Journal of Machine Learning **ISSN**: 2790-2048(e), 2790-203X(p)

RNN-Attention Based Deep Learning for Solving Inverse Boundary Problems in Nonlinear Marshak Waves

Di Zhao [1], Weiming Li[1], Wengu Chen [1], Peng Song[1,2], and Han Wang [*1,2]

[1]Institute of Applied Physics and Computational Mathematics, Beijing 100088, China.
[2]HEDPS, Center for Applied Physics and Technology, College of Engineering, Peking University, Beijing 100871, China.

Abstract. Radiative transfer, described by the radiative transfer equation (RTE), is one of the dominant energy exchange processes in the inertial confinement fusion (ICF) experiments. The Marshak wave problem is an important benchmark for time-dependent RTE. In this work, we present a neural network architecture termed RNN-attention deep learning (RADL) as a surrogate model to solve the inverse boundary problem of the nonlinear Marshak wave in a data-driven fashion. We train the surrogate model by numerical simulation data of the forward problem, and then solve the inverse problem by minimizing the distance between the target solution and the surrogate predicted solution concerning the boundary condition. This minimization is made efficient because the surrogate model by-passes the expensive numerical solution, and the model is differentiable so the gradient-based optimization algorithms are adopted. The effectiveness of our approach is demonstrated by solving the inverse boundary problems of the Marshak wave benchmark in two case studies: where the transport process is modeled by RTE and where it is modeled by its nonlinear diffusion approximation (DA). Last but not least, the importance of using both the RNN and the factor-attention blocks in the RADL model is illustrated, and the data efficiency of our model is investigated in this work.

Keywords:
Marshak Wave,
Inverse Problem,
Deep Learning,
Surrogate Model.

Article Info.:
Volume: 2
Number: 2
Pages: 83- 107
Date: June/2023
doi.org/10.4208/jml.221209

Article History:
Received: 09/12/2022
Accepted: 27/03/2023

Handler:
Weinan E

1 Introduction

Inertial confinement fusion (ICF) refers to the fusion energy released when a small amount of hot nuclear fuel is ignited by high-power substances (such as laser, electron beam, and ion beam) to make it reach the ignition conditions under inertial confinement. The laser inertial confinement fusion device uses a powerful laser to irradiate the target, compress the fuel inward, and the plasma formed by the target material is heated to a very high temperature and a fusion reaction occurs before it has time to fly around due to its own inertia, full thermonuclear combustion is carried out to release a large amount of fusion energy.

The ICF implosion is characterized by the equations of radiation hydrodynamics, which are mainly composed of equations describing fluid motion, electron heat conduction, ion heat conduction, photon transport, nuclear reaction and charged particle transport [8, 30, 35]. Among the processes, the photon transport in a medium which absorbs, emits, and scatters radiation is governed by the frequency-dependent radiative transfer equa-

*wang_han@iapcm.ac.cn

tion. One of the main challenges in the numerical simulation of the RTE is its high-dimensionality, as its independent variables include time, space, the radiation frequency and the propagation direction of photons. In practice, a common way of addressing this challenge is to reduce its dimension by integrating the transport equation against all angles and approximating the radiation flux using the Fick's law of diffusion. This yields the diffusion approximation (DA) to the transfer equation [4]. When away from the boundary and initial layers, the DA is valid in optically thick regions where the photon's mean free path is small, but it leads to notable deviation in the optically thin regions [26, 32, 34, 44].

In this work, we mainly concern with the material temperature and radiation temperature in the ICF process, which are spatial-temporal distributed functions, and investigate the inverse problem: Given the desired target temperature, we are aiming to work out a boundary condition with which the solution of the RTE and its nonlinear diffusion approximation approaches the target temperature. The corresponding forward problem is thus to predict the material temperature from an input boundary condition. The accuracy and efficiency of solving the inverse problem largely depend on the effectiveness of the forward problem solver. A common practice is to build the forward problem surrogate models that are expected to preserve the accuracy of the original forward problem solution, and to substantially relieve the computational burden [2, 9, 15, 18, 31, 46]. Recently deep artificial neural networks are used to build surrogate models [36, 43]. From the theoretical perspective, the success of the deep neural networks may be attributed to their power of universal approximation [5, 21], especially for high-dimensional functions [12, 13].

In the forward RTE problem, both the boundary condition and the solution (material temperature) are time sequences, thus the ordering of the input and output should be properly taken into consideration in the construction of the surrogate forward model. A large amount of model constructions that map sequence to sequence (seq2seq) have been proposed in the field of natural language processing (NLP), including but not limited to recurrent neural network [24], Long Short-Term Memory (LSTM) [20], bidirectional recurrent neural network (BRNN) [42], bidirectional LSTM [17]. Recently large scale pre-trained models based on the attention architecture [47] have prospered, and some examples are XLNet [48], ELMo [37], BERT [11] and GPT-3 [6]. Using the seq2seq modeling to solve PDE problems is not unprecedented. Anshuman *et al.* applied a novel LSTM-based seq2 model to solve the groundwater contaminant sources identification and parameter estimation. The model takes both sequential inputs to predict breakthrough curves at observation points, effectively reducing the computational cost caused by the numerous runs of the computationally expensive optimization algorithm [3]. To the best of our knowledge, the seq2seq idea has not yet been used to establish the surrogate model and solve the inverse RTE problems.

In this work, we introduce an RNN-attention deep learning (RADL) model architecture for the surrogate model of the RTE and its diffusion approximation. We have shown that both the RNN and the factor-attention blocks in the RADL model are crucial for accuracy. Besides the high efficiency, the RADL model is differentiable, which makes it possible to solve the inverse problem with gradient-based optimization algorithms. The effectiveness of the proposed RADL is demonstrated by solving the Marshak wave problem for a model RTE and its diffusion approximation.

The rest of this paper is organized as follows. In Section 2, we introduce the preliminary knowledge of the equations, involving the radiative transfer equation, the gray equation of transfer and its diffusion approximation, the Marshak wave problem and the corresponding inverse problem setup. Section 3 introduces our RADL surrogate model architecture and the methodology for solving the inverse boundary problem. Section 4 exhibits numerical experiments. Section 5 concludes the work.

2 Preliminaries

2.1 The frequency-dependent radiative transfer equation

The radiative transfer equation is a mathematical description of the conservation of photon as it transports in medium. Under the assumption that the scattering kernel is both coherent and isotropic, and in the absence of internal source, the RTE is written in the following form [40]:

$$\frac{1}{c}\frac{\partial I}{\partial t} + \boldsymbol{\Omega}\cdot\nabla I = -\big(\sigma_a(\nu,T)+\sigma_s(\nu,T)\big)I + \sigma_a(\nu,T)B(\nu,T) + \frac{\sigma_s(\nu,T)}{4\pi}\int_{\mathbb{S}^2} I\,\mathrm{d}\boldsymbol{\Omega}. \tag{2.1}$$

In Eq. (2.1), $I(\boldsymbol{x},t,\nu,\boldsymbol{\Omega})$ is the specific intensity of radiation, $t\in\mathbb{R}^+$ denotes time, $\boldsymbol{x}\in\mathbb{R}^3$ are the spatial coordinates, and $\nu\in\mathbb{R}^+$ is frequency. $\boldsymbol{\Omega}$ is the angular variable which lies on $\mathbb{S}^2$, the surface of the unit sphere, denoting the direction in which radiation propagates. c denotes the speed of light. σ_a and σ_s are the absorption and scattering coefficients respectively, describing the interaction of photons with matter. Note that σ_a is modified to take into account induced emission. The assumption of local thermodynamic equilibrium (LTE) leads to $B(\nu,T)$ taking the form of the Planck function

$$B(\nu,T) = \frac{2h\nu^3}{c^2}\big(e^{h\nu/k_{\mathrm{B}}T}-1\big)^{-1}, \tag{2.2}$$

where h is Planck's constant, k_{B} is Boltzmann's constant, and T is the local temperature of matter. We will discuss the governing equation of T in the next section.

