of different regions. The difference equations connect the points in the physics-informed loss function, which

Table 1
The configuration of heat source in case 1.

Component	Size (m × m)	Intensity (w/m^2)
1–20	0.01 × 0.01	10000

means errors in these regions will propagate to others. We propose pixel-level online hard example mining (P-OHEM) to balance the optimization dynamically to address this issue. The error in each pixel of the output map is represented as

$$\delta_{ij} = \left\| T(x_i, y_j) - T(x_i, y_j)' \right\|. \tag{18}$$

Making δ the set of pixel-level errors in map, the weight of pixel in position (i, j) can be denoted as

$$w_{ij} = \eta_1 + \eta_2 \frac{\delta_{ij} - \min(\delta)}{\max(\delta) - \min(\delta)}, \tag{19}$$

where η_1 and η_2 are the shift and scale factors. Therefore, the loss function with P-OHEM is expresses as follows:

$$\mathcal{L} = \frac{1}{|D_I \cup D_N|} \sum_{(x_i, y_j) \in (D_I \cup D_N)} w_{ij} \left\| T(x_i, y_j) - T(x_i, y_j)' \right\|. \tag{20}$$

Algorithm 1 The framework of the physics-informed convolutional neural networks for HSL-TFP

Input: Intensity distribution set $\boldsymbol{\Phi}$, Network parameters θ, Hyperparameter η_1,η_2.
Output: θ^*

```
Randomly initialize θ
while not done do
    Sample batch of intensity distribution Φ_N from Φ
    Predict the temperature field T_N with θ
    Process boundary of T_N with hard-constraint
    Calculate T_N' with Eq. (13) in no auto-grad mode
    Compute pixel-level weight w_ij with Eqs. (18) and (19)
    Compute the physics-informed loss ℒ with Eq. (20)
    Update θ using ℒ by auto-grad
end while
return optimized parameters θ*
```

For each sample, P-OHEM makes the optimizer more effort on the position with larger error, and the weight changes dynamically with the training. Benefiting from this clear idea, we expect the error of each position can be uniformly decreased. Finally, the training pseudocode of the physics-informed convolutional neural networks for HSL-TFP is given in Algorithm 1.

4. Experiment

4.1. Datasets and experimental settings

Datasets. Following previous works (Chen et al., 2017, 2020, 2021), two cases with varying degrees of complexity are studied to demonstrate the proposed method. As shown in Fig. 7, each data pairs in the dataset contain the heat source layout and corresponding steady-state temperature field. Heat sources are placed on the 0.1 m × 0.1 m square-shaped conduction domain for the layout data. One isothermal sink with a length of 0.01 m is located in the middle of the left boundary, and the other is adiabatic. The thermal conductivity λ of both domain and heat source equals 1 $W/(m \cdot K)$. For case 1, several square-shaped heat sources with the same setting are placed in the computation domain. There are 12 rectangular heat sources for the more complex case with various sizes and intensities. The configurations of heat source for two cases are respectively presented in Tables 1 and 2. We severally generate 10000 heat source layout maps for two cases using the layout sampling methods presented by Chen et al. (2020, 2021). Though the proposed physics-informed surrogate model is trained without annotated data, the referenced temperature field is produced by finite difference method (Reimer and Cheviakov, 2013) for results validation.

Table 2
The configuration of heat source in case 2.

Component	Size (m × m)	Intensity (w/m^2)
1	0.016 × 0.012	4000
2	0.012 × 0.006	16000
3	0.018 × 0.009	6000
4	0.018 × 0.012	8000
5	0.018 × 0.018	10000
6	0.012 × 0.012	14000
7	0.018 × 0.006	16000
8	0.009 × 0.009	20000
9	0.006 × 0.024	8000
10	0.006 × 0.012	16000
11	0.012 × 0.024	10000
12	0.024 × 0.024	20000

Table 3
Comparisons of FDM computations and PI-UNet predictions.

	Training set				Test set			
	MAE	CMAE	Max-AE	MT-AE	MAE	CMAE	Max-AE	MT-AE
Case 1	0.0098	0.0099	0.0235	0.0123	0.0108	0.0109	0.0280	0.0140
Case 2	0.0230	0.0228	0.0536	0.0277	0.0275	0.0271	0.0925	0.0336

