

A graph neural network approach to the inverse design for thermal transparency with periodic interparticle system

Bin Liu(刘斌) and Yixi Wang(王译浠)

Citation: Chin. Phys. B, 2024, 33 (**8**): 084401. DOI: 10.1088/1674-1056/ad4326

What follows is a list of articles you may be interested in

General three-dimensional thermal illusion metamaterials

Tianfeng Liu(刘天丰), Zhaochen Wang(王兆宸), Zhan Zhu(朱展), and Run Hu(胡润)
Chin. Phys. B, 2024, 33 (**4**): 044401. DOI: 10.1088/1674-1056/ad09aa

Extraction method of nanoparticles concentration distribution from magnetic particle image and its application in thermal damage of magnetic hyperthermia

Yundong Tang(汤云东), Ming Chen(陈鸣), Rodolfo C.C. Flesch, and Tao Jin(金涛)
Chin. Phys. B, 2023, 32 (**9**): 094401. DOI: 10.1088/1674-1056/acde50

Thermal transport properties of two-dimensional boron dichalcogenides from a first-principles and machine learning approach

Zhanjun Qiu(邱占均), Yanxiao Hu(胡晏箫), Ding Li(李顶), Tao Hu(胡涛), Hong Xiao(肖红),Chunbao Feng(冯春宝), and Dengfeng Li(李登峰)
Chin. Phys. B, 2023, 32 (**5**): 054402. DOI: 10.1088/1674-1056/acb9e6

Accurate prediction of the critical heat flux for pool boiling on the heater substrate

Fengxun Hai(海丰勋), Wei Zhu(祝薇), Xiaoyi Yang(杨晓奕), and Yuan Deng(邓元)
Chin. Phys. B, 2022, 31 (**6**): 064401. DOI: 10.1088/1674-1056/ac523c

Two-dimensional thermal illusion device with arbitrary shape based on complementary media

Ge Xia(夏舸), Wei Kou(寇蔚), Li Yang(杨立), Yong-Cheng Du(杜永成)
Chin. Phys. B, 2017, 26 (**10**): 104403. DOI: 10.1088/1674-1056/26/10/104403

A graph neural network approach to the inverse design for thermal transparency with periodic interparticle system

Bin Liu(刘斌)[1,†] and Yixi Wang(王译浠)[2]

[1] *Department of Electronic Information and Artificial Intelligence, LeShan Normal University, LeShan* 614099, *China*
[2] *Department of Physics, State Key Laboratory of Surface Physics, and Key Laboratory of Micro and Nano Photonic Structures (MOE), Fudan University, Shanghai* 200438, *China*

(Received 23 February 2024; revised manuscript received 8 April 2024; accepted manuscript online 25 April 2024)

Recent years have witnessed significant advances in utilizing machine learning-based techniques for thermal metamaterial-based structures and devices to attain favorable thermal transport behaviors. Among the various thermal transport behaviors, achieving thermal transparency stands out as particularly desirable and intriguing. Our earlier work demonstrated the use of a thermal metamaterial-based periodic interparticle system as the underlying structure for manipulating thermal transport behavior and achieving thermal transparency. In this paper, we introduce an approach based on graph neural network to address the complex inverse design problem of determining the design parameters for a thermal metamaterial-based periodic interparticle system with the desired thermal transport behavior. Our work demonstrates that combining graph neural network modeling and inference is an effective approach for solving inverse design problems associated with attaining desirable thermal transport behaviors using thermal metamaterials.

Keywords: thermal metamaterial, thermal transparency, inverse design, machine learning, graph neural network

PACS: 44.05.+e, 44.90.+c, 07.05.Mh, 65.90.+i **DOI:** 10.1088/1674-1056/ad4326

1. Introduction

The emergence of thermal metamaterials, coupled with their ongoing evolution and application, has fundamentally transformed the landscape of research in the manipulation of thermal transport behavior.[1–9] This evolution has given rise to a myriad of intriguing and practical applications, including thermal cloaks,[10–24] thermal concentrators,[14,25,26] thermal camouflage,[27–35] and thermal transparency.[36–46]

In recent years, the exploration of thermal metamaterials and their applications has experienced a considerable boost thanks to the introduction of machine learning, deep learning, and topology optimization algorithms.[47–55] The "learning from data" paradigm inherent in machine-learning approaches has exhibited unparalleled efficacy in solving inverse design problems for structures and devices based on thermal metamaterials.[54,55]

Among the diverse applications of thermal metamaterial-based structures and devices, thermal transparency stands out as particularly intriguing, offering practical implications for industrial or military use. Thermal transparency is achieved when a device based on thermal metamaterials is designed in a manner that renders its internal composite material or structure capable of generating an effective thermal conductivity that is indistinguishable from the surrounding background. The efficacy of thermal transparency can be evaluated experimentally by utilizing an infrared camera to capture images depicting temperature distribution, as demonstrated in our prior research.[56]

A periodic interparticle system (PIS) as the underlying structure to achieve thermal transparency was first proposed by Xu *et al.*[56] Then, in our previous work,[54] the constraints of a square lattice and the requirement for particles to be alternately positioned along both the horizontal and vertical directions are dropped to allow more degrees of freedom, i.e., more design parameters in a PIS. Xu *et al*. initially introduced the concept of employing a PIS as the underlying structure for achieving thermal transparency.[56] In our earlier research,[54] we relaxed the constraints of a square lattice and the stipulation that particles must be alternately positioned along both the horizontal and vertical directions. This modification allows for greater degrees of freedom, resulting in an increased number of design parameters within a PIS. However, with these constraints removed, a closed-form solution to find the design parameters of a PIS no longer exists. This challenge compels us to resort to a machine learning-based approach. In this work, we aim to tackle the challenging inverse design problems associated with the design parameters of PIS by using a graph neural network (GNN) based approach. We show in this paper that a simple node- and edge-labeled undirected graph can effectively model a PIS and graph convolution has the capability to aggregate the node- and edge-level features into a graph level feature, accurately modeling the thermal transport behavior of the PIS. This enables us to develop a GNN model that is capable of predicting the PIS's thermal transport behavior based on the corresponding design parameters. Additionally,

[†]Corresponding author. E-mail: liubin@fudan.edu.cn

© 2024 Chinese Physical Society and IOP Publishing Ltd

our model can make predictions in reverse order, i.e., given the desired thermal transport behavior, it can predict a suitable set of design parameters.

2. Graph neural network as an approach to the inverse design problem for manipulating thermal transport behavior

GNNs refer to a class of artificial neural network-based models that operate on graph data. To be technically precise, we refrain from categorizing GNNs as a class of deep learning models since, in most cases, a GNN model consists of just a few layers.[57] Beyond this, two fundamental issues arise: first, prohibitive computation cost due to neighborhood explosion; second, oversmoothing leading to an inability to capture long-range dependencies in the graph data, resulting in deteriorated expressivity.[57]

Since their debut in 2009,[58] the exploration and applications in GNNs have burgeoned into one of the most productive fields of AI research, with numerous applications in various fields, including traffic forecasting,[59] text classification,[60] and traffic forecasting.[61] The applications of GNNs in drug discovery and design are advancing rapidly, extending from the discovery of novel antibiotic[62] to predicting emerging drug interactions.[63,64] A survey of geometric GNNs was recently published to give a panoramic view of their data structures, models and applications.[65]

Graphs serve as a method to describe and capture relationships within data, constituting a form of data structure that is both nonlinear and abstract. Given their abstract nature, graphs necessitate a concrete representation to be effectively operated upon. Additionally, graphs can be defined to have certain properties that may necessitate different representations. A graph usually consists of two kinds of elements: nodes and connections. According to how the nodes are ordered and connected, graphs can be divided into two large categories: directed graphs and undirected graphs. In the former, nodes are connected by directed edges, while in the latter the order of nodes and their connection is irrelevant. Given that our paper utilizes an undirected graph to model the PIS for achieving thermal transparency, we will specifically discuss the subtypes of undirected graphs only.