We consider a system confined within a volume D with its boundary denoted by ∂D. The initial condition at $t=0$ can be given by

$$I(\boldsymbol{x},0,\nu,\boldsymbol{\Omega}) = \Lambda(\boldsymbol{x},\nu,\boldsymbol{\Omega}), \quad \forall(\boldsymbol{x},\nu,\boldsymbol{\Omega})\in D\times[0,\infty]\times\mathbb{S}^2, \tag{2.3}$$

where Λ is a specified function. Assuming the system surface is non-re-entrant and considering a simulation time interval of $t\in[0,t_{\mathrm{f}}]$, where t_{f} is the ending time, one can impose the inflow boundary condition

$$I(\boldsymbol{x},t,\nu,\boldsymbol{\Omega}) = \Gamma(\boldsymbol{x},t,\nu,\boldsymbol{\Omega}) \quad \text{on } \partial D_-, \tag{2.4}$$

where $\partial D_- = \{(\boldsymbol{x},t,\nu,\boldsymbol{\Omega})\in\partial D\times[0,t_{\mathrm{f}}]\times[0,\infty]\times\mathbb{S}^2 : \boldsymbol{n}(\boldsymbol{x})\cdot\boldsymbol{\Omega}<0\}$, $\boldsymbol{x}$ is an arbitrary point on ∂D, and $\boldsymbol{n}(\boldsymbol{x})$ is the outward unit normal vector at this point. Γ is a specified function of all its arguments. The above initial (2.3) and boundary (2.4) conditions together with Eq. (2.1), completely specify the radiative transfer problem.

2.2 The gray equation of transfer and its diffusion approximation

Our discussion focuses on the case where there is no scattering and the absorption coefficient $\sigma_a(T)$ is independent of frequency, but in general depends on the temperature T. In this case, integrating the RTE over the whole frequency domain yields the gray equation of transfer [40]

$$\frac{1}{c}\frac{\partial I}{\partial t} + \mathbf{\Omega}\cdot\nabla I = \sigma_a(T)\left(\frac{1}{4\pi}acT^4 - I\right). \tag{2.5}$$

With some abuse of notation, here $I(\boldsymbol{x}, t, \mathbf{\Omega})$ is the integration of the specific intensity against $\nu \in [0,\infty)$. a is the radiation constant given by

$$a = \frac{8\pi^5 k_{\mathrm{B}}^4}{15h^3c^3}. \tag{2.6}$$

Following the dimensional analysis procedure in [33], we write the gray equation of transfer in nondimensional form as

$$\frac{\epsilon^2}{c}\frac{\partial I}{\partial t} + \epsilon\mathbf{\Omega}\cdot\nabla I = \sigma_a(T)\left(\frac{1}{4\pi}acT^4 - I\right), \tag{2.7}$$

where ϵ is the ratio between the typical mean free path and the macroscopic length scale. In the absence of convection and heat conduction, as in [10,44], we consider the case where material temperature, denoted by $T(\boldsymbol{x}, t)$, satisfies the energy balance equation

$$\epsilon^2 C_v\frac{\partial T}{\partial t} \equiv \epsilon^2\frac{\partial U}{\partial t} = \sigma_a(T)\left(\int_{\mathbb{S}^2} I\,\mathrm{d}\Omega - acT^4\right). \tag{2.8}$$

$C_v(\boldsymbol{x}, t)$ is the heat capacity. The relationship between the material temperature $T(\boldsymbol{x}, t)$ and the material energy density $U(\boldsymbol{x}, t)$ satisfies

$$\frac{\partial U}{\partial T} = C_v > 0. \tag{2.9}$$

Integrating Eq. (2.7) over $\mathbf{\Omega}$ and combining with Eq. (2.8) produce the conservation of energy

$$\epsilon C_v\frac{\partial T}{\partial t} + \epsilon\frac{\partial E}{\partial t} + \int_{\mathbb{S}^2}\mathbf{\Omega}\cdot\nabla I\,\mathrm{d}\mathbf{\Omega} = 0, \tag{2.10}$$

where E is the energy density defined as

$$E = \frac{1}{c}\int_{\mathbb{S}^2} I\,\mathrm{d}\mathbf{\Omega}. \tag{2.11}$$

The total energy is then defined as

$$\mathcal{E} = U + E. \tag{2.12}$$

When ϵ goes to zero, the specific intensity goes to a Planckian at the local temperature [4,45]. In such cases, the radiative flux $F(\boldsymbol{x}, t)$ is related to the material temperature by the Fick's law of diffusion given by

$$F(\boldsymbol{x}, t) = \int_{\mathbb{S}^2}\mathbf{\Omega} I\,\mathrm{d}\mathbf{\Omega} = -\frac{ac}{3\sigma}\nabla T^4. \tag{2.13}$$

Therefore, when away from initial and boundary layers, the corresponding local temperature $T^{(0)}$ satisfies the nonlinear diffusion equation

$$\frac{\partial U\big(T^{(0)}\big)}{\partial t} + a\frac{\partial}{\partial t}\big(T^{(0)}\big)^4 = \nabla\cdot\frac{ac}{3\sigma}\nabla\big(T^{(0)}\big)^4. \tag{2.14}$$

Moreover, at this time the total energy equation (2.12) is expressed as

$$\mathcal{E} = U^{(0)} + a\big(T^{(0)}\big)^4. \tag{2.15}$$

2.3 The Marshak wave problem

The Marshak wave problem is an important benchmark for time-dependent RTE simulations. Its setup consists of an initially cold material occupying a halfspace with radiation incident on its boundary [39]. We are concerned with how the radiation wavefront penetrates the slab.

This problem has been studied in various literature [23, 38, 39, 44]. In our studies, we consider a system with slab geometry, the same as in [44]. Also, scattering effects are omitted, and the absorption coefficient is assumed to be independent of frequency. Therefore, the gray equation of transfer characterizing this physical process could be written in the following dimension reduced form:

$$\begin{cases} \dfrac{1}{c}\dfrac{\partial I}{\partial t} + \mu\dfrac{\partial I}{\partial x} = -\sigma_a(T)I + \dfrac{1}{2}\sigma_a(T)\,acT^4, \\ C_v\dfrac{\partial T}{\partial t} = \sigma_a(T)\displaystyle\int_{-1}^{1} I(\mu)d\mu - \sigma_a(T)acT^4, \end{cases} \tag{2.16}$$

where $\mu = \Omega_x$. The corresponding initial condition for Eq. (2.16) is

$$I(x,0) = \frac{1}{2}acT_r^4, \quad T(x,0) = T_r(x), \quad \forall x \in [0,\infty), \tag{2.17}$$

which means that we assume initially the specific intensity is isotropic, and that initially radiation and material temperature are equal. The following inflow boundary condition is imposed upon $x = 0$:

$$I(0,t,\mu) = \frac{1}{2}acT_{bd}^4, \quad \forall t \geq 0, \quad \mu > 0. \tag{2.18}$$

The boundary condition is assumed to be isotropic in terms of the photon propagation direction, i.e. $I(0,t,\mu)$ is independent of μ. When $\sigma_a(T)$ approaches infinity, the same analysis as in Section 2.2 yields the following diffusion approximation:

$$\frac{\partial}{\partial t}\big(aT^4 + C_vT\big) - \frac{\partial}{\partial x}\left(\frac{1}{3\sigma_a(T)}\cdot\frac{\partial}{\partial x}\big(acT^4\big)\right) = 0. \tag{2.19}$$

The interpretation of the physical quantities are the same as defined in Sections 2.1 and 2.2 except that the x is the 1D spatial variable, and $\mu = \Omega_x \in [-1,1]$ is the cosine of the angle between the photon propagation direction and the x coordinate.

Following the setup in [44], we take

$$\sigma_a(T) = \sigma_{a,0}\left(\frac{T_{\text{keV}}}{T}\right)^3, \tag{2.20}$$

where T_{keV} is the temperature corresponding to the energy of 1 keV, i.e. $T_{\text{keV}} = 1\ \text{keV}/k_{\text{B}}$ with k_{B} being the Boltzmann constant. $\sigma_{a,0}$ is a constant of unit cm^{-1}. Other constants are chosen as $C_v = 0.0259\,\text{GJ/MK/cm}^3$, $c = 29.98\,\text{cm/ns}$, $a = 7.5651 \times 10^{-7}\text{GJ/cm}^3/\text{MK}^4$. The initial material temperature $T(x,0)$ is set to $T(x,0) \equiv 10^{-6}\text{keV}/k_{\text{B}}$.

The physical process described by the Marshak wave problem could be approximated either by direct solving the transport equation, or its diffusion approximation. In generating our data, we solve both equations. When solving the transport equation, we employ the spherical harmonics (P_n) method for angular discretization. The P_n method is a popular way of approximating the RTE, and more details of its derivation and properties could be found in [40]. Specifically, we use the P_{11} system, and evaluate it using the finite volume method. In solving the diffusion equation, we employ a backward Euler scheme for temporal discretization, and a central difference scheme for spatial discretization. The resulting nonlinear equation for T is solved using standard Newton iteration. We will refer to solutions produced by the above methods as the ground truth.