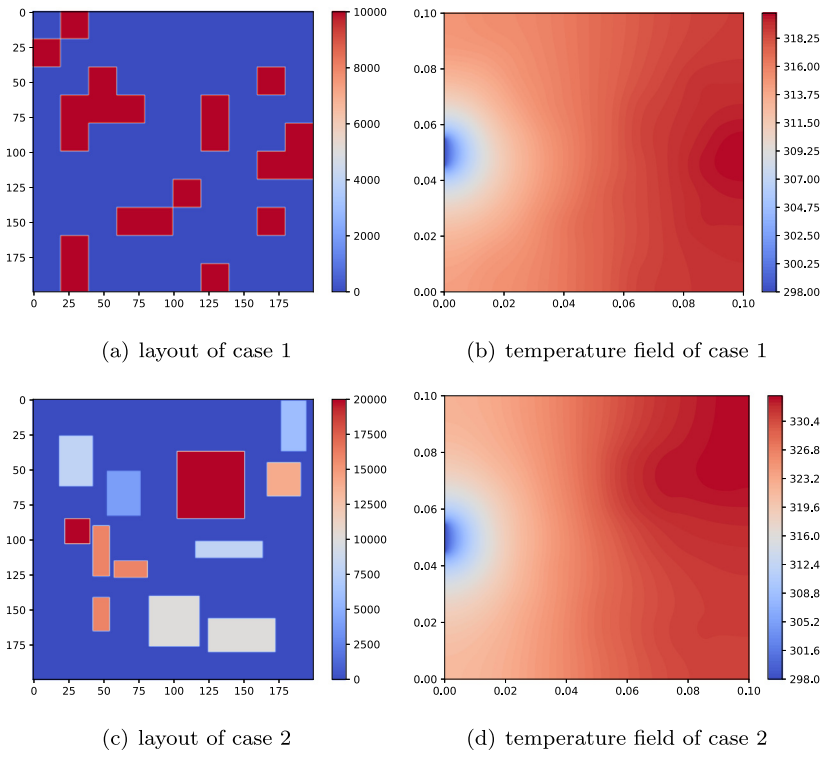
Evaluation Metrics. Chen et al. (2021) proposed a variety of metrics to evaluate the neural network performance for temperature field prediction. In this paper, we choose the following metrics, including the mean absolute error on the whole domain (MAE), the mean absolute error on the domain with components (CMAE), the maximum absolute error on the whole domain (Max-AE), and the absolute error of the maximum temperature (MT-AE). These metrics are closely related to thermal design and analysis. MAE and Max-AE metrics evaluate the overall and worst prediction accuracy. CMAE and MT-AE metrics are considered because some thermal design tasks concern the temperature around components, and the controlling of global maximum temperature (Chen et al., 2017).

Implementation Details. For each case, we split 10000 heat source layout maps into 8000, 1000, and 1000 for training, validation, and testing. The hyperparameters of η_1 and η_2 are set to 0 and 10. For training, we utilize Adam optimizer with the initial learning rate of 0.01, and the learning rate is adjusted by polynomial decay policy with multiplicative factor of 0.85. The batch size and the total epoch are set to 1 and 30. All models are trained on a single NVIDIA GeForce RTX 3090 GPU. The Pytorch implementation is released at https://github.com/zhaoxiaoyu1995/PI-UNet_HSL-TFP.

4.2. Prediction performance

Numerical method is sophisticated to compute the temperature field by solving PDEs. In this paper, the finite difference method (FDM), one of the traditional numerical methods, is chosen as the benchmark to verify the proposed method. The implementation of FDM is following (Reimer and Cheviakov, 2013).

Since the physics-informed loss guides the CNN training without labeled data, the label of training set is unseen for the network. Therefore, we evaluate the network performance both in the training and test sets. Our method is called as PI-UNet, and the results are presented in Table 3. First, the results show PI-UNet can make predictions with high accuracy, as the MAE with FDM computation is less than 0.03K in two cases. All evaluation metrics are larger in Case 2, meaning Case 2 is more difficult than Case 1. Second, the difference between



(a) layout of case 1 (b) temperature field of case 1

(c) layout of case 2 (d) temperature field of case 2

Fig. 7. The layout and temperature field of two cases. The rectangles in (a) and (c) represent the heat sources with different thermal intensities.

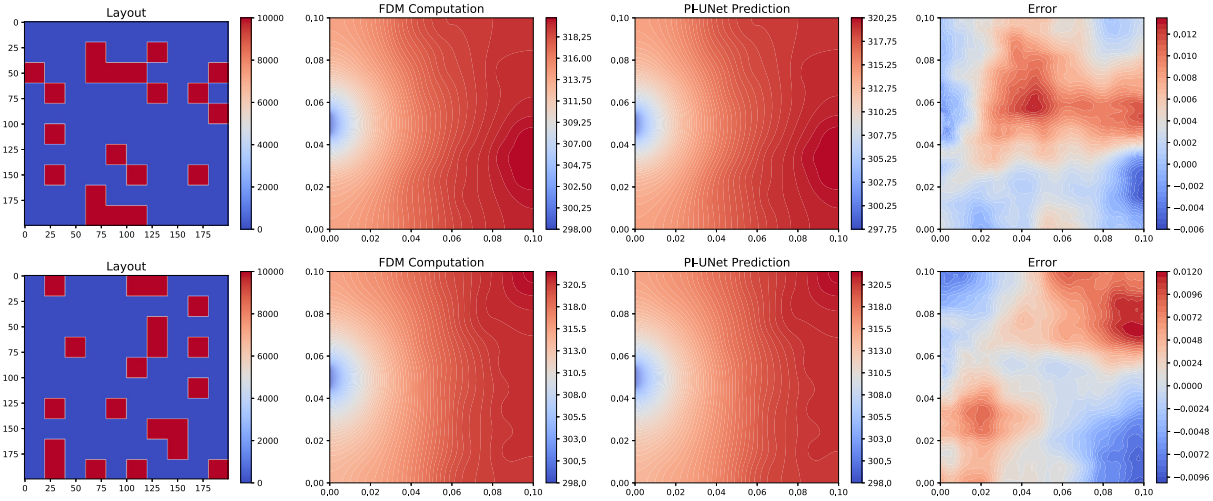


Fig. 8. Prediction examples of case 1. The 2nd and 3rd columns show the steady-state temperature field obtained by FDM and the proposed method, and the error between two methods on the computation domain is presented in last column.