There are generally three subtypes of undirected graphs: node-labeled undirected graph, edge-labeled undirected graph, and node- and edge-labeled undirected graph. In a node-labeled undirected graph, only nodes have features and edges are featureless. A concrete example of a node-labeled undirected graph is a social network where each node represents a person and has a label indicating the person's occupation. In an edge-labeled undirected graph, only edges have features and nodes are featureless. A concrete example of an edge-labeled undirected graph is a transportation network where each edge represents a road and has a label indicating the speed limit.

In a node- and edge-labeled undirected graph, both nodes and edges have features. The best example for a node- and edge-labeled undirected graph is probably a molecule in which each node is a chemical atom (for example, carbon, hydrogen or nitrogen atoms), and each edge represents the type of bond (for example, single or double bond) between two nodes.

The PIS, proposed as the underlying structure for achieving thermal transparency, was initially introduced by Xu *et al.*[56] and further developed by Liu *et al.*[54] We can create a sample of a PIS utilizing two naturally occurring substances: copper and air. By employing the effective medium theory to identify the optimal ratio, these materials can collectively produce a diverse spectrum of effective thermal conductivity. The nature of the PIS mandates that it can be adeptly modeled by a node- and edge-labeled undirected graph because the 10 design parameters that fully characterize a PIS can be attributed as either node features or edge features. From now on, we use node A to denote the whole set of particles A, node B to denote the whole set of particles B, and edge to denote the relative positioning between particles A and B, along with other system-wide controllable design parameters. To illustrate, we replicate the schematic description of the PIS from our prior research (Fig. 1).

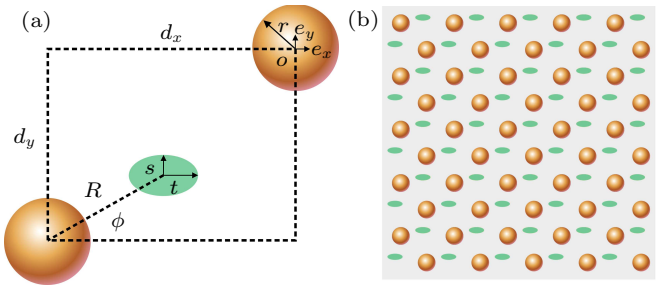


Fig. 1. Periodic interparticle system, adapted from Ref. [54]. (a) The configuration of a unit cell in the PIS is characterized by the parametrization of the relative positioning between two particles, A and B. This parametrization involves the distance R between their centers and the angular position ϕ relative to the x axis. Particle A exhibits anisotropic thermal conductivities, which are parameterized in cylindrical coordinates. On the other hand, particle B possesses uniform thermal conductivities, and its shape is defined by a semi-minor axis s and a semi-major axis t. The ranges for these design parameters used in this work are listed in Table 1. (b) Illustration of the periodic structure discussed in this paper involves particles A (represented as circles) and particles B (represented as ellipses) positioned within their respective rectangular lattices.

In a PIS, two particle types are systematically arranged on a periodic lattice with respect to the background. Particles A are characterized by a circular shape and anisotropic thermal conductivities, while particles B are characterized by an elliptical shape and isotropic thermal conductivities. Both types of particles are fixed on a horizontally and vertically alternating rectangular lattice, with the distance between two horizontally adjacent particles of the same type denoted as $2d_x$, and the distance between two vertically adjacent particles of the same

type denoted as $2d_y$. The value of d_x is not controllable and fixed at 1 cm. The relative positioning between pairing circles (particles A) and ellipses (particles B) is framed into a cylindrical coordinate R and ϕ, with the origin located at the center of the circles (see Fig. 4(f) of Ref. [54]). More specifically, R denotes the distance between the centers of pairing circle and ellipse, while ϕ denotes the angle between the line connecting the centers of pairing circle and ellipse and the horizontal axis. The background is only characterized by its thermal conductivity κ_m. Now, we designate each of the 10 design parameters as either node A, node B, or edge features.

The area fraction of particles A, p_a $\left[=\pi r^2/\left(2d_xd_y\right)\right]$ and its anisotropic thermal conductivities $\overleftrightarrow{\kappa}_a = \mathrm{diag}\left(\kappa_{\rho\rho}, \kappa_{\theta\theta}\right)$ are node A features. Here, $\overleftrightarrow{\kappa}_a$ is represented in cylindrical coordinates (ρ, θ), with the origin at the center of a particle A.

The area fraction of particle B, p_b $[=\pi st/\left(2d_xd_y\right)]$, its uniform thermal conductivity κ_b, and the aspect ratio e denoting the ratio of the semi-minor axis s to semi-major axis t, are node B features. Finally, we can assign all the remaining design parameters, i.e., R, ϕ, d_y and κ_m, as edge features.

Then, a node- and edge-labeled undirected graph composed of node A, node B, and edge can fully parameterize a PIS. From the viewpoint of GNN, the thermal transport characteristics of a PIS can be regarded as a graph level feature $\boldsymbol{y}$.

Our approach to achieving thermal transparency through the inverse design for a PIS is inspired by the work of Dold *et al*. on utilizing a GNN model for the inverse design of lattice materials.[66] The entire procedure for performing the inverse design for a PIS to achieve thermal transparency consists of two stages.

In the first stage, we train a GNN model capable of predicting the thermal transport characteristics as a graph-level feature from the features of node A, node B, and edge. In the second stage, with the weights of our GNN model frozen, node A, node B, and edge features $\boldsymbol{x}_{\mathrm{NodeA}}$, $\boldsymbol{x}_{\mathrm{NodeB}}$, $\boldsymbol{x}_{\mathrm{edge}}$ are first randomly initialized and then optimized so that our trained and frozen GNN model can output $\boldsymbol{y}_{\mathrm{prediction}}$ as close to the ground truth $\boldsymbol{y}$ as possible.

To accomplish the entire inverse design procedure, we first need to effectively depict and quantify the thermal transport characteristics of a PIS. A baseline situated at $x=-6$ cm, relative to the simulation box's origin, is defined for this purpose. The baseline is then discretized into 400 evenly spaced points, on which heat flux values are measured and concatenated into a 400-dimensional response vector. Our previous work[54] indicates that the 400-dimensional response vector has significant information redundancy, and an autoencoder is capable of removing this redundancy by reducing the dimensionality of the response vector with little loss of information. In this work, instead of using original 400-dimensional response vector, a 10-dimensional reduced response vector generated by compressing the original response vector using a trained autoencoder is used to represent the thermal transport characteristics of a PIS. Thus, the graph level feature $\boldsymbol{y}$ is a 10-dimensional vector.

Our GNN model is capable of capturing the relationship between the features of node A, node B, and edge, as well as the graph-level feature. It is built on the idea of graph convolutions.[67] A graph convolutional network (GCN) model is an approach for semi-supervised learning and a flavor of GNN that can leverage the graph structure and aggregate node and edge information from the neighborhoods in a convolutional fashion. Strictly speaking, our GNN model does not qualify as a conventional GCN model because a conventional GCN model involves graph convolution operations on nodes only.[67] We only borrow the underlying idea of message passing for GCNs to build our GNN model. As illustrated in the second panel of Fig. 2, our GNN model is constructed upon two forward passes of graph convolutions followed by a concatenation pooling layer and two fully-connected layers.