2.4 The inverse problem of the transport equation and its diffusion approximation

Assuming that all the constants and the initial condition are given and that a compact time domain $[0, t_{\text{f}}]$ is considered, the forward problem of the Marshak wave equation (2.16) is a functional from the boundary condition $T_{bd}(t)$ to the material temperature $T(x,t)$, which is defined as

$$T = \mathcal{F}[T_{bd}]. \tag{2.21}$$

Given a target material temperature $T^* = T^*(x,t)$, the inverse boundary problem of the Marshak wave is to find a boundary condition, denoted by $\hat{T}_{bd}(t)$, so that the solution of the forward problem is as close to the target T^* as possible, i.e.

$$\hat{T}_{bd} = \arg\min_{T_{bd}} \mathcal{L}\big(\mathcal{F}[T_{bd}], T^*\big), \tag{2.22}$$

where $\mathcal{L}$ is a measure of the distance between the two spatial-temporal functions. In this work, the distance in the L^2 sense is used, i.e.

$$\mathcal{L}\big(\mathcal{F}[T_{bd}], T^*\big) = \int \big|\mathcal{F}[T_{bd}](x,t) - T^*(x,t)\big|^2 dxdt. \tag{2.23}$$

Other measures can be considered analogously without substantial difficulty as long as they are smooth with respect to the inputs.

The inverse problem (2.22) is an optimization problem which can be solved by well-developed algorithms, like the gradient descent method. Solving the optimization problem requires solving the forward problem multiple times, i.e. solving the RTE multiple

times, which can be very expensive. Moreover, if the optimization algorithm requires the evaluation of the gradients, a common requirement of the gradient-based optimization algorithms, the gradient of the forward mapping with the boundary condition should be evaluated. This further increases the computational cost. Therefore, we propose a surrogate model

$$\mathcal{F}^{\mathrm{s}}[T_{bd}] \approx \mathcal{F}[T_{bd}], \tag{2.24}$$

which is an approximation of the forward problem (2.21). We require that the computational cost of the surrogate is inexpensive, and that the gradients of the surrogate model should be easy to evaluate. If the surrogate model approximates the forward problem very well, we solve the surrogate inverse problem

$$\hat{T}^{s}_{bd} = \arg\min_{T_{bd}} \mathcal{L}\big(\mathcal{F}^{\mathrm{s}}[T_{bd}], T^{*}\big), \tag{2.25}$$

instead of the original inverse problem (2.22). If the surrogate model is a close approximation to the original forward problem, one expect that the loss $\mathcal{L}(\mathcal{F}[\hat{T}^{s}_{bd}], T^{*})$ is close to zero. The inverse and surrogate-based inverse problems of the diffusion approximation can be introduced in an analogous way to Eqs. (2.22) and (2.25).

3 Method

In this section, we first give an explicit introduction to our RNN attention deep learning surrogate model architecture. Then we present how to apply the already built-up surrogate model to the inverse boundary problem to efficiently produce an approximated boundary condition on the promise of a given desired target solution.

3.1 Surrogate model architecture

The structure of the RADL model is schematically illustrated in Fig. 3.1. The time-discretized boundary condition $T_{bd}(t_i)$, where $i = 1, \ldots, N$, with N being the number of the temporal discretization grid points. Each $T_{bd}(t_i)$ represents the boundary value at the temporal grid t_i, is passed to a factor-attention block and an RNN block to extract the inherent content features and the time-series features, respectively. Then the features are concatenated and passed through a fitting block to predict the spatial-temporal discretized solution $T(x_j, t_i)$, $j = 1, \ldots, M$, with M being the number of spatial discretization grid points, and $i = 1, \ldots, N$. $T(x_j, t_i)$ represents the numerical solution at the spatial-temporal grid (x_j, t_i). We now present the details of the model structure. The factor-attention block is composed of the factorization layer and the attention layer.

Factorization layer. The time-discretized boundary condition $T_{bd}(t_i)$ is firstly embedded into a hidden space by a linear transform

$$\boldsymbol{g}^{f}_{i} = \boldsymbol{w} T_{bd}(t_i), \tag{3.1}$$

where $\boldsymbol{w} \in \mathbb{R}^{N_h}$ is the trainable weight, with N_h being the dimension of the hidden space.

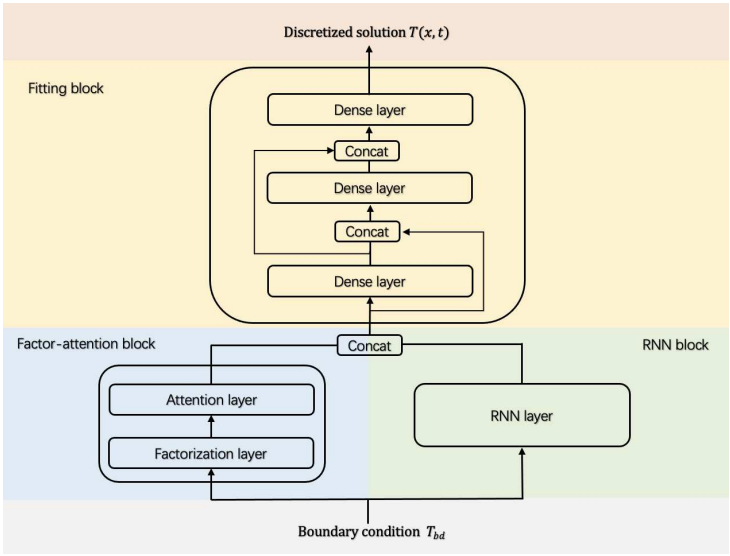

Figure 3.1: A schematic plot of the architecture of the RADL model.

Attention layer. The attention mechanism is usually used to encode long-range correlation in the input series into the feature [47]. From the data-driven perspective, all kinds of information in the equation are expressed with data as the carrier. Attention score can reflect the relationship between the inputs, and a strong correlation leads to a large attention score, so the attention mechanism can be regarded as a weighted sum of all the input information. The factorized boundary condition g_i^f is passed through a multi-head attention layer denoted by A

$$\{g_i^a\} = A\left(\left\{g_i^f\right\}\right), \tag{3.2}$$

where both the input $\{g_i^f\}$ and the output feature $\{g_i^a\}$ of the multi-head attention layer are time series in the N_h dimensional hidden space, with each element in the series denoted by g_i^f and g_a^i, respectively. The attention layer is permutationally covariance, which means if we exchange the order of any two steps in the input series, the corresponding output steps also exchange. The architecture of the attention layer A is explained in detail in Appendix A.

RNN block. RNN is a type of neural network commonly used for processing sequential data such as time series data. RNNs have loops that allow information to persist and be passed from one step of the network to the next. This allows RNNs to model sequential dependencies and make use of information from previous inputs. The RNN block in our paper, denoted by R, is used to encode the temporal dynamics of the boundary condition $T_{bd}(t)$, i.e.

$$\{g_i^r\} = R(\{T_{bd}(t_i)\}). \tag{3.3}$$

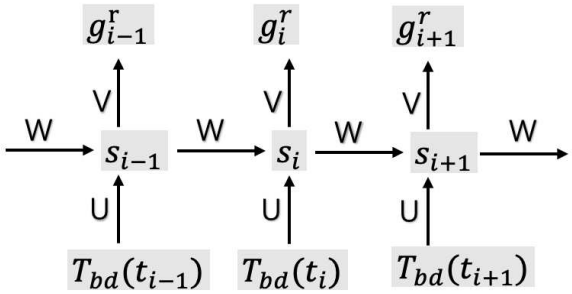

Figure 3.2: The RNN block structure.

Fig. 3.2 shows the basic structure of the RNN block. The calculation process can be succinctly described as follows:

$$s_i = \tanh\left(T_{bd}(t_i)U^T + s_{i-1}W^T\right), \tag{3.4}$$

$$g_i^r = \tanh\left(s_i V^T\right). \tag{3.5}$$

In this expression, the hyperbolic tangent activation function $\tanh(\cdot)$ [7] is utilized, and the learnable weights W, U and V play a crucial role in determining the output of the RNN block. The hidden state s_i at each time step i is calculated based on the combination of these weights and biases, as well as the current input at time step $T_{bd}(t_i)$. As a result of this calculation, the output feature g_i^r of the RNN block is represented as a time series in the hidden space, where each output is a function of the prior node's information s_{i-1} and the current input $\{T_{bd}(t_i)\}$ information. This allows the RNN to maintain information from previous time steps and effectively process sequences of inputs, making it well-suited for tasks that involve time-series correlation.