various evaluation metrics is small. In particular, Max-AE evaluates the maximum absolute error, which is smaller than 0.1K. It means the error between prediction and FDM computation is uniform. It is essential because unusual errors occurring in the local region will damage the prediction reliability. Finally, the small gap of evaluation metrics in the training and test set indicates that the proposed physics-informed CNN is well generalized. Figs. 8 and 9 severally show the visual temperature field examples of prediction and FDM computation in case 1 and case 2. The network prediction is great close to the FDM computation. Since the prediction of the trained network is end-to-end and the forward propagation is very fast, the prediction by the trained network is superior in time consumption. In conclusion, the proposed method can guarantee rapid and high-accuracy prediction.

The Table 4 shows the results of existing DNN-based surrogate model for HSL-TFP. These methods require annotated data for training, and we use 8000 annotated data to train the network to present the data-driven results. In addition, the proposed label-free training method is performed on these surrogate models to obtain physics-informed results. Since our architecture is well-designed for physics-informed learning, the proposed method outperforms more remarkably than existing methods when trained without labeled data, and our method is even better than their data-driven results. It is worth noticing that the data-driven results of FPN are slightly better than our physics-informed method evaluated on MAE when the labeled data is sufficient, but the Max-AE results reverse.

Additionally, there are no extra parameters in the proposed physics-informed CNN in contrast with original CNN architecture driven by labeled data. For model complexity, we compare the params (FLOPs) of UNet used in our work and FPN (Chen et al., 2020) with Resnet50 backbone, and the results are 21.94M (23.67G) and 26.10M (4.97G). The

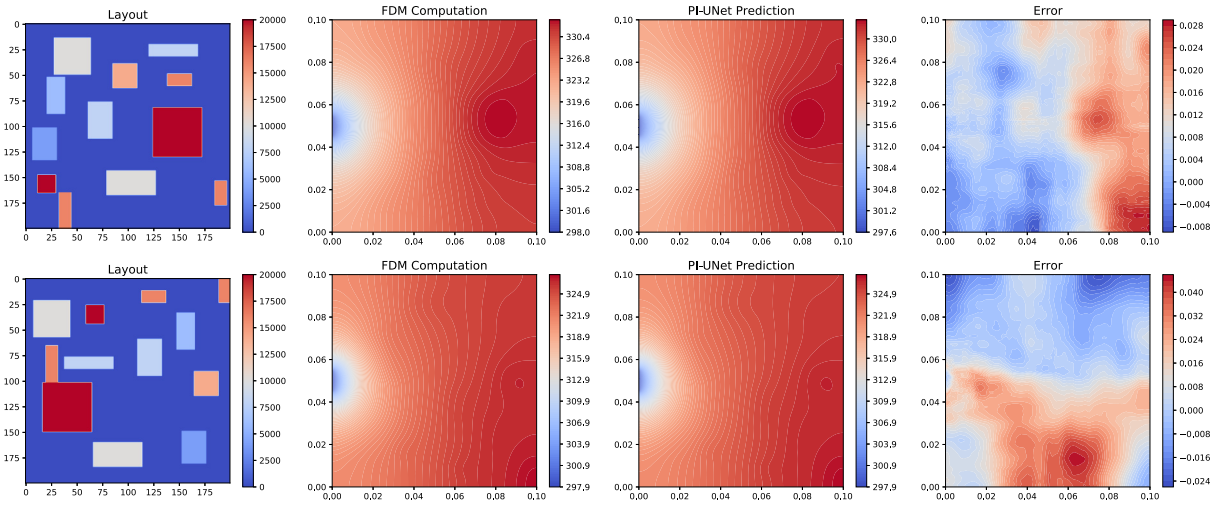


Fig. 9. Prediction examples of case 2. The 2nd and 3rd columns show the steady-state temperature field obtained by FDM and the proposed method, and the error between two methods on the computation domain is presented in last column.

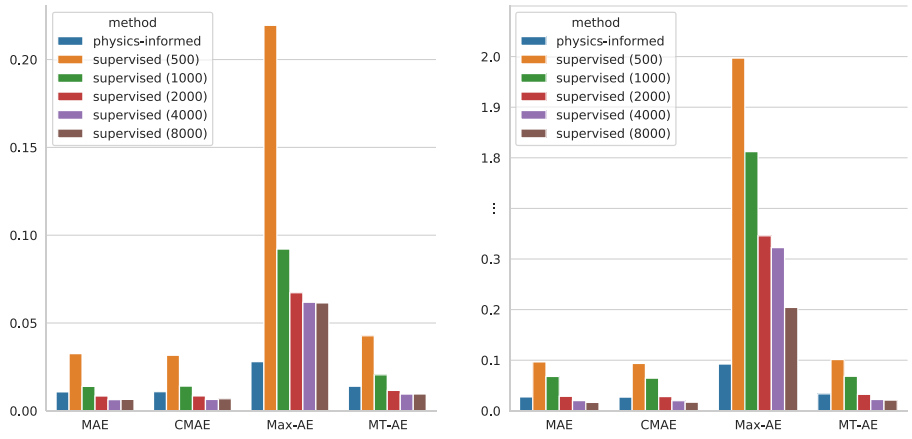


Fig. 10. Results of physics-informed learning and supervised learning with varying amounts of training data. The vertical axis displays absolute error of various evaluation metrics in test set.