The essence of graph convolution can be understood as message passing. During the first forward pass of graph convolution (see details in Fig. 2(b) and Eqs. (1)–(3)), the original node A, node B, and edge features $\boldsymbol{x}^{(0)}_{\mathrm{NodeA}}$, $\boldsymbol{x}^{(0)}_{\mathrm{NodeB}}$, $\boldsymbol{x}^{(0)}_{\mathrm{edge}}$ are first multiplied by respective weight matrices and summed with a bias term $\boldsymbol{b}$ to obtain the updated node A, node B, and edge features $\boldsymbol{x}^{(1)}_{\mathrm{NodeA}}$, $\boldsymbol{x}^{(1)}_{\mathrm{NodeB}}$, $\boldsymbol{x}^{(1)}_{\mathrm{edge}}$. Here, each of the $\boldsymbol{W}$ matrices of learnable filter weights has the shape of $f_{\mathrm{in}} \times f_{\mathrm{out}}$, where f_{in} stands for the dimension of the feature that is being multiplied by the respective matrix, and f_{out} is the dimension of the feature on the left-hand side of the equation. $\boldsymbol{b}$ is a learnable bias vector of length f_{out}.

$$\begin{aligned}\boldsymbol{x}^{(1)}_{\mathrm{NodeA}} &= \boldsymbol{x}^{(0)}_{\mathrm{NodeA}}\boldsymbol{W}_{\mathrm{NodeA,NodeA}} + \boldsymbol{x}^{(0)}_{\mathrm{NodeB}}\boldsymbol{W}_{\mathrm{NodeB,NodeA}} \\ &\quad + \boldsymbol{x}^{(0)}_{\mathrm{edge}}\boldsymbol{W}_{\mathrm{edge,NodeA}} + \boldsymbol{b}_{\mathrm{NodeA}}, \qquad (1)\\ \boldsymbol{x}^{(1)}_{\mathrm{NodeB}} &= \boldsymbol{x}^{(0)}_{\mathrm{NodeA}}\boldsymbol{W}_{\mathrm{NodeA,NodeB}} + \boldsymbol{x}^{(0)}_{\mathrm{NodeB}}\boldsymbol{W}_{\mathrm{NodeB,NodeB}} \\ &\quad + \boldsymbol{x}^{(0)}_{\mathrm{edge}}\boldsymbol{W}_{\mathrm{edge,NodeB}} + \boldsymbol{b}_{\mathrm{NodeB}}, \qquad (2)\\ \boldsymbol{x}^{(1)}_{\mathrm{edge}} &= \boldsymbol{x}^{(0)}_{\mathrm{NodeA}}\boldsymbol{W}_{\mathrm{NodeA,edge}} + \boldsymbol{x}^{(0)}_{\mathrm{NodeB}}\boldsymbol{W}_{\mathrm{NodeB,edge}} \\ &\quad + \boldsymbol{x}^{(0)}_{\mathrm{edge}}\boldsymbol{W}_{\mathrm{edge,edge}} + \boldsymbol{b}_{\mathrm{edge}}. \qquad (3)\end{aligned}$$

In this way, the messages (feature information) from node A, node B, and edge are delivered to themselves and the other components of the graph. This means that after the first forward pass, the updated node A, node B, and edge features $\boldsymbol{x}^{(1)}_{\mathrm{NodeA}}$, $\boldsymbol{x}^{(1)}_{\mathrm{NodeB}}$, $\boldsymbol{x}^{(1)}_{\mathrm{edge}}$ do not only contain information about themselves but also incorporate information about the whole graph.

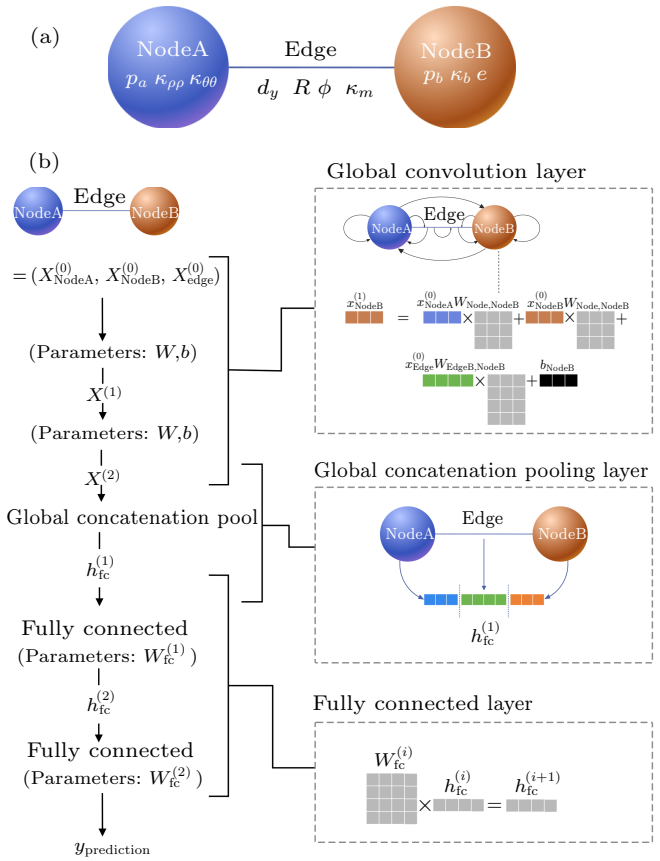


Fig. 2. (a) Schematic diagram of the graph representing a PIS. Node A has features p_a, $\kappa_{\rho\rho}$ and $\kappa_{\theta\theta}$. Node B has features p_b, κ_b and e. Edge has features d_y, R, ϕ and κ_m. (b) A visualization of the layers in the GNN model. Two graph convolutions are performed first to obtain the updated node A, node B, and edge features $\boldsymbol{x}^{(1)}_{\text{NodeA}}$, $\boldsymbol{x}^{(1)}_{\text{NodeB}}$, $\boldsymbol{x}^{(1)}_{\text{edge}}$ in the first pass, and $\boldsymbol{x}^{(2)'}_{\text{NodeA}}$, $\boldsymbol{x}^{(2)'}_{\text{NodeB}}$, $\boldsymbol{x}^{(2)'}_{\text{edge}}$ in the second pass. A global concatenation pooling layer joins $\boldsymbol{x}^{(2)'}_{\text{NodeA}}$, $\boldsymbol{x}^{(2)'}_{\text{NodeB}}$, $\boldsymbol{x}^{(2)'}_{\text{edge}}$ into a 10-dimensional vector, which is fed into two fully connected layers to output $y_{\text{prediction}}$.