Fitting block. The output features of the factor-attention block ($\boldsymbol{g}_i^a$) and that of the RNN block ($\boldsymbol{g}_i^r$) are concatenated as the overall input feature $\boldsymbol{g}_i = \text{concat}(\boldsymbol{g}_i^a, \boldsymbol{g}_i^r) \in \mathbb{R}^{2N_h}$, and are sent to the fitting block for final prediction $T(x,t)$. The fitting block is a customized neural network involving three feed-forward cells, each cell composed of several dense layers. Additionally, there are two concatenation operations between the network cells to assist in learning the function's mapping knowledge better. The prediction process of the fitting block can be represented as

$$\begin{aligned}
&\boldsymbol{h}_i^{(1)} = \mathrm{G}_{\mathrm{BN}}^{(1,2)} \circ \mathrm{L}^{(1,2)} \circ \mathrm{G}_{\mathrm{BN}}^{(1,1)} \circ \mathrm{L}^{(1,1)}(\boldsymbol{g}_i), \\
&\boldsymbol{h}_i^{(2)} = \mathrm{G}_{\mathrm{BN}}^{(2,3)} \circ \mathrm{L}^{(2,3)} \circ \mathrm{G}_{\mathrm{BN}}^{(2,2)} \circ \mathrm{L}^{(2,2)} \circ \mathrm{G}_{\mathrm{BN}}^{(2,1)} \circ \mathrm{L}^{(2,1)} \left(\text{concat}\left(\boldsymbol{g}_i, \boldsymbol{h}_i^{(1)}\right)\right), \\
&\boldsymbol{T}_i = \mathrm{L}^{(3,2)} \circ \mathrm{G}_{\mathrm{BN}}^{(3,1)} \circ \mathrm{L}^{(3,1)} \left(\text{concat}\left(\boldsymbol{h}_i^{(1)}, \boldsymbol{h}_i^{(2)}\right)\right), \\
&T(x_j, t_i) \equiv (\boldsymbol{T}_i)_j,
\end{aligned} \tag{3.6}$$

where $\circ$ denotes function composition of layers. The L denotes dense linear layer, and the G_{BN} denotes a function combining the batch normalization operator [22] and the Gaussian

Error Linear Unit activation function [19]. The superscript (i, j) represent the j-th layer in the i-th cell. The linear layer $\mathrm{L}^{(1,1)}$ doubles the dimension of the input vector, i.e. the output size is $2N_h$. The output dimensions of the linear layers $\mathrm{L}^{(1,2)}$, $\mathrm{L}^{(2,1)}$ are $4N_h$ and $6N_h$ respectively, while $\mathrm{L}^{(2,2)}$, $\mathrm{L}^{(2,3)}$ and $\mathrm{L}^{(3,1)}$ are $8N_h$. The output of the final linear layer $\mathrm{L}^{(3,2)}$ is M. The j-th element of the output vector of the third layer, $(\boldsymbol{T}_i)_j$, gives the solution at the j-th spacial discretization point, i.e. $T(x_j, t_i)$.

3.2 Solving the inverse problem

Given a target solution, the inverse problem seeks for a boundary condition $\hat{T}_{bd}$ that minimizes the difference between the solution $\mathcal{F}[\hat{T}_{bd}]$ and the target T^*, i.e. Eq. (2.22). We replace the forward problem solver $\mathcal{F}$ with the surrogate model $\mathcal{F}^s$, and introduce an approximation equation (2.25) to the original inverse problem. The benefit of using the surrogate model is two-fold: (1) the surrogate model is much faster than solving the forward problem, and (2) the surrogate model is differentiable, i.e., the gradient of the MSE concerning the boundary condition

$$\nabla T_{bd} = \frac{\partial \mathcal{L}(\mathcal{F}^{\mathrm{s}}(T_{bd}), T^*)}{\partial T_{bd}} \tag{3.7}$$

is easily calculated by the backward propagation technique [41]. Therefore, the optimization problem (2.25) is solved by gradient-based optimization algorithms. In this work, we find that the simplest steepest descent approach, i.e.

$$T_{bd}^{new} = T_{bd}^{old} - \alpha \nabla T_{bd}, \tag{3.8}$$

can efficiently determine reasonably accurate solutions of inverse problems. In Eq. (3.8), the parameter α is a fixed descent step-size, or learning-rate. We set $\alpha = 500$ in experiments.

4 Experiments

In this section, we omit the unit of the physical quantities. It is noted that when we mention the values of the spatial variable x, temporal variable t, the temperature $T(x, t)$, the absorption coefficient $\sigma_{a,0}$, their units are cm, ns, keV/k_{B} and cm^{-1}, respectively.

4.1 Data preparation

Throughout this work, the time domain $[0, 1]$ is discretized with a time step of $\Delta t = 0.01$, thus we have in total $N = 101$ temporal discretization points. The spatial domain is chosen as $[0, 1]$, and is evenly discretized with a step of $\Delta x = 0.0025$, thus we have $M = 401$ discretization points.

We use three methods, the delta, the constant and the piecewise constant method to generate a sample of discretized boundary conditions for the Marshak wave problem.

We generate 2500 boundary conditions by the delta method, 2000 of which serve as the training data while the other 500 as the validation data. As two test sets, we generate 500 boundaries by the constant method and 500 boundaries by the piecewise constant method.

Delta method. The increment of the T_{bd} at each time step is randomly generated, i.e.

$$\begin{aligned} &T_{bd}(t_0) \sim U(0.2, 2), \\ &T_{bd}(t_{i+1}) = T_{bd}(t_i) + \delta_i, \quad \delta_i \sim U(-0.1, 0.1), \quad i = 0, 1, \ldots, N-1, \end{aligned} \tag{4.1}$$

where $U(a, b)$ denotes the uniform distribution on the interval (a, b). The material temperature T should not be larger than 0. If any $T_{bd}(t_i)$ in a sample is less than 0, the sample is rejected and a new trail sample will be generated.

Constant method. The boundary condition is a random constant, i.e.

$$T_{bd}(t_0) \sim U(0.2, 2), \quad T_{bd}(t_i) = T_{bd}(t_0), \quad i = 1, \ldots, N. \tag{4.2}$$

Piecewise constant method. The time interval is equally divided into 5 sub-intervals. The material temperature T is chosen as a random constant on each of the sub-interval, i.e. for $k = 0, 1, \ldots, 4$,

$$\begin{aligned} &T(t_{20k}) \sim U\big(1.1 - \sqrt{0.33}, 1.1 + \sqrt{0.33}\big), \\ &T_{bd}(t_i) = T_{bd}(t_{20k}), \quad i = 20k+1, \ldots, 20k+19, \\ &T_{bd}(t_{100}) = T_{bd}(t_{80}). \end{aligned} \tag{4.3}$$

We employ both the RTE and its diffusion approximation as the radiative transfer model, and numerically solve the Marshak wave problem with boundaries generated by the three methods.

The boundary conditions and the corresponding numerical Marshak wave solutions together form the dataset of this work. We use the delta method to generate the training and validation datasets. The constant and piecewise constant methods are used to generate two test datasets.

4.2 The training of the surrogate model

The MSE of the RADL surrogate model prediction with respect to the numerical solution is taken as the loss function $\mathcal{L}$, i.e.

$$\mathcal{L} = \frac{1}{|\mathcal{B}|} \sum_{T_{bd} \in \mathcal{B}} \frac{1}{MN} \sum_{j=1}^{M} \sum_{i=1}^{N} \left| \mathcal{F}^s \left[T_{bd}\right](x_j, t_i) - \mathcal{N}\left[T_{bd}\right](x_j, t_i) \right|^2, \tag{4.4}$$

where $\mathcal{N}[T_{bd}]$ denotes the numerical solution of the Marshak wave problem given the boundary condition T_{bd}, $\mathcal{B}$ denotes a mini-batch of training data, $|\mathcal{B}|$ denotes the batch-size.