Table 4
Comparisons of existing DNN-based method for HSL-TFP.

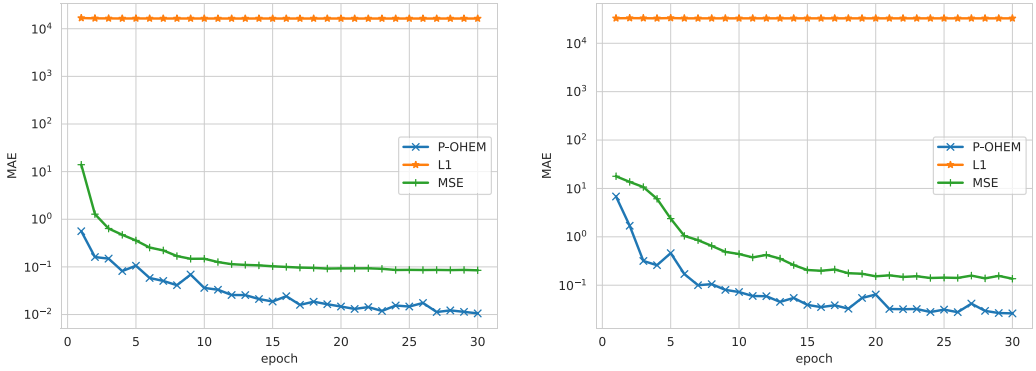
Method	Case 1				Case 2			
	Physics-informed		Data-driven (8000)		Physics-informed		Data-driven (8000)	
	MAE	Max-AE	MAE	Max-AE	MAE	Max-AE	MAE	Max-AE
FCN (Chen et al., 2021)	0.7473	2.3787	0.0271	3.3483	2.9683	6.7658	0.0476	5.0602
FPN (Chen et al., 2020)	0.1616	1.1640	0.0091	1.9150	0.1048	1.7160	0.0232	2.6780
Ours	0.0108	0.0280	0.0065	0.0615	0.0275	0.0925	0.0166	0.2044

selected architecture has fewer parameters but requires more floating point operations. We perform 1000 times forward propagation that the time of UNet and FPN are 4.65 s and 10.69 s. From the perspective of network architecture, the complexity of selected model maintains the same level as previous data-driven studies (Chen et al., 2020).

4.3. Comparisons with supervised learning

The proposed method employs physics knowledge to guide the network training. To investigate physics-informed learning and traditional supervised learning characteristics for HSL-TFP, we conduct supervised learning experiments with varying amounts of labeled data. The experimental setup is the same as physics-informed learning, except the physics-informed loss function is not applied. The number of labeled data is respectively set to 500, 1000, 2000, 4000, and 8000.

The results are compared in Fig. 10. From MAE, CMAE, and MT-AE, the accuracy of physics-informed learning is close to the supervised learning with 4000 labeled data. With the increase of labeled data, the model accuracy of supervised learning is higher, but there is no significant improvement. Supervised learning performs better when the labeled data is more than 4000. However, the gap between physics-informed learning and supervised learning is slight, and the accuracy of the proposed method meets the requirements of the application. For Max-AE, physics-informed learning is significantly lower than supervised learning. Max-AE evaluates the maximum absolute error with the FDM computation. It indicates that the supervised paradigm only learns to approximate the whole domain from labeled data, and the prediction



(a) case 1 (b) case 2

Fig. 11. The trend of validation MAE trained with P-OHEM, L1, and MSE loss.

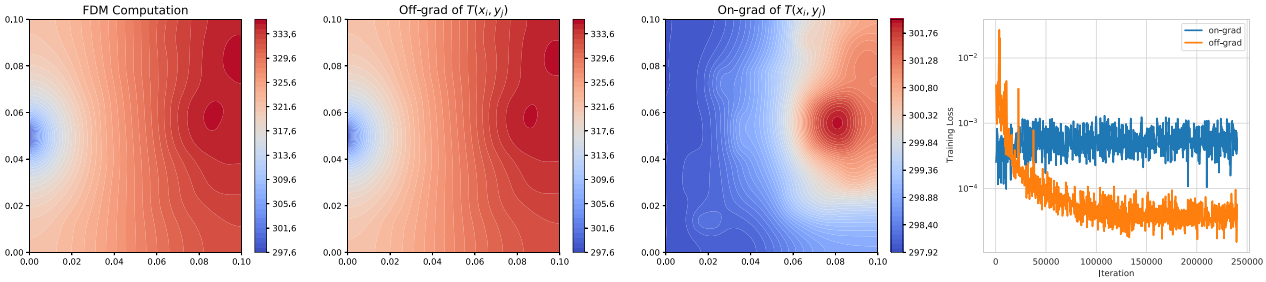


Fig. 12. The effect of grad computation on or off for $T(x_i, y_j)'$. The last column present the trend of training loss with turning on/off grad computation.

precision in the local region is weaker than the physics guidance. It is promising to conduct semi-supervised learning to combine the physics guidance and labeled data.