After the first forward pass of graph convolution, immediately followed by an ELU activation layer to introduce nonlinearity to the GNN, the second forward pass is conducted (see details in Fig. 2(b) and Eqs. (4)–(6)). The second forward pass allows a deeper exchange of feature information between node A, node B, and edge, leading to a more complete encoding of the graph's information. Note that the learnable filter weight matrices $\boldsymbol{W}$ and the learnable bias vectors $\boldsymbol{b}$ of the second forward pass are the same as were used in the first forward pass. In principle, the learnable filter weight matrices $\boldsymbol{W}$ and the learnable bias vectors $\boldsymbol{b}$ in different forward passes of graph convolution do not need to be the same. However, our numerical experiments indicate that using two different sets of learnable filter weight matrices and bias vectors for the two forward passes leads to a propensity to overfit the training dataset. Therefore, we choose a shared set of learnable filter weight matrices and bias vectors for the two forward passes to build our GNN model.

$$\begin{aligned}\boldsymbol{x}^{(2)}_{\text{NodeA}} &= \boldsymbol{x}^{(1)}_{\text{NodeA}}\boldsymbol{W}_{\text{NodeA,NodeA}} + \boldsymbol{x}^{(1)}_{\text{NodeB}}\boldsymbol{W}_{\text{NodeB,NodeA}} \\ &\quad + \boldsymbol{x}^{(1)}_{\text{edge}}\boldsymbol{W}_{\text{edge,NodeA}} + \boldsymbol{b}_{\text{NodeA}}, \end{aligned} \tag{4}$$

$$\begin{aligned}\boldsymbol{x}^{(2)}_{\text{NodeB}} &= \boldsymbol{x}^{(1)}_{\text{NodeA}}\boldsymbol{W}_{\text{NodeA,NodeB}} + \boldsymbol{x}^{(1)}_{\text{NodeB}}\boldsymbol{W}_{\text{NodeB,NodeB}} \\ &\quad + \boldsymbol{x}^{(1)}_{\text{edge}}\boldsymbol{W}_{\text{edge,NodeB}} + \boldsymbol{b}_{\text{NodeB}}, \end{aligned} \tag{5}$$

$$\begin{aligned}\boldsymbol{x}^{(2)}_{\text{edge}} &= \boldsymbol{x}^{(1)}_{\text{NodeA}}\boldsymbol{W}_{\text{NodeA,edge}} + \boldsymbol{x}^{(1)}_{\text{NodeB}}\boldsymbol{W}_{\text{NodeB,edge}} \\ &\quad + \boldsymbol{x}^{(1)}_{\text{edge}}\boldsymbol{W}_{\text{edge,edge}} + \boldsymbol{b}_{\text{edge}}. \end{aligned} \tag{6}$$

The second forward pass of graph convolution is followed by another ELU activation layer and then a pooling layer concatenating the twice updated and non-linearly activated node A, node B, and edge features $\boldsymbol{x}^{(2)'}_{\text{NodeA}}$, $\boldsymbol{x}^{(2)'}_{\text{NodeB}}$, $\boldsymbol{x}^{(2)'}_{\text{edge}}$ together into a 10-dimensional vector. This 10-dimensional vector is in turn fed into a fully-connected layer with ELU activation followed by a 10-dimensional output layer. Once sufficiently trained, the output layer of our GNN model is supposed to generate the prediction for the graph level feature $\boldsymbol{y}_{\text{prediction}}$. The mean squared error (MSE) between $\boldsymbol{y}_{\text{prediction}}$ and ground truth $\boldsymbol{y}$ is calculated as the loss, which is back-propagated to the learnable filter weight matrices $\boldsymbol{W}$ and the learnable bias vectors $\boldsymbol{b}$ to optimize them using a gradient descent-based optimization algorithm.

Once the MSE loss between $\boldsymbol{y}_{\text{prediction}}$ and ground truth $\boldsymbol{y}$ has converged, the first stage involving training a GNN capable of predicting the graph level feature $\boldsymbol{y}$ from the node and edge features is accomplished. Our numerical experiments show that 3000 epochs of training are sufficient to reach convergence for our GNN model.

In the second stage of the inverse design process, which aims to generate the design parameters of a PIS that yields the desired thermal transport behavior, all the trainable weights of our GNN model, including the learnable filter weight matrices $\boldsymbol{W}$ and the learnable bias vectors $\boldsymbol{b}$, are frozen. The ground truth $\boldsymbol{y}$, which is a 10-dimensional reduced response vector representing the desired thermal transport behavior, is given. Employing a reduced response vector instead of a complete response vector primarily aims to prevent structural imbalances in the graph neural network and mitigate the risk of overfitting. This choice reduces the number of trainable neural network parameters that would otherwise increase substantially with a full response vector. Additionally, our prior research illustrates that compressing from a full response vector to a reduced one results in minimal information loss. The node A, node B, and edge features $\boldsymbol{x}_{\text{NodeA}}$, $\boldsymbol{x}_{\text{NodeB}}$, $\boldsymbol{x}_{\text{edge}}$ are randomly initialized and become the optimization target of any gradient descent-based optimization algorithm. The optimization objective is to tune node A, node B, and edge features $\boldsymbol{x}_{\text{NodeA}}$, $\boldsymbol{x}_{\text{NodeB}}$, $\boldsymbol{x}_{\text{edge}}$ so that our trained GNN model could output $\boldsymbol{y}_{\text{prediction}}$ as close to the ground truth $\boldsymbol{y}$ as possible. Our numerical experiments demonstrate that 1000000 optimization steps are needed to reach convergence for the optimization process. Once the optimization process is finished, the optimized node A, node B, and edge features $\boldsymbol{x}_{\text{NodeA}}$, $\boldsymbol{x}_{\text{NodeB}}$, $\boldsymbol{x}_{\text{edge}}$ become the design

parameters for a PIS, which is expected to yield the desired thermal transport behavior.

To train and validate the GNN model, we conducted 50000 finite-element method (FEM) simulations of PIS using the COMSOL Multiphysics commercial software package.[68] The design parameters for a PIS in the dataset are randomly generated within specified ranges detailed in Table 1. To ensure that the design parameter set is valid, an in-house Python code has been developed to filter out invalid design parameter sets. We divide the dataset into training, validation and test sets in a commonly accepted ratio of 8:1:1. The validation set is reserved for hyperparameter tuning, which includes selecting and evaluating the performance of different GNN architectures. Then, the best performing GNN architecture is evaluated on the test set. We use AdamW as an optimizer with a learning rate of 1×10^{-4}, simple linear decay, and a batch size of 100 for other hyperparameters. PyTorch[69] is used as the framework for training the GNN model and performing inferences, while pandas[70] is utilized for data analysis.

Table 1. Ranges of the design parameters, in which the design parameter sets for the training dataset are randomly generated. Specifically, the range for R is carefully chosen to prevent any overlap between particles A and particles B.

Design parameter	Range	Design parameter	Range
d_y	(0.8, 1.2) cm	e	(0.05, 0.2)
R	(0.67, 1.36) cm	$\kappa_{\rho\rho}$	(0, 1) W·m^{-1}·K^{-1}
ϕ	$(-\frac{\pi}{4}, \frac{\pi}{4})$	$\kappa_{\theta\theta}$	(0, 1) W·m^{-1}·K^{-1}
p_a	(0.05, 0.4)	κ_b	(0, 1) W·m^{-1}·K^{-1}
p_b	(0.05, 0.4)	κ_m	(0, 1) W·m^{-1}·K^{-1}

All the computations conducted in this research were performed on a dual-socket 24-physical-core Haswell-EP Xeon server. The 3000 epochs of training of the GNN model take around 6 hours to accomplish. Afterwards, we employed the snapshot of the GNN model trained for 3000 epochs for the inference process. For each interaction of inference, 1000000 optimization steps take 10 min to conclude. The FEM simulation usually takes approximately 2 min for a given set of design parameters.

3. Results

The second stage of inverse design to generate the design parameters of a PIS yielding the desired thermal transport behavior can be seen as an inference process. Since the node A, node B, and edge features $\boldsymbol{x}_{\text{NodeA}}$, $\boldsymbol{x}_{\text{NodeB}}$, $\boldsymbol{x}_{\text{edge}}$ are randomly initialized, different iterations of inference will lead to different sets of design parameters, which may or may not generate the desired thermal transport behavior to a satisfactory degree. However, the different sets of design parameters yielded by different iterations of inference should conform to some distribution from which a satisfactory set of design parameters can be sampled.