We take two extra criteria to evaluate the error of the model prediction, i.e. the L^2 error ($\mathcal{E}_2$ with a proper normalization) and the relative L^2 error (ε_2), whose definition are

$$\mathcal{E}_2 = \left[\frac{1}{|\mathcal{S}|} \sum_{T_{bd} \in \mathcal{S}} \frac{1}{MN} \sum_{j=1}^{M} \sum_{i=1}^{N} \left| \mathcal{F}^s \left[T_{bd} \right] (x_j, t_i) - \mathcal{N} \left[T_{bd} \right] (x_j, t_i) \right|^2 \right]^{\frac{1}{2}}, \tag{4.5}$$

$$\varepsilon_2 = \frac{\left[\sum_{T_{bd} \in \mathcal{S}} \sum_{j=1}^{M} \sum_{i=1}^{N} \left| \mathcal{F}^s \left[T_{bd} \right] (x_j, t_i) - \mathcal{N} \left[T_{bd} \right] (x_j, t_i) \right|^2 \right]^{\frac{1}{2}}}{\left[\sum_{T_{bd} \in \mathcal{S}} \sum_{j=1}^{M} \sum_{i=1}^{N} \left| \mathcal{N} \left[T_{bd} \right] (x_j, t_i) \right|^2 \right]^{\frac{1}{2}}}, \tag{4.6}$$

where $\mathcal{S}$ represents the data set on which we calculate the errors. $|\mathcal{S}|$ denotes the number of samples in $\mathcal{S}$.

The stochastic gradient descent optimizer Adam [25] is used to train the model parameters of RADL. As for the optimizer, we use an initial learning rate of 10^{-4} with a batch size of 100 to minimize the loss function. Besides, we use a learning-rate decay method to assist model training. We empirically find that the cosine annealing decay [28] strategy outperforms other learning rate decaying schemes. The idea of cosine annealing decay strategy is to jump out of the local minima by periodically increasing the learning rate once $\tau_{\max}$ epochs are performed. We compute the learning rate for each epoch by

$$\eta(\tau) = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min}) \left(1 + \cos \left((\tau \mod \tau_{\max}) \cdot \pi \right) \right), \tag{4.7}$$

where mod is modulo operation, τ is the current number of epochs since the last restart, $\tau_{\max}$ is a hyper-parameter, $\eta(\tau)$ is the learning rate at epoch τ, $\eta_{\max}$ and $\eta_{\min}$ denote the maximum and the minimum learning rate, respectively.

Additionally, we compare the performance of the Adam and L-BFGS optimizers in the case of surrogate RTE model $\sigma_{a,0} = 30$ (see details in Section 4.3). Upon convergence, the absolute error on the validation set using Adam as the optimizer was 0.0240 and the relative error was 0.0219, while the absolute error using L-BFGS as the optimizer was 0.0268 and the relative error was 0.0239. We conducted five independently repeated experiments to estimate the number of epochs needed to achieve convergence. The Adam optimizer, on average, convergences in 2588 epochs, while the L-BFGS optimizer could achieve convergence in 2440 epochs on average. Nonetheless, when the network representation and the training samples were adequate, we found the difference between the two optimizers to be not critical.

All of the following experiments adopt the same model structure and hyper-parameters. The parameters configuration during the model training process is as follows. We initialize the network parameters by the uniform Xavier method [16]. In cosine decay strategy, $\eta_{\max} = 10^{-4}, \eta_{\min} = 10^{-7}, \tau_{\max} = 10$. In our model training, we apply an early stopping mechanism (with patience of 300 epochs) to prevent overfitting. α in solving inverse problem (Section 3.2) is 500.

4.3 Surrogate for the Marshak wave problem

For the Marshak wave problem, we consider two absorption constants with $\sigma_{a,0} = 30$ and $\sigma_{a,0} = 150$, which cover the typical range of material opacity in real-world applications. The surrogate models for the two cases are trained by the delta method (4.1). The labels for case $\sigma_{a,0} = 30$ and $\sigma_{a,0} = 150$ are separately generated by the numerical scheme described in Section 2.3. The training protocols are detailed in Section 4.2. During the training, the models are validated against a validation dataset generated by the delta method (4.1). When the training finishes, the models are tested with datasets generated by the constant (4.2) and piecewise constant (4.3) methods, and the errors are evaluated in the L^2 (4.5) and relative L^2 (4.6) senses.

The accuracy of the surrogate model for solving the RTE is presented in Table 4.1. We find that both the absolute $\mathcal{E}_2$ error and the relative ε_2 errors on the validation set and two test sets all achieve a magnitude of 10^{-2}. The constant test error is close to the validation error, while the piecewise constant test error is roughly twice as large as the validation error. The high accuracy demonstrates that the surrogate model preserves high confidence within or out-of-the-training sample space, thus presenting a considerable well-fitting ability and generalization ability. Furthermore, for the validation set, we calculate the, $\mathcal{E}_2$ and ε_2 error at different time (see Table 4.2). The results illustrate that the surrogate model is uniformly accurate in the temporal domain.

Table 4.1: The absolute ($\mathcal{E}_2$) and the relative (ε_2) errors in the L^2 sense of the surrogate model for the RTE and its DA. The errors on the validation, constant test and piecewise constant test datasets are presented. Absorption coefficients with $\sigma_{a,0} = 30$ and 150 are considered.

Model	Data	$\sigma_{a,0} = 30$		$\sigma_{a,0} = 150$	
		$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2
Surrogate RTE	Validation	0.0240	0.0219	0.0285	0.0291
	Constant test	0.0241	0.0253	0.0222	0.0287
	Piecewise constant test	0.0582	0.0649	0.0423	0.0623
Surrogate DA	Validation	0.0360	0.0820	0.0461	0.0816
	Constant test	0.0881	0.1267	0.0522	0.1235
	Piecewise constant test	0.0640	0.1343	0.0423	0.1536

Table 4.2: The validation error of the RTE surrogate model at different time. The absorption coefficient satisfies $\sigma_{a,0} = 30$.

t	$\sigma_{a,0} = 30$		$\sigma_{a,0} = 150$	
	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2
0.2	0.0261	0.0291	0.0321	0.0422
0.4	0.0242	0.0222	0.0287	0.0289
0.6	0.0267	0.0222	0.0262	0.0249
0.8	0.0301	0.0235	0.0288	0.0240
1.0	0.0362	0.0277	0.0301	0.0253

The surrogate model for the DA is less accurate than that for the RTE (see Table 4.1): The absolute error is around 0.05 and the relative error ε_2 is lower than 0.15. Interestingly, the constant test error of the RTE surrogate is closer to the validation error, while it is closer to the piecewise constant error in the case of DA surrogate model. The surrogate model predicted material temperature $T(x,t) = \mathcal{F}^s[T_{bd}](x,t)$ of a piecewise constant boundary condition is graphically presented in Fig. 4.1, and is compared with the reference solution $\mathcal{N}[T_{bd}]$. The surrogate model is of an absolute error of 0.0640 and a relative error of 0.1343, and is able to accurately capture the wave front location and the magnitude of the material energy field.

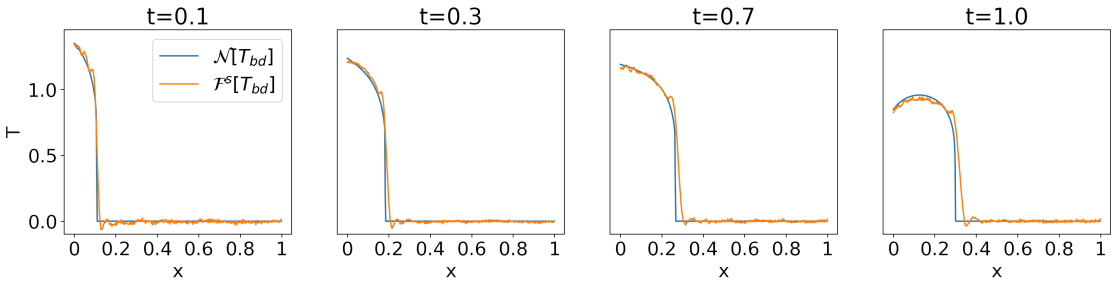

Figure 4.1: Comparison between the numerical solution ($\mathcal{N}[T_{bd}]$) and the DA surrogate model prediction ($\mathcal{F}^s[T_{bd}]$) on a sample from the piecewise constant test set. The absolute and relative errors of the sample are $\mathcal{E}_2 = 0.0755$ and $\varepsilon_2 = 0.1533$, respectively. The absorption coefficient satisfies $\sigma_{a,0} = 30$.

4.4 Inverse problem for Marshak wave problem

To investigate how accurately the inverse problems are solved, we set the numerical solutions of the boundary conditions generated by the delta, constant and piecewise constant methods as target solutions. The approximated inverse problem (2.25) is then solved by the steepest descent approach (3.8), and the solution is denoted by $\hat{T}^s_{bd}$. The accuracy $\hat{T}^s_{bd}$ is measured by the absolute and relative L^2 errors between the numerically solved $\mathcal{N}[\hat{T}^s_{bd}]$ and the target.