4.4. Ablation studies

We conduct a set of ablation experiments to study the effect of the loss function, training skills, and several network components.

Effect of P-OHEM. We perform ablation experiments to compare the performance of L1, MSE, and the proposed P-OHEM loss functions. The results of MAE evaluated in test data after each epoch can be seen in Fig. 11. When employing L1 loss, the network is unable to optimize properly, but this issue does not occur when using MSE and P-OHEM loss. Since the physics-informed loss is constructed based on PDEs and finite difference, numerical stability may be affected by the L1 loss. P-OHEM performs much better than MSE by properly processing different optimization difficulties in the computation domain. In particular, decreasing the loss in these domains is more challenging since the heat dissipated from the isothermal hole and temperature changes dramatically in the domain close to the hole. In P-OHEM, there is a larger weight for the position with a larger error, and more efforts will be made to decrease the physics-informed loss.

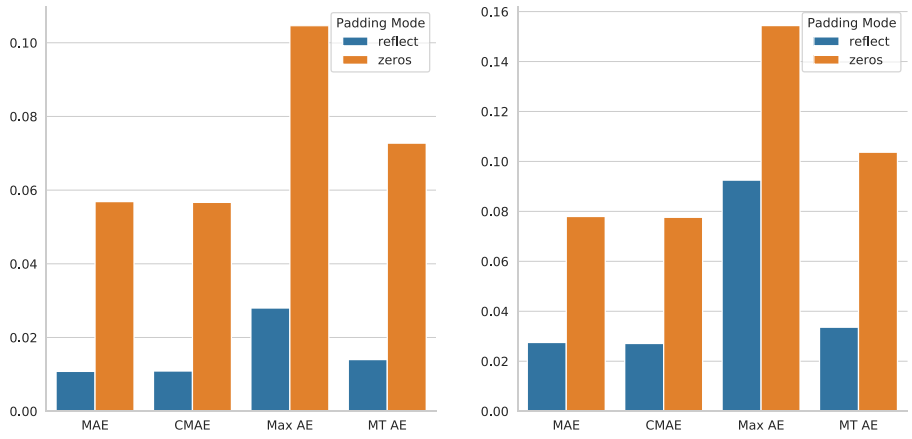
On–off of grad computation for $T(x_i, y_j)'$. In the physics-informed loss function Eq. (12), there is a intermediate variable $T(x_i, y_j)'$. The on–off of grad computation for $T(x_i, y_j)'$ has dramatically influenced the network optimization. The effect is shown in Fig. 12. When turning on the grad computation for $T(x_i, y_j)'$, the network is hard to optimize, and it tends to output zero values (the prediction values are started at 298K). The intermediate variable $T(x_i, y_j)'$ is constructed from surrounding predictions. When turning on the grad computation, the loss values in the current position will affect the adjacent position through back propagation, which may cause an abnormal trend of training loss.

Padding mode. In the convolution layer with 3×3 filter, the padding operator is adopted to make the input and output size unchanged. As shown in Fig. 13, 'reflect' padding mode performs much better than 'zeros' in all evaluation metrics. The distribution of loss values in two padding modes can be seen in Fig. 14. The loss values are larger in the boundary when using 'zeros' padding mode. Since the physics-informed loss is based on finite difference, the prediction error will diffuse from the boundary, reducing the prediction accuracy in the whole domain.

Batch Size. Fig. 15 shows the ablation study on the batch size. A larger batch size in computer vision usually benefits network training. However, the results are better when the batch size is set to smaller for the task at hand. In fixed training epochs, a smaller batch size means more iteration steps, contributing to the network convergence for HSL-TFP.

Hyperparameters. Hyperparameter η_2 determines the weight of each pixel in predicted temperature field, and larger η_2 means a greater gap of weight between pixels with maximum and minimum error. The results with different parameters η_2 are presented in Fig. 16. The proposed method is robust to the value of η_2, and it is appropriate to select hyperparameters η_2 as 10. With the increase of η_2, the optimizer makes more efforts to reduce the error in the larger loss values, but the η_2 should not be too large to influence global optimization.

Normalization methods, upsampling methods, and activation function. We conduct ablation experiments to analyze the choices of normalization, upsampling, and activation functions. As shown in Table 5, the results of GN and IN are much better than BN, and GN is slightly better than IN. There is a great improvement in BN when turning on the training mode in evaluation. It means the statistics of BN are not incorrect in training. However, the results of BN are not notably improved with increasing the batch size to 16. For two cases, bilinear upsampling yields significantly better performance than transpose convolution. There is no clear difference in the performance of the three activation functions, but GELU is a little better. The experiments demonstrate that the combination of GN, bilinear upsampling, and GELU is most effective.