We illustrate the nature of the inference process using the thermal conductivity of the background, denoted as κ_m. Given that the background occupies a more significant portion of space than the combined area of particles A and B, κ_m plays a dominant role in influencing the baseline thermal transport behavior. This is derived from the Fourier thermal conductivity equation

$$\boldsymbol{q} = -\kappa \nabla T, \tag{7}$$

where $\boldsymbol{q}$ is the heat flux, κ is the thermal conductivity, and ∇T is the temperature gradient. In a scenario where the system contains solely the background and experiences a temperature difference of 40 K (313 K on one side and 273 K on the other side) within a simulation box measuring 20 cm in width, determining the value of κ_m becomes straightforward when the desired heat flux value is provided. In a more concrete example, if we want to obtain desired heat fluxes $J_S \approx 74$ W·m^{-2} and $J_S \approx 84$ W·m^{-2}, the value for κ_m should be 0.37 W·m^{-1}·K^{-1} and 0.42 W·m^{-1}·K^{-1}, respectively.

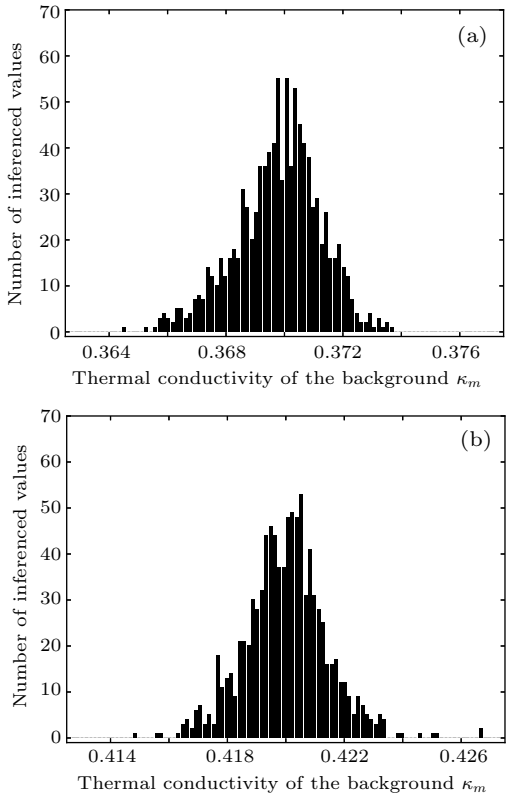


Fig. 3. The histogram of 1000 inferred values for the design parameter of thermal conductivity of the background κ_m. (a) The distribution of κ_m when thermal transparency with heat fluxes $J_S \approx 74$ W·m^{-2} is desired. (b) The distribution of κ_m when thermal transparency with heat fluxes $J_S \approx 84$ W·m^{-2} is desired.

We use the 10-dimensional reduced response vectors listed in Table 2 as the ground truth $\boldsymbol{y}$ to perform the inference process for 1000 times for each reduced response vector $\boldsymbol{y}$. These two reduced response vectors correspond to the perfect thermal transparency cases with desired heat fluxes $J_S \approx 74$ W·m^{-2} and $J_S \approx 84$ W·m^{-2} at the baseline. The

histograms for inferred κ_m are plotted in Fig. 3. In each case, our GNN model accurately forecasts that the distributions for κ_m are concentrated around 0.37 $\mathrm{W{\cdot}m^{-1}{\cdot}K^{-1}}$ and 0.42 $\mathrm{W{\cdot}m^{-1}{\cdot}K^{-1}}$, correspondingly.

Next, we address two inverse design challenges involving the attainment of thermal transparency for heat fluxes $J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$ and $J_S \approx 84$ $\mathrm{W{\cdot}m^{-2}}$, both measured at the baseline $x = -6$ cm. Utilizing the 10-dimensional reduced response vector presented in Table 2 as the ground truth $\boldsymbol{y}$, we conduct the inference process 100 times, generating 100 sets of design parameters for each scenario. Subsequently, FEM simulations are performed for each set of design parameters in both cases. The two sets of design parameters that demonstrate the closest approximation to thermal transparency for heat fluxes $J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$ and $J_S \approx 84$ $\mathrm{W{\cdot}m^{-2}}$ are detailed in Table 3 and illustrated in Figs. 4 and 5, respectively. The performance assessment of each system generated through our GNN-based approach relies on utilizing the MSE between heat flux values measured at 400 evenly spaced points on the baseline and those of perfect thermal transparency as the evaluation metric.

Table 2. The reduced response vectors employed as ground truth $\boldsymbol{y}$ for the inference process of the trained GNN model to obtain desired heat flux values $J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$ and $J_S \approx 84$ $\mathrm{W{\cdot}m^{-2}}$ at the baseline.

$J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$	9.99×10^{-1}	3.11×10^{-1}	8.95×10^{-1}	9.98×10^{-1}	1.15×10^{-6}	9.99×10^{-1}	1.00×10^{0}	9.02×10^{-1}	2.30×10^{-1}	6.09×10^{-3}
$J_S \approx 84$ $\mathrm{W{\cdot}m^{-2}}$	9.99×10^{-1}	3.02×10^{-1}	8.49×10^{-1}	9.97×10^{-1}	4.71×10^{-7}	9.99×10^{-1}	1.00×10^{0}	9.08×10^{-1}	1.67×10^{-1}	3.23×10^{-3}

Table 3. The design parameters of the systems to achieve thermal transparency at the baseline located at $x = -6$ cm with heat fluxes $J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$ and $J_S \approx 84$ $\mathrm{W{\cdot}m^{-2}}$. The design parameters are generated by the GNN model. The efficacy of each system in achieving thermal transparency is evaluated through the MSE, comparing heat flux values measured at 400 evenly spaced points against those of perfect thermal transparency. When the MSE value is below 1, a satisfactory system that achieves thermal transparency is deemed to have been found. Among the systems in randomly generated training sets, the closest one exhibits an MSE of 11.471 relative to perfect thermal transparency with a heat flux $J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$, and 27.138 relative to perfect thermal transparency with a heat flux $J_S \approx 84$ $\mathrm{W{\cdot}m^{-2}}$. The FEM simulated results for each system are depicted in Figs. 4 and 5.

Design parameter	d_y	R	ϕ	p_a	p_b	e	$\kappa_{\rho\rho}$	$\kappa_{\theta\theta}$	κ_b	κ_m	MSE
$J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$	0.978	1.274	–0.155	0.089	0.165	1.658	0.446	0.446	0.337	0.370	0.462
$J_S \approx 84$ $\mathrm{W{\cdot}m^{-2}}$	0.978	1.125	–0.047	0.197	0.053	1.168	0.446	0.446	0.337	0.420	0.324

In Figs. 4 and 5, we also showcase systems with the minimum MSE to perfect thermal transparency with heat fluxes $J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$ and $J_S \approx 84$ $\mathrm{W{\cdot}m^{-2}}$, drawn from the 40000 samples in the training dataset. In other words, these systems are the best random guess for the purpose of achieving thermal transparency. The MSE values for each system generated by our GNN-based approach are one order lower than those of the best performing systems in the randomly generated training set (the MSE values presented in the caption of Table 3). Our analysis leads to the conclusion that the design parameter sets identified by our GNN model significantly outperform the best outcomes from random guessing, underscoring the predictive and generative capabilities of our GNN model.