The errors of the inverse RTE and DA are presented in Table 4.3. The errors of the inverse RTE problem on all datasets and those of the DA on the constant and piecewise constant test datasets are only slightly larger than the corresponding errors of the surrogate model (compare Table 4.1 and Table 4.3), which means the inverse problem is solved at almost the best accuracy one may expect. We notice that the error of the inverse DA on the validation set is significantly larger than that of the surrogate DA validated against the same dataset. The reason for the phenomenon is still not clear to us. We sample one target solution from the DA validation set ($\sigma_{a,0} = 150$) and visualize the numerical solution of the inverse boundary in Fig. 4.2. This sample is chosen because its absolute and the relative L^2 error are 0.1001 and 0.4441, respectively, and are comparable to the average errors of the whole validation dataset in Table 4.3. It is observed that the numerical solution of the inverse boundary can accurately capture the wave front location of the target solution, thus the accuracy is acceptable.

Table 4.3: The absolute ($\mathcal{E}_2$) and the relative (ε_2) errors of the numerical solution of the inverse RTE and DA boundaries. The errors on the validation, constant test and piecewise constant test datasets are presented. Absorption coefficients with $\sigma_{a,0} = 30$ and 50 are considered.

Problem	Data	$\sigma_{a,0} = 30$		$\sigma_{a,0} = 150$	
		$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2
RTE	Validation	0.0280	0.0265	0.0263	0.0278
	Constant test	0.0221	0.0229	0.0260	0.0332
	Piecewise constant test	0.0660	0.0762	0.0660	0.0796
DA	Validation	0.1280	0.1857	0.1580	0.3114
	Constant test	0.1006	0.1506	0.0623	0.1486
	Piecewise constant test	0.0721	0.1511	0.0480	0.1509

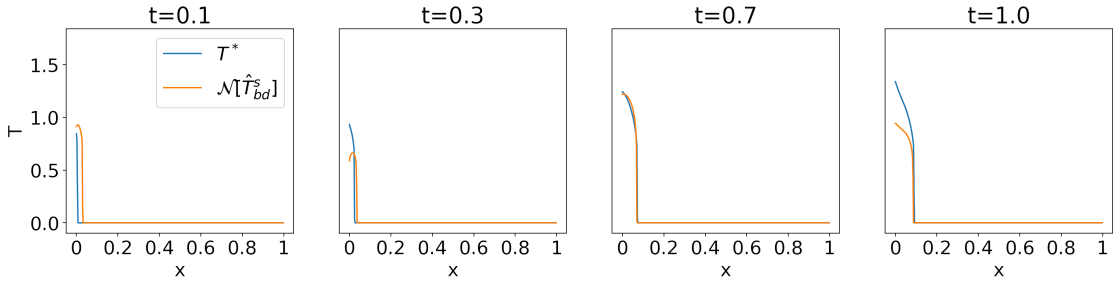

Figure 4.2: Comparison between the numerical solution of the inverse DA boundary ($\mathcal{N}[\hat{T}^s_{bd}]$) and the target solution (T^*) sampled from the validation set. The absolute and relative errors of the sample are $\mathcal{E}_2 = 0.1001$ and $\varepsilon_2 = 0.4441$, respectively. The absorption coefficient satisfy $\sigma_{a,0} = 150$.

4.5 Discussions

4.5.1 Ablation study on the network architecture

We investigate the role of the factor-attention and RNN blocks on the accuracy of the surrogate models through an ablation study. We respectively prune the RNN block (denoted as "noRNN"), the factor-attention block (denoted as "noAtt") and both blocks (denoted as "noBoth") from the model, and test the accuracy of the RTE surrogate model against the validation, constant and piecewise constant test datasets.

The absolute and relative L^2 errors ($\mathcal{E}_2$ and ε_2) are presented in Table. 4.4. Not surprisingly, the "noBoth" model presents errors that are almost an order of magnitude larger than the complete model architecture. The poor fitting and generalization performance is attributed to the lack of an effective temporal feature characterization. The "noRNN" model that only uses the factor-attention block for extracting features seems to miss some critical time information, thus it gives nearly the same accuracy as the "noBoth" model. The "noAtt" model, using the RNN block for featurization, is significantly more accurate than the "noRNN" model, but is still notably less accurate than the complete model structure. The ablation study proves that both the factor-attention and the RNN blocks are crucial for extracting the temporal feature from the boundary condition, thus are indispensable for the high accuracy of the surrogate models.

Table 4.4: The ablation study of the RADL model architecture. The errors of the surrogate RTE on the validation, the constant test and the piecewise constant test sets are presented. The RNN block ("noRNN"), the factor-attention block ("noAtt") or both of them ("noBoth") are pruned. The absorption coefficient satisfies $\sigma_{a,0} = 30$.

Data	noRNN		noAtt		noBoth		complete	
	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2
Validation	0.2740	0.2514	0.0412	0.0455	0.2740	0.2495	0.0180	0.0180
Constant test	0.1880	0.1961	0.0398	0.0401	0.1581	0.1654	0.0220	0.0224
Piecewise constant test	0.4700	0.5219	0.0524	0.0520	0.4540	0.5041	0.0340	0.0391

4.5.2 Estimation for the model uncertainty

In this section, we take the bootstrap method [14] to estimate the model uncertainty with the effect on the training size. The bootstrap method is a type of re-sampling where large numbers of smaller samples of the same size are repeatedly drawn, with replacement, from a single original sample. In our experiment, we prepare a training dataset, denoted by $\mathcal{D}$, that has 5000 boundary conditions generated by the delta method (4.1), and the corresponding RTE solution is prepared by the numerical scheme. We use the bootstrap method to randomly draw a training dataset $\mathcal{D}_k$ of size $|\mathcal{D}_k|$ from the dataset $\mathcal{D}$, and then train a surrogate model by the dataset $\mathcal{D}_k$. The error of the surrogate model is evaluated on the validation dataset and the test dataset generated by the constant method (4.2). The above procedure is repeated 5 times. The model error and the uncertainty in the error are estimated by the average and the standard deviation of the 5 model errors.

We visualize the accuracy and uncertainty of the RADL surrogate model at different training sample sizes in Fig. 4.3. It is observed from the figure that the error reduces as the

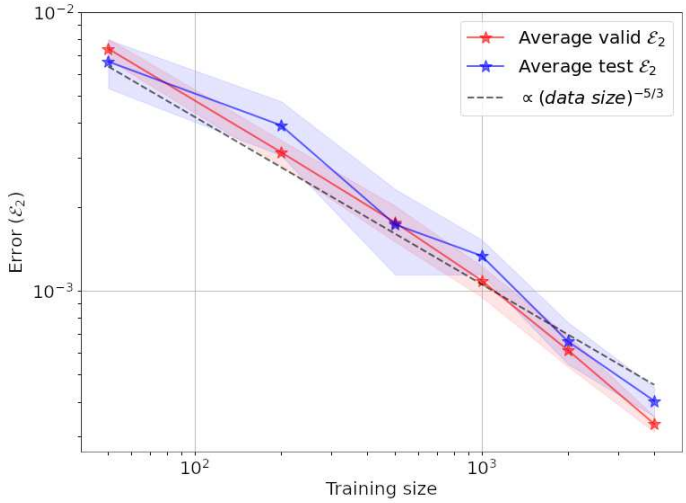

Figure 4.3: The uncertainty of the RTE surrogate model versus the size of the training data. The absorption coefficient satisfy $\sigma_{a,0} = 30$.

training data size increase up to the largest training data size of 4000 investigated in the work. The speed of the error reduction fits almost perfectly to the $-5/3$-th power of the size of the training data set (see the black dashed line in Fig. 4.3).

With the increase of the training size, the fluctuation range of errors on the validation set and test set is narrowing (noticing that the error axis in the Fig. 4.3 is log-scaled). This proves that the robustness of the surrogate model strengthens as the training size increases. Besides, the model uncertainty on the validation set is significantly smaller than that on the constant test set.

4.5.3 Comparative experiments with other methods

This section presents a comparative analysis of the proposed method with two commonly used approaches for operator approximation using neural networks: DeepONet [29] and the Fourier neural operator (FNO) [27]. We control the numbers of network parameters to be comparable and evaluate the accuracy of the models using the absolute ($\mathcal{E}_2$) and relative (ε_2) errors as the metrics. The errors of the three methods in the spatial and temporal domain are compared in Table 4.5. In our experiment, FNO contains 4 layers of the Fourier layer. We set the parameters for all Fourier layers as follows: the maximum mode $k_{\max}$ is set to 16, the dimension of the Fourier layer d is set to 64, other configurations are set the same as the default ones provided in the FNO source code [27]. Regarding the DeepONet, we set the depth of the trunk network to be 4 with [128, 256, 256, 400] neurons in each layer, respectively, and set the depth of the branch network to be 7 with [512, 512, 512, 512, 256, 128, 400] neurons in each layer, respectively. We employ the delta method to generate 4000 boundary conditions as the training set, and 500 as the validation. We generate 500 boundaries by the constant method and 500 boundaries by the piecewise constant method as two test sets. All other hyper-parameters in both DeepONet and FNO are kept consistent with those of RADL aforementioned.