(a) case 1 (b) case 2

Fig. 13. Comparisons of 'reflect' and 'zeros' padding mode.

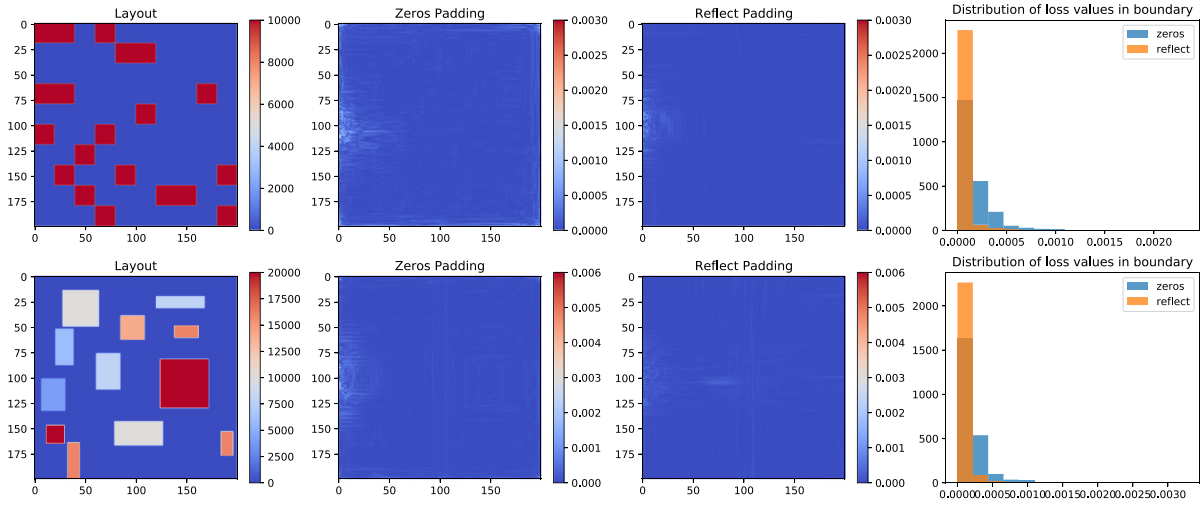
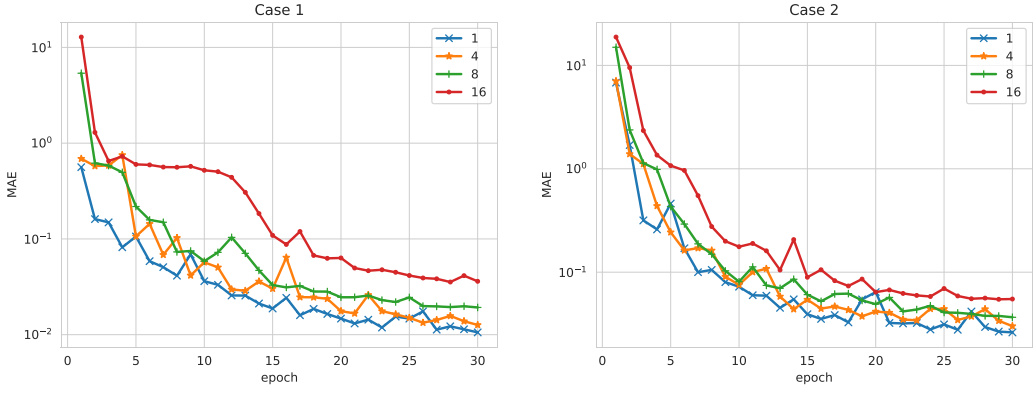


Fig. 14. The distribution of physics-informed loss in 'reflect' and 'zeros' mode.

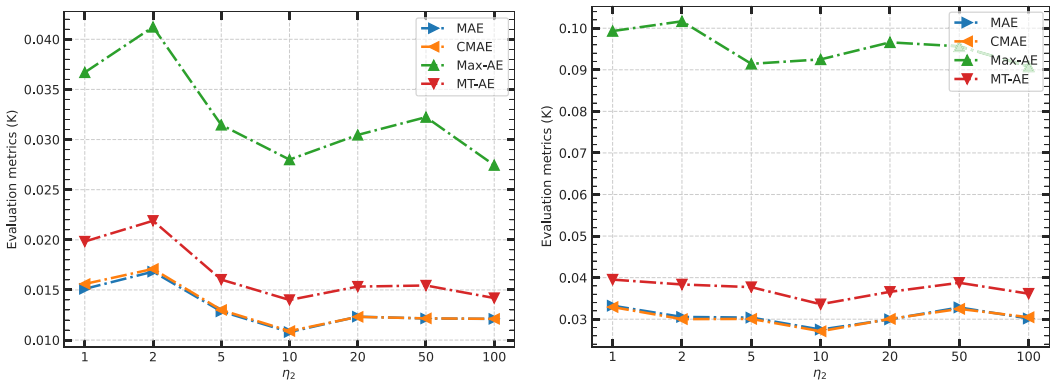


(a) case 1 (b) case 2

Fig. 15. The trend of validation MAE trained with different batch sizes. MAE is computed between the proposed method and FDM.