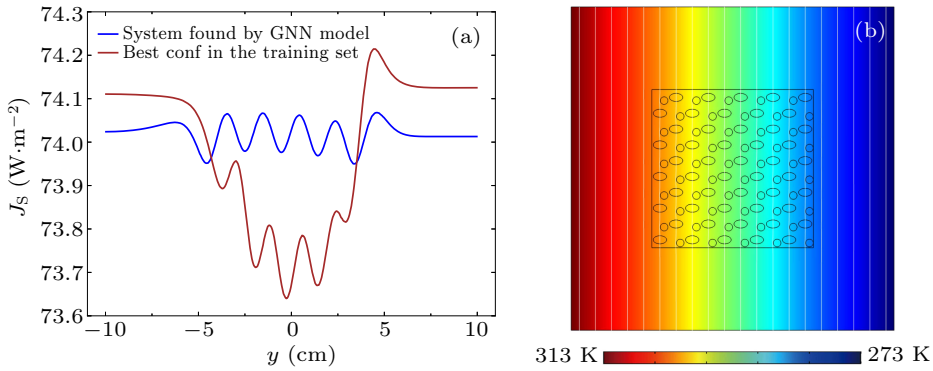


Fig. 4. Finite-element simulations for thermal transparency with heat fluxes $J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$. The size of the simulation box is 20×20 cm, and that of the periodic composite material is 10×10 cm. The baseline for measuring heat fluxes is located at $x = -6$ cm (not shown in the figures), and the origin is located at the center of the simulation box. (a) Heat flux J_S values on the baseline for simulated systems. The design parameters for the system represented by blue line are generated by the GNN model and shown in Table 3. The brown line is the heat fluxes measured at the baseline, generated by the system with the least MSE distance to the perfect thermal transparency with heat fluxes $J_S \approx 74$ $\mathrm{W{\cdot}m^{-2}}$, out of the 40000 systems in the randomly generated training set. (b) The FEM simulated result of the system represented by blue line in (a). The white lines represent isotherms. The corresponding design parameters are listed in Table 3. The blue line has an MSE distance 0.462 to the perfect thermal transparency, while the brown line has an MSE distance 11.471 to the perfect thermal transparency.

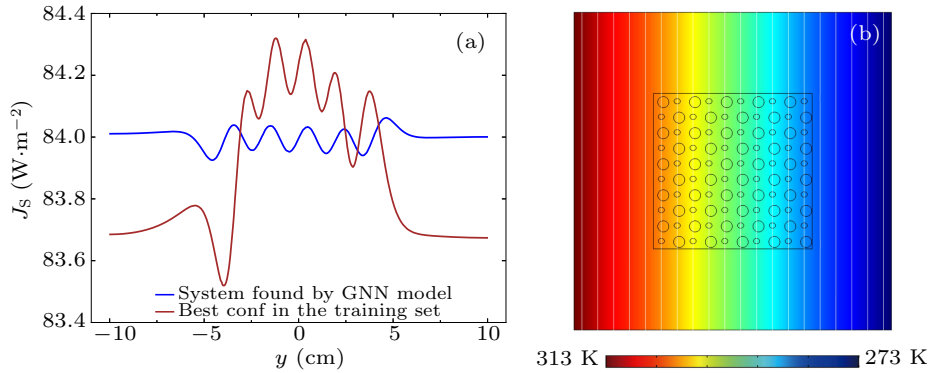


Fig. 5. Finite-element simulations for thermal transparency with heat fluxes $J_S \approx 84$ W·m^{-2}. The size of the simulation box is 20×20 cm, and that of the periodic composite material is 10×10 cm. The baseline for measuring heat fluxes is located at $x = -6$ cm (not shown in the figures), and the origin is located at the center of the simulation box. (a) Heat flux J_S values on the baseline for simulated systems. The design parameters for the system represented by blue line are generated by the GNN model and shown in Table 3. The brown line is the heat fluxes measured at the baseline, generated by the system with the least MSE distance to the perfect thermal transparency with heat fluxes $J_S \approx 84$ W·m^{-2}, out of the 40000 systems in the randomly generated training set. (b) The FEM simulated result of the system represented by blue line in (a). The white lines represent isotherms. The corresponding design parameters are listed in Table 3. The blue line has an MSE distance 0.324 to the perfect thermal transparency, while the brown line has an MSE distance 27.138 to the perfect thermal transparency.

4. Discussion

In contrast to our previous autoencoder-based method for inverse design,[54] the GNN-based approach not only presents as a paradigm shift from discriminative machine learning algorithms to generative algorithms but also demonstrates a few solid advantages in terms of computational efficiency and the removal of the artificial discretization of full design parameter values. In our prior autoencoder-based method, the many-to-one relationship from full to reduced parameters leads to an adverse outcome within the framework of the autoencoder-based method, i.e., we only had a forward neural network to map full design parameters to reduced ones because the neural network for the reverse mapping from reduced parameters to full ones was absent. Consequently, upon acquiring reduced design parameters enabling thermal transparency, we had to resort to an exhaustive grid search across feasible parameter ranges (with 10 intervals for each parameter) to obtain the corresponding original design parameters. Our GNN-based approach obviates the necessity for this exhaustive grid search to recover full design parameters. Moreover, the removal of the exhaustive grid search for full design parameters presents an additional advantage over the autoencoder-based method. In the autoencoder-based approach, the grid search for full design parameters introduces a side effect: the artificial discretization of full design parameters. This implies that only 10 discrete values can be considered for each design parameter. However, in our GNN-based approach, this limitation has been completely eliminated. The design parameters generated by the GNN-based can take any values within continuous feasible design parameter ranges.

The scope of application of our GNN-based inverse design approach is by no means limited to the periodic interparticle structures proposed in our manuscript. In other words, our GNN-based inverse design approach can be readily extended to other structures or geometries to realize the desired manipulation of thermal transport behavior. As long as a sufficiently large dataset comprising design parameters and their corresponding response vectors is built, theoretically, our GNN-based inverse design approach is applicable, paving the way for extending the application of our GNN-based inverse design approach to realize other kinds of functionalities, such as thermal cloaks and thermal camouflage. An example for which our GNN-based approach could be applicable is the inverse design for a three-dimensional invisible sensor with camouflaging in both thermal and electric fields. Yang[71] proposed the idea of invisible sensors with camouflaging in multiphysical fields and demonstrated a two-dimensional implementation in both simulations and experiments. Since a three-dimensional implementation of an invisible sensor with camouflaging in both thermal and electric fields can be fully characterized by the inner and outer radii of the homogeneous and isotropic shell, as well as the thermal and electrical conductivities of the sensor, the shell, and the background, the entire workflow of our GNN-based inverse design approach is naturally applicable. A graph representation for such an implementation of an invisible sensor with camouflaging is presented in Fig. 6(a). One technical challenge that we could envision is how to sample the resulting heat flux and DC current into discrete values to form a response vector. Other than that, the majority of our GNN-based inverse design workflow is straightforward to implement.