Table 4.5 reveals that RADL performed the best on both the validation and test sets, followed by FNO. Notably, both RADL and FNO reached convergence after 3000 training iterations. On the other hand, the training results of DeepONet, as displayed in the table, were obtained after an extensive 1.2 million iterations. While further training would likely result in a decline in the errors, it is evident that DeepONet's performance is notably inferior to that of RADL and FNO. Therefore, we opted not to continue training DeepONet any further.

Table 4.5: Comparative Analysis of RADL, DeepONet, and FNO. The absorption coefficient satisfies $\sigma_{a,0} = 30$.

Data	RADL		FNO		DeepONet	
	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2
Validation	0.0180	0.0180	0.0273	0.0249	0.0758	0.0696
Constant test	0.0220	0.0224	0.0254	0.0263	0.0721	0.0747
Piecewise constant test	0.0340	0.0391	0.0453	0.0503	0.1265	0.1405

4.5.4 Surrogate model for anisotropic radiative transfer in a slab

This section further confirms the accuracy of the proposed RADL model for problems involving anisotropy. We consider the radiation problem in a plane-parallel slab with an anisotropic scattering,

$$\frac{1}{c}\frac{\partial I}{\partial t} + \mu\frac{\partial I}{\partial x} = -\sigma_s I + \sigma_s \int_{-1}^{1} p(\mu,\mu')I(x,\mu')\mathrm{d}\mu'. \tag{4.8}$$

In Eq. (4.8), σ_s is a constant coefficient taken to be 100. The normalized scattering kernel $p(\mu,\mu')$ is described by an approximation to the Henyey-Greenstein phase function as discussed in [1],

$$p(\mu,\mu') = \sum_{l=0}^{\infty} \frac{2l+1}{2} g^l P_l(\mu) P_l(\mu'), \tag{4.9}$$

where g is the anisotropic coefficient and P_l is the Legendre polynomial of order l. The initial and boundary conditions the specific intensity I subject to are given in Eqs. (2.17) and (2.18), respectively. The forward problem (2.21) is replaced by

$$E = \frac{1}{c}\mathcal{F}[T_{bd}], \tag{4.10}$$

where E is the energy density as defined in (2.11). The surrogate model is for the RTE, in the scenario of a forward-peaked scattering with $g = 0.9$. During the training phase, the labels were generated via solving the P_{11} system by the finite volume method. The training hyper-parameters were kept the same as the isotropic scattering cases.

The validation and test errors of the surrogate model for the RTE with anisotropic scattering are reported in Table 4.6. It is observed that the absolute and relative errors reach an order of magnitude of 10^{-2}, across the validation, constant test, and piecewise constant test cases. Notably, the errors on the constant test are even slightly lower than those on the validation set. This proves that the proposed RADL handles the anisotropic scattering cases as well as the isotropic cases. We visualize one sample in the piecewise constant test set in Fig. 4.4. We select this sample because its errors ($\mathcal{E}_2 = 0.0236$ and $\varepsilon_2 = 0.0770$) are

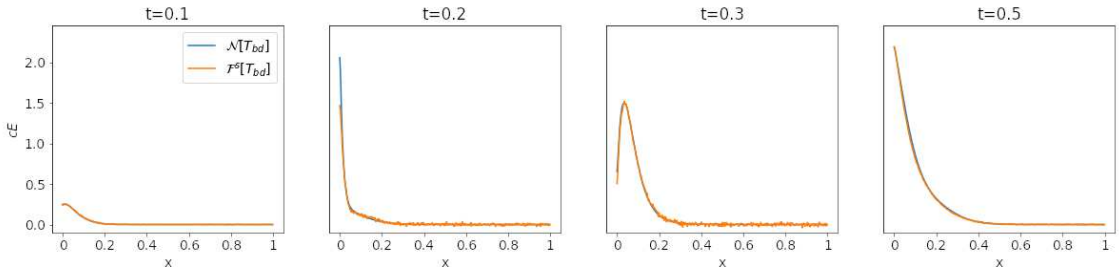

Figure 4.4: Comparison between the numerical solution ($\mathcal{N}[T_{bd}]$) and the surrogate model prediction ($\mathcal{F}^s[T_{bd}]$) for the RTE with anisotropic scattering on a sample from the piecewise constant test. The absolute and relative errors of the sample are $\mathcal{E}_2 = 0.0236$ and $\varepsilon_2 = 0.0770$, respectively. The scattering cross-section satisfies $\sigma_s = 100$ and $g = 0.9$.

Table 4.6: The absolute ($\mathcal{E}_2$) and the relative (ε_2) errors in the L^2 sense of the surrogate model for the RTE with anisotropic scattering. The errors on the validation, constant test and piecewise constant test are presented. The scattering cross-section satisfies $\sigma_s = 100$ and $g = 0.9$.

Data/Metrics	$\mathcal{E}_2$	ε_2
Validation	0.0116	0.0133
Constant test	0.0078	0.0127
Piecewise constant test	0.0224	0.0769

similar to the average errors of the entire test set. Despite the model's highest errors on the piecewise constant test set, as evidenced by the visualization in Fig. 4.4, its performance remains remarkably good, which validates the high generalization capacity of the proposed model. Importantly, the accuracy in simulating the RTE with anisotropic scattering is maintained without modifications to the network structure, any increase in the network parameter quantity and complexity, or adjustments to any hyper-parameters.

4.5.5 Impact of the spatial sampling

This section demonstrates the reliance of the model on different spatial sampling rates. In this section, we use the delta method to generate 4000 boundary conditions for the training set and 500 for the validation set. We generate two test sets: one with 500 boundaries using the constant method, and another with 500 boundaries using the piecewise constant method. The training data was subsampled in the spatial dimension at rates of 50%, 25%, and 10%. The efficacy of the surrogate models was evaluated at these rates and compared against non-subsampled models, providing a comprehensive analysis of the impact of subsampling on the validation and testing errors.

Table 4.7 leads to the conclusion that the surrogate model yields results that are nearly indistinguishable from those of non-subsampled models when the subsample rate is 50%. At a rate of 25%, there is a slight increment in the model error. When the rate drops below 25%, the piecewise constant test set exhibits a noticeable increase in the errors, while the validation set and constant test sets are still retaining a relatively high accuracy. Our analysis suggests that the model is robust to the spatial subsampling of training data of rate $\geq$ 50%. As the subsample rate further decreases, there is a gradual increase in the validation and testing errors, indicating a dependence on the sampling rate.

Table 4.7: Discretization analysis of RTE surrogate models at various spatial subsampling rates. The absorption coefficient satisfies $\sigma_{a,0} = 30$, r is the subsampling rate.

Data	$r = 100\%$		$r = 50\%$		$r = 25\%$		$r = 10\%$	
	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2	$\mathcal{E}_2$	ε_2
Validation	0.0180	0.0180	0.0192	0.0175	0.0210	0.0212	0.0224	0.0204
Constant test	0.0220	0.0224	0.0201	0.0201	0.0228	0.0234	0.0256	0.0255
Piecewise constant test	0.0340	0.0391	0.0354	0.0390	0.0399	0.0421	0.0596	0.0658

4.5.6 Surrogate model for a two-dimensional RTE

We construct a surrogate model for the two-dimensional (2D) RTE hohlraum benchmark problem, following the layout discussed in literature [26, 32]. For this problem, the inflow boundary condition is imposed at $x = 0$. The boundary condition is assumed to be isotropic in the photon's direction, equilibrious in frequency and invariant in space, but is varying in time. We evaluated the performance of our surrogate model using the $\mathcal{E}_2$ and ε_2 metrics. The labels are generated by solving the P_5 system with the finite volume method on a 100×100 2D mesh. The size of the training set, validation, and test sets remains unchanged. We use the same hyper-parameters and network architecture as the 1D problem, with the exception of modifying the final layer of the fitting block to 10,000 neurons to predict the solution on the 2D mesh.