5. Conclusion

This paper investigates the physics-informed convolutional neural networks for temperature field prediction of heat source layout. First, we construct a physics-informed loss inspired by physics equations and the finite difference method. The loss guides the network to learn an operator mapping from intensity distribution function to solution function without simulation data. Furthermore, we study to impose hard constrain on the Dirichlet and Neumann boundary to speed the network convergence. Finally, we develop a well-designed architecture



(a) case 1 (b) case 2

Fig. 16. Effect of hyperparameter η_2 on the results of two cases.

Table 5
Comparisons of various network components.

	Normalization	Upsampling	Activation function	Batch size	Train MAE	Test MAE
Case 1	GN	Bilinear	GELU	1	0.0098	0.0108
	GN	Bilinear	ReLU	1	0.0170	0.0191
	GN	Bilinear	Tanh	1	0.0198	0.0214
	BN	Bilinear	GELU	1	0.5623	0.5824
	BN	Bilinear	GELU	16	0.4701	0.4943
	IN	Bilinear	GELU	1	0.0184	0.0197
	GN	Transpose	GELU	1	0.0409	0.0421
Case 2	GN	Bilinear	GELU	1	0.0230	0.0275
	GN	Bilinear	ReLU	1	0.0464	0.0515
	GN	Bilinear	Tanh	1	0.0435	0.0491
	BN	Bilinear	GELU	1	1.1383	1.1736
	BN	Bilinear	GELU	16	0.8775	0.9358
	IN	Bilinear	GELU	1	0.0340	0.0396
	GN	Transpose	GELU	1	0.0964	0.0978

based on UNet for the HSL-TFP task, and pixel-level online hard example mining is proposed to balance different optimization difficulties in the computation domain. For two HSL-TFP tasks, the experiments demonstrate that the proposed method can make accurate temperature field predictions comparable to the numerical method and data-driven models. The trained network can directly output the solution of HSL-TFP without solving PDE. It can be employed as a surrogate model to assist the layout optimization benefit from rapid and high-precision prediction.

Similar to steady-state temperature field prediction, the developed methods can be generalized to other physics problems, employing networks to solve parameterized PDEs, such as elastic mechanics, fluid mechanics, and electromagnetism. Furthermore, it is promising to investigate the methods in irregular computation domain and large-scale mesh. In some applications, the data-driven method is more effective. Combining physics knowledge and labeled data is valuable to perform model verification, semi-supervised learning, and transfer learning.

CRediT authorship contribution statement

Xiaoyu Zhao: Writing – original draft, Methodology, Implementation, Investigation, Data generation. **Zhiqiang Gong:** Writing – review & editing, Methodology, Investigation. **Yunyang Zhang:** Writing – review & editing, Methodology, Visualization. **Wen Yao:** Writing – review & editing, Methodology, Project administration, Supervision. **Xiaoqian Chen:** Writing – review & editing, Methodology, Project administration, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the Postgraduate Scientific Research Innovation Project of Hunan Province (No. CX20200006) and the National Natural Science Foundation of China (No. 11725211 and 52005505).

References

Anandkumar, A., Azizzadenesheli, K., Bhattacharya, K., Kovachki, N., Li, Z., Liu, B., et al., 2020. Neural operator: Graph kernel network for partial differential equations. In: ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations.

Aslan, Y., Puskely, J., Yarovoy, A., 2018. Heat source layout optimization for two-dimensional heat conduction using iterative reweighted l1-norm convex minimization. Int. J. Heat Mass Transfer 122, 432–441. http://dx.doi.org/10.1016/j.ijheatmasstransfer.2018.02.001.

Ates, G.C., Gorguluarslan, R.M., 2021. Two-stage convolutional encoder–decoder network to improve the performance and reliability of deep learning models for topology optimization. Struct. Multidiscip. Optim. 63 (4), 1927–1950.

Baydin, A.G., Pearlmutter, B.A., Radul, A.A., Siskind, J.M., 2018. Automatic differentiation in machine learning: A survey. J. Mach. Learn. Res. 18.

Bhattacharya, K., Hosseini, B., Kovachki, N.B., Stuart, A.M., 2021. Model reduction and neural networks for parametric PDEs. SMAI J. Comput. Math. 7, 121–157.

Chen, X., Chen, X., Zhou, W., Zhang, J., Yao, W., 2020. The heat source layout optimization using deep learning surrogate modeling. Struct. Multidiscip. Optim. 62 (6), 3127–3148.

Chen, K., Wang, S., Song, M., 2016. Optimization of heat source distribution for two-dimensional heat conduction using bionic method. Int. J. Heat Mass Transfer 93, 108–117. http://dx.doi.org/10.1016/j.ijheatmasstransfer.2015.09.041.

Chen, K., Xing, J., Wang, S., Song, M., 2017. Heat source layout optimization in two-dimensional heat conduction using simulated annealing method. Int. J. Heat Mass Transfer 108, 210–219. http://dx.doi.org/10.1016/j.ijheatmasstransfer.2016.12.007.