We also consider the potential of extending our GNN-based approach with the PIS as the underlying structure to achieve thermal transparency with an anisotropic background, i.e., making the effect of thermal transparency omnidirectional. We are inspired by Han's realization of omnidirectional thermal metadevices of anisotropic geometry,[72] in which the anisotropic background material is constructed as a periodic structure of a T-shaped unit with varying period L along the

x-axis of the background. A similar anisotropic background structure can be employed for our purpose. Furthermore, both particles A and B need to be allowed to have an elliptic shape with anisotropic thermal conductivity, giving the PIS system more degrees of freedom to accommodate system-wide anisotropic thermal conductivity. Then, the entire workflow of our GNN-based workflow can be applied to achieve the inverse design for thermal transparency. The graph representation for such an extended PIS is presented in Fig. 6(b). However, the major challenge that we will encounter is the dramatically increased computational cost, due to several reasons. First, because there are more design parameters now, a much larger training dataset needs to be constructed to allow adequate sampling along the dimension of each design parameter. Second, for each specific PIS, or equivalently each set of design parameters, multiple FEM simulations are needed because multiple directions for the temperature gradient are required to adequately account for the simulation box's anisotropic heat transfer behavior. Third, for each set of design parameters, the response generated by multiple FEM simulations would have a much higher dimension. Therefore, the dimensionality reduction effort to produce the reduced response vector will be more challenging to accomplish.

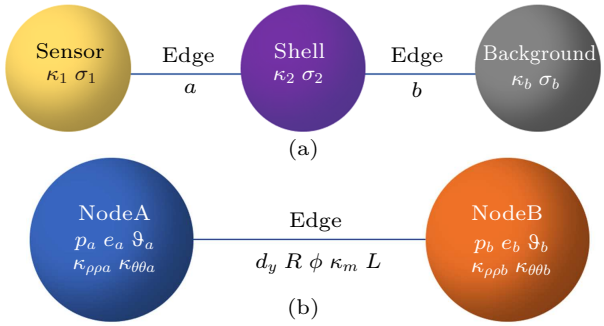


Fig. 6. The graph representations for thermal metamaterials-based systems. (a) Schematic diagrams of the graph representing the implementation of an invisible sensor with camouflaging. This implementation was proposed by Yang.[71] The spherical sensor with radius a, thermal conductivity κ_1 and electric conductivity σ_1 is surrounded by a spherical shell with inner radius a, outer radius b, thermal conductivity κ_2 and electric conductivity σ_2. The spherical shell is embedded into a background with thermal conductivity κ_b and electric conductivity σ_b. In the graph representation, the sensor is denoted by a node with features κ_1 and σ_1, the shell is denoted by a node with features κ_2 and σ_2, while the background is denoted by a node with features κ_b and σ_b. The sensor node and the shell node are connected by an edge with a feature a. The shell node and the background node are connected by an edge with a feature b. (b) Schematic diagrams of the graph representing a PIS that achieves thermal transparency with an anisotropic background. The structure of the anisotropic background was proposed by Han.[72] The background has a periodic structure of a T-shaped unit. The varying period along the x-axis of the background becomes a new feature L affiliated to edge. The meaning of the other features of edge is the same as those for edge in Fig. 2. To achieve thermal transparency with an anisotropic background, both particles A and B are augmented with new design parameters, which translate into new features affiliated to node A and node B, respectively. More specifically, both node A and node B have the feature e representing the ratio between two semi-axes, indicating that both particles A and B have an elliptic shape. Both node A and node B have anisotropic thermal conductivity $\kappa_{\rho\rho}$, $\kappa_{\theta\theta}$. The orientation of particles A and B with respect to the x-axis of the simulation box becomes features ϑ_a and ϑ_b for node A and node B, respectively.

We also aim to extend our GNN-based approach with the PIS as the underlying structure to enable the inverse design of thermal metamaterials with nonlinear time dependency. Nonlinear time dependency often results in transient heat transfer behavior, the manipulation of which is desirable yet challenging. Previous studies[73,74] have shown that incorporating density and specific heat capacity as tunable parameters is essential for manipulating transient heat transfer behavior. Therefore, we need to augment our existing PIS structure by including design parameters that represent density and specific heat capacity. However, several technical challenges remain to be addressed for a successful implementation, such as recording and discretizing transient heat fluxes into a response vector. Since heat flux values are now time-dependent, encoding time into the response vector is inevitable and likely computationally expensive. We are assimilating the latest advances in the AI algorithms to tackle these problems.

5. Conclusion

In this study, we introduce a thermal metamaterial-based approach for achieving thermal transparency, employing a PIS as the underlying structure. The central challenge in implementing this mechanism involves determining the design parameters based on the desired heat flux values. Consequently, this task can be characterized as an inverse design problem. We successfully address this inverse design problem by initially training a GNN model that is capable of predicting the graph-level feature $\boldsymbol{y}$, which represents thermal transport behavior, from the node and edge features. In the inference stage, we optimize the randomly initialized node and edge features to align the predicted graph level feature $\boldsymbol{y}_{\text{prediction}}$ to the ground truth $\boldsymbol{y}$.

Additionally, we showcase that multiple iterations of inference generate an accurate distribution for the specific design parameter of thermal conductivity of the background. Finally, we demonstrate that our GNN-based approach identifies satisfactory sets of design parameters for two specific cases of achieving thermal transparency with heat fluxes $J_S \approx 74\ \text{W}\cdot\text{m}^{-2}$ and $J_S \approx 84\ \text{W}\cdot\text{m}^{-2}$.

Data availability statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgements

We gratefully acknowledge funding from the National Natural Science Foundation of China (Grant Nos. 12035004 and 12320101004) and the Innovation Program of Shanghai Municipal Education Commission (Grant No. 2023ZKZD06).

We thank the Science Nexus lab at the University of Waterloo for providing the COMSOL simulation environment. We thank Dr. Gaole Dai for his contribution of Python codes.