Chen, X., Zhao, X., Gong, Z., Zhang, J., Zhou, W., Chen, X., et al., 2021. A deep neural network surrogate modeling benchmark for temperature field prediction of heat source layout. arXiv preprint arXiv:210311177.

Deng, W., Li, Z., Li, X., Chen, H., Zhao, H., 2022a. Compound fault diagnosis using optimized mckd and sparse representation for rolling bearings. IEEE Trans. Instrum. Meas. 71, 1–9.

Deng, W., Zhang, X., Zhou, Y., Liu, Y., Zhou, X., Chen, H., et al., 2022b. An enhanced fast non-dominated solution sorting genetic algorithm for multi-objective problems. Inform. Sci. 585, 441–453.

Gao, H., Sun, L., Wang, J.X., 2021. PhyGeoNet: Physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state PDEs on irregular domain. J. Comput. Phys. 428, 110079.

Kallioras, N.A., Kazakis, G., Lagaros, N.D., 2020. Accelerated topology optimization by means of deep learning. Struct. Multidiscip. Optim. 62 (3), 1185–1212.

Kim, B., Azevedo, V.C., Thuerey, N., Kim, T., Gross, M., Solenthaler, B., 2019. Deep fluids: A generative network for parameterized fluid simulations. In: Computer Graphics Forum, vol. 38, Wiley Online Library, pp. 59–70.

Lee, T., Bilionis, I., Tepole, A.B., 2020. Propagation of uncertainty in the mechanical and biological response of growing tissues using multi-fidelity gaussian process regression. Comput. Methods Appl. Mech. Engrg. 359, 112724.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., et al., 2020a. Fourier neural operator for parametric partial differential equations. arXiv preprint arXiv:201008895.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., et al., 2020b. Multipole graph neural operator for parametric partial differential equations. arXiv preprint arXiv:200609535.

Liu, D., Wang, Y., 2019. Multi-fidelity physics-constrained neural network and its application in materials modeling. J. Mech. Des. 141 (12).

Liu, X., Wu, Y., Wang, B., Ding, J., Jie, H., 2017. An adaptive local range sampling method for reliability-based design optimization using support vector machine and kriging model. Struct. Multidiscip. Optim. 55 (6), 2285–2304.

Long, Z., Lu, Y., Dong, B., 2019. PDE-Net 2.0: Learning PDEs from data with a numeric-symbolic hybrid deep network. J. Comput. Phys. 399, 108925.

Long, Z., Lu, Y., Ma, X., Dong, B., 2018. PDE-Net: Learning PDEs from data. In: International Conference on Machine Learning. PMLR, pp. 3208–3216.

Lu, L., Jin, P., Pang, G., Zhang, Z., Karniadakis, G.E., 2021. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. Nat. Mach. Intell. 3 (3), 218–229.

Mao, Z., Jagtap, A.D., Karniadakis, G.E., 2020. Physics-informed neural networks for high-speed flows. Comput. Methods Appl. Mech. Engrg. 360, 112789.

Misiulia, D., Elsayed, K., Andersson, A.G., 2017. Geometry optimization of a deswirler for cyclone separator in terms of pressure drop using cfd and artificial neural network. Sep. Purif. Technol. 185, 10–23.

Misyris, G.S., Venzke, A., Chatzivasileiadis, S., 2020. Physics-informed neural networks for power systems. In: 2020 IEEE Power & Energy Society General Meeting. PESGM, IEEE, pp. 1–5.

Moustapha, M., Bourinet, J.M., Guillaume, B., Sudret, B., 2018. Comparative study of kriging and support vector regression for structural engineering applications. ASCE-ASME J. Risk Uncertain. Eng. Syst. A 4 (2), 04018005.

Patel, R.G., Trask, N.A., Wood, M.A., Cyr, E.C., 2021. A physics-informed operator regression framework for extracting data-driven continuum models. Comput. Methods Appl. Mech. Engrg. 373, 113500.

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

Reimer, A.S., Cheviakov, A.F., 2013. A matlab-based finite-difference solver for the Poisson problem with mixed dirichlet–neumann boundary conditions. Comput. Phys. Comm. 184 (3), 783–798.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 234–241.

Sun, L., Gao, H., Pan, S., Wang, J.X., 2020. Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data. Comput. Methods Appl. Mech. Engrg. 361, 112732.

Teichert, G.H., Garikipati, K., 2019. Machine learning materials physics: Surrogate optimization and multi-fidelity algorithms predict precipitate morphology in an alternative to phase field dynamics. Comput. Methods Appl. Mech. Engrg. 344, 666–693.

Wang, N., Zhang, D., Chang, H., Li, H., 2020. Deep learning of subsurface flow via theory-guided neural network. J. Hydrol. 584, 124700.

Wu, Y., He, K., 2018. Group normalization. In: Proceedings of the European Conference on Computer Vision. ECCV, pp. 3–19.

Yao, H., Gao, Y., Liu, Y., 2020. FEA-Net: A physics-guided data-driven model for efficient mechanical response prediction. Comput. Methods Appl. Mech. Engrg. 363, 112892.

Zhu, Y., Zabaras, N., Koutsourelakis, P.S., Perdikaris, P., 2019. Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. J. Comput. Phys. 394, 56–81.

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