References

[1] Xu L, Dai G and Huang J 2020 *Phys. Rev. Appl.* **13** 024063
[2] Yang S, Xu L J, Wang R Z and Huang J P 2017 *Appl. Phys. Lett.* **111** 121908
[3] Xu L, Xu G, Huang J and Qiu C W 2022 *Phys. Rev. Lett.* **128** 145901
[4] Shen X, Jiang C, Li Y and Huang J 2016 *Appl. Phys. Lett.* **109** 201906
[5] Xu L J and Huang J P 2020 *Chin. Phys. Lett.* **37** 120501
[6] Jin P, Xu L, Jiang T, Zhang L and Huang J 2020 *Int. J. Heat Mass Transfer* **163** 120437
[7] Yang F, Zhang Z, Xu L, Liu Z, Jin P, Zhuang P, Lei M, Liu J, Jiang J H, Ouyang X, Marchesoni F and Huang J 2024 *Rev. Mod. Phys.* **96** 015002
[8] Li Y, Li W, Han T, Zheng X, Li J, Li B, Fan S and Qiu C W 2021 *Nature Reviews Materials* **6** 488
[9] Yang S, Wang J, Dai G, Yang F and Huang J 2021 *Phys. Rep.* **908** 1
[10] Li Y, Shen X, Wu Z, Huang J, Chen Y, Ni Y and Huang J 2015 *Phys. Rev. Lett.* **115** 195503
[11] Shen X, Li Y, Jiang C, Ni Y and Huang J 2016 *Appl. Phys. Lett.* **109** 031907
[12] Fan C Z, Gao Y and Huang J P 2008 *Appl. Phys. Lett.* **92** 251907
[13] Chen T Y, Weng C N and Chen J S 2008 *Appl. Phys. Lett.* **93** 114103
[14] Narayana S and Sato Y 2012 *Phys. Rev. Lett.* **108** 214303
[15] Schittny R, Kadic M, Guenneau S and Wegener M 2013 *Phys. Rev. Lett.* **110** 195901
[16] Xu H Y, Shi X H, Gao F, Sun H D and Zhang B L 2014 *Phys. Rev. Lett.* **112** 054301
[17] Han T C, Bai X, Gao D L, Thong J T L, Li B W and Qiu C W 2014 *Phys. Rev. Lett.* **112** 054302
[18] Ma Y G, Liu Y C, Raza M, Wang Y D and He S L 2014 *Phys. Rev. Lett.* **113** 205501
[19] Dai G and Huang J 2018 *J. Appl. Phys.* **124** 235103
[20] Yang S, Xu L and Huang J 2019 *J. Appl. Phys.* **125** 055103
[21] Yang S, Xu L, Dai G and Huang J 2020 *J. Appl. Phys.* **128** 095102
[22] Zhu N Q, Shen X Y and Huang J P 2015 *AIP Advances* **5** 053401
[23] Xu L, Wang J, Dai G, Yang S, Yang F, Wang G and Huang J 2021 *Int. J. Heat Mass Transfer* **165** 120659
[24] Yang F, Xu L and Huang J 2019 *ES Energy Environment* **6** 45
[25] Kapadia R S and Bandaru P R 2014 *Appl. Phys. Lett.* **105** 233903
[26] Xu L J, Yang S and Huang J P 2018 *Phys. Rev. E* **98** 052128
[27] Han T C, Bai X, Thong J T L, Li B W and Qiu C W 2014 *Adv. Mater.* **26** 1731
[28] He X and Wu L Z 2014 *Appl. Phys. Lett.* **105** 221904
[29] Yang T Z, Su Y, Xu W and Yang X D 2016 *Appl. Phys. Lett.* **109** 121905
[30] Hu R, Zhou S L, Li Y, Lei D Y, Luo X B and Qiu C W 2018 *Adv. Mater.* **30** 1707237
[31] Zhou S L, Hu R and Luo X B 2018 *Int. J. Heat Mass Transfer* **127** 607
[32] Xu L J and Huang J P 2018 *Phys. Lett. A* **382** 3313
[33] Xu L, Wang R and Huang J 2018 *J. Appl. Phys.* **123** 245111
[34] Li Y, Bai X, Yang T, Luo H and Qiu C W 2018 *Nat. Commun.* **9** 273
[35] Peng Y G, Li Y, Cao P C, Zhu X F and Qiu C W 2020 *Adv. Funct. Mater.* **30** 2002061
[36] He X and Wu L Z 2013 *Phys. Rev. E* **88** 033201
[37] Zeng L W and Song R X 2014 *Appl. Phys. Lett.* **104** 201905
[38] Yang T Z, Bai X, Gao D L, Wu L Z, Li B W, Thong J T L and Qiu C W 2015 *Adv. Mater.* **27** 7752
[39] Wang R Z, Xu L J, Ji Q and Huang J P 2018 *J. Appl. Phys.* **123** 115117
[40] Dong L, Huang J P, Yu K W and Gu G Q 2003 *J. Appl. Phys.* **95** 621
[41] Huang J P, Karttunen M, Yu K W and Dong L 2003 *Phys. Rev. E* **67** 021403
[42] Li Y, Zhu K J, Peng Y G, Li W, Yang T, Xu H X, Chen H, Zhu X F, Fan S and Qiu C W 2019 *Nat. Mater.* **18** 48
[43] Li J, Li Y, Cao P C, Yang T, Zhu X F, Wang W and Qiu C W 2020 *Adv. Mater.* **32** 2003823
[44] Xu G, Dong K, Li Y, Li H, Liu K, Li L, Wu J and Qiu C W 2020 *Nat. Commun.* **11** 6028
[45] Li Y, Qi M, Li J, Cao P C, Wang D, Zhu X F, Qiu C W and Chen H 2022 *Nat. Commun.* **13** 2683
[46] Guo J, Xu G, Tian D, Qu Z and Qiu C W 2022 *Adv. Mater.* **34** 2201093
[47] Guo J, Xu G, Tian D, Qu Z and Qiu C W 2022 *Adv. Mater.* **34** 2200329
[48] Sha W, Xiao M, Zhang J, Ren X, Zhu Z, Zhang Y, Xu G, Li H, Liu X, Chen X, Gao L, Qiu C W and Hu R 2021 *Nat. Commun.* **12** 7228
[49] Sha W, Hu R, Xiao M, Chu S, Zhu Z, Qiu C W and Gao L 2022 *Npj Comput. Mater.* **8** 179
[50] Yu S, Zhou P, Xi W, Chen Z, Deng Y, Luo X, Li W, Shiomi J and Hu R 2023 *Light Sci. Appl.* **12** 291
[51] Chen Z, Yu S, Yuan C, Hu K and Hu R 2023 *J. Appl. Phys.* **134** 203101
[52] Wang Z, Zhu Z, Liu T and Hu R 2022 *J. Appl. Phys.* **132** 145102
[53] Fujii G, Akimoto Y and Takahashi M 2018 *Appl. Phys. Lett.* **112** 061108
[54] Liu B, Xu L and Huang J 2021 *J. Appl. Phys.* **129** 065101
[55] Liu B, Xu L and Huang J 2021 *J. Appl. Phys.* **130** 045103
[56] Xu L, Yang S and Huang J 2019 *Phys. Rev. Applied* **11** 034056
[57] Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, Wang L, Li C and Sun M 2020 *AI Open* **1** 57
[58] Scarselli F, Gori M, Tsoi A C, Hagenbuchner M and Monfardini G 2009 *IEEE Transactions on Neural Networks* **20** 61
[59] Monti F, Bronstein M M and Bresson X 2017 arXiv:1704.06803 [cs.LG]
[60] Veličković P, Cucurull G, Casanova A, Romero A, Liò P and Bengio Y 2018 arXiv:1710.10903 [stat.ML]
[61] Li Y, Yu R, Shahabi C and Liu Y 2018 arXiv:1707.01926 [cs.LG]
[62] Wong F, Zheng E J, Valeri J A, Donghia N M, Anahtar M N, Omori S, Li A, Cubillos-Ruiz A, Krishnan A, Jin W, Manson A L, Friedrichs J, Helbig R, Hajian B, Fiejtek D K, Wagner F F, Soutter H H, Earl A M, Stokes J M, Renner L D and Collins J J 2024 *Nature* **626** 177
[63] Zhang Y, Yao Q, Yue L, Wu X, Zhang Z, Lin Z and Zheng Y 2023 *Nat. Comput. Sci.* **3** 1023
[64] Sun X, Jia X, Lu Z, Tang J and Li M 2023 *Bioinformatics* **40** btad748
[65] Han J, Cen J, Wu L, Li Z, Kong X, Jiao R, Yu Z, Xu T, Wu F, Wang Z, Xu H, Wei Z, Liu Y, Rong Y and Huang W 2024 arXiv:2403.00485 [cs.LG]
[66] Dold D and Aranguren van Egmond D 2023 *Cell Rep. Phys. Sci.* **4** 101586
[67] Kipf T N and Welling M 2017 arXiv:1609.02907 [cs.LG]
[68] See https://www.comsol.com/ for information about the COMSOL Multiphysics software package.
[69] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Köpf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J and Chintala S 2019 Pytorch: An imperative style, high-performance deep learning library
[70] The pandas development team 2020 pandas-dev/pandas: Pandas
[71] Yang T, Bai X, Gao D, Wu L, Li B, Thong J and Qiu C W 2015 *Adv. Mater.* **27** 7752
[72] Han T, Yang P, Li Y, Lei D, Li B, Hippalgaonkar K and Qiu C W 2018 *Adv. Mater.* **30** 1804019
[73] Li Y, Shen X Y, Huang J P and Ni Y S 2016 *Phys. Lett. A* **380** 1641
[74] Zhang X W, He X and Wu L Z 2022 *Int. J. Heat Mass Transfer* **193** 122960