We train a surrogate model for the 2D RTE and evaluate the model performance on the validation, constant test and the piecewise constant test by the absolute L^2 error $\mathcal{E}_2$ and the relative L^2 error ε_2. As Table 4.8 shows, the errors on the constant test are slightly higher than those on the validation, while the absolute and the relative errors on the piecewise constant test are 0.0288 and 0.1353 respectively. We visualize the true and predicted solutions of a randomly selected sample from the piecewise constant test at distinct times in Fig 4.5.

Table 4.8: The absolute ($\mathcal{E}_2$) and the relative (ε_2) errors in the L^2 sense of the surrogate model for 2D RTE. The errors on the validation, constant test and piecewise constant test datasets are presented. The absorption coefficient satisfies $\sigma_{a,0} = 30$.

Data/Metrics	$\mathcal{E}_2$	ε_2
Validation	0.0097	0.0205
Constant test	0.0103	0.0345
Piecewise constant test	0.0288	0.1353

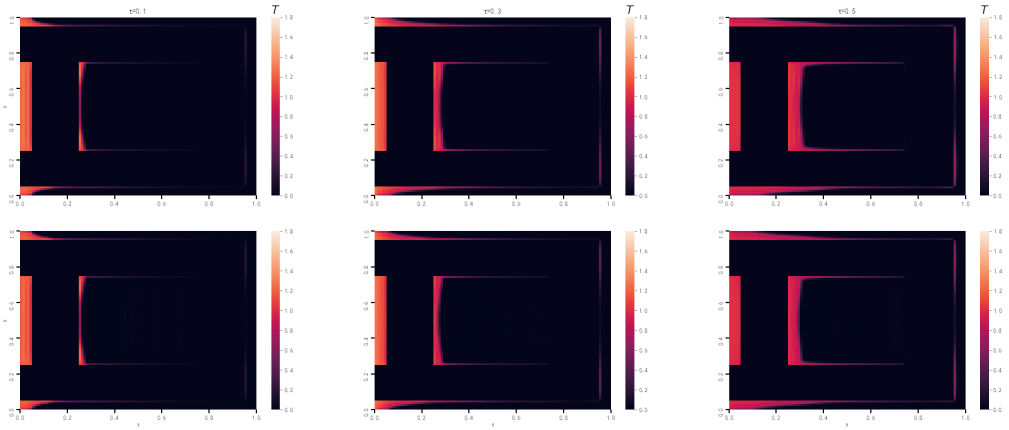

Figure 4.5: Surrogate model performance for 2D RTE hohlraum benchmark problem.The absolute and relative errors of the sample are $\mathcal{E}_2 = 0.0326$ and $\varepsilon_2 = 0.1343$, respectively.

The example presented in this section demonstrates the efficacy of our surrogate model for the 2D scenario. It is worth noting that we did not account for anisotropic scattering, frequency-dependent nor spatially inhomogeneous sources. Such challenging cases are left for future investigations.

5 Conclusion

In this paper, we propose a new surrogate model structure, termed RADL, to solve the inverse boundary problem of the Marshak wave problem for both the radiative transfer equation and its diffusion approximation. We demonstrate the effectiveness of our approach by examples of solving the forward and inverse Marshak wave problems with absorption coefficients satisfying $\sigma_{a,0} = 30$ and 150. We have shown that the absolute and relative errors in the L^2 sense reach an order of 10^{-2} for both the forward and inverse problems of the non-equilibrium Marshak wave radiative transport. The accuracy in solving the Marshak diffusion problem is lower than that of solving the RTE, but still reaches $\sim 10^{-1}$, which is acceptable.

By an ablation study, we argue that both the RNN and the factor-attention structures are indispensable for the accuracy of the RADL. By numerical examples, we demonstrate the effectiveness of the proposed RADL surrogate model in solving problems of 1D RTE with anisotropic scattering and 2D RTE hohlraum benchmark problem. Importantly, the high accuracy of the RADL model in these cases is achieved without modification to the model structure and training hyper-parameters. We conduct a comparative study of the RADL approach with two existing operator learning methods, i.e. DeepONet and FNO that does not consider the source and solution of the RTE as time series, and show that RADL outperforms both DeepONet and FNO in terms of accuracy when the numbers of model parameters are comparable.

In the future, the RADL will be extended to model the forward solver at different absorption constant $\sigma_{a,0}$ values by one model. The model takes the $\sigma_{a,0}$ and other parameters as input and is able to predict the solution given a boundary condition for a group of RTEs with different absorption parameters. The forward surrogate model for solving inverse boundary problems of higher dimensional RTEs and their DA is still an open question and worth investigating in the future.

Acknowledgments

The work is supported by the National Science Foundation of China (Grant Nos.11871110, 12122103, 12271050, 12031001, 12001051) and by the Key Laboratory of Computational Physics Foundation (Grant Nos. 6142A05210502, 6142A05220501, 6142A05RW202206).

Appendix A

Optionally include extra information (complete proofs, additional experiments and plots) in the appendix. This section will often be part of the supplemental material.

A.1 The architecture of attention layer

The attention layer A is a multi-head attention mechanism [47] . The multi-head attention mechanism (Fig. A.1) is a powerful feature extractor. It is equivalent to determining which information each element should pay attention to according to the similarity between sequence elements. In this work we employ $h = 16$ parallel attention heads. For each head, the input $\{g_i^f\}, i = 1, \ldots, N$ is firstly transformed into query, key, and value by a linear mapping. We obtain the attention score by the following formula:

$$\begin{aligned} Q &= \left\{g_i^f\right\} W^Q, \\ K &= \left\{g_i^f\right\} W^K, \\ V &= \left\{g_i^f\right\} W^V, \\ \text{Attention}(Q, K, V) &= \text{softmax}\left(\frac{QK^T}{\sqrt{N_h}}\right) V, \end{aligned} \tag{A.1}$$

where $W^Q \in \mathbb{R}^{N_h \times N_h}, W^K \in \mathbb{R}^{N_h \times N_h}, W^V \in \mathbb{R}^{N_h \times N_h}$ are the trainable parameters. Multi-head attention allows the model to ensemble the information from different representation subspaces and each head to focus on its own key point. We compute the attention layer A

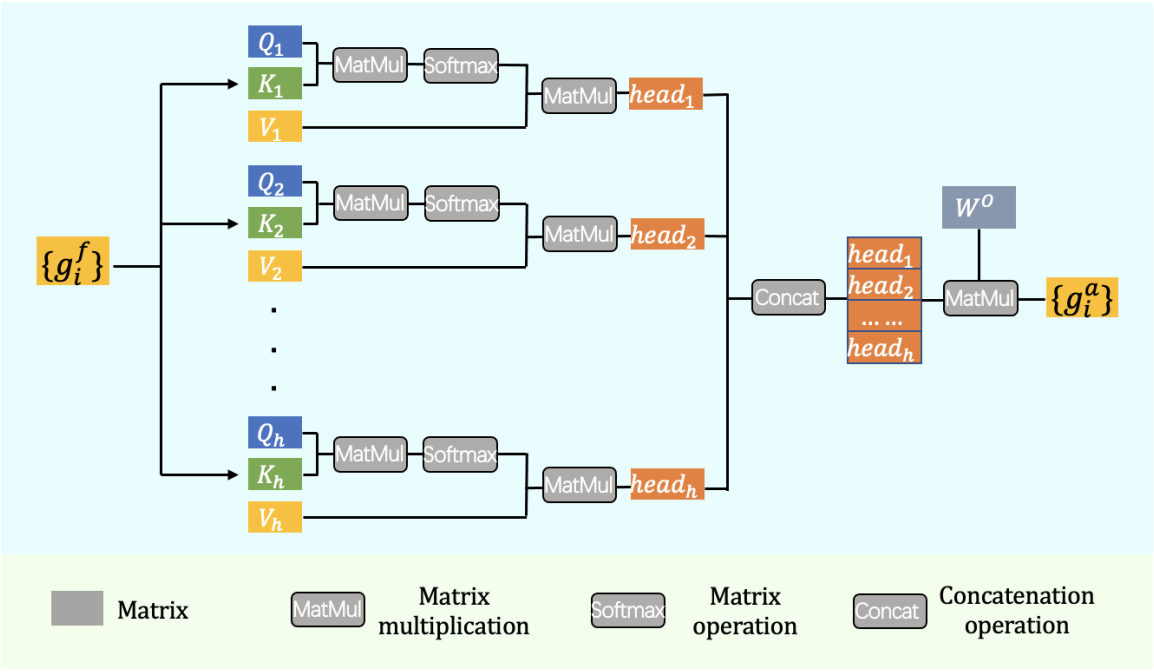

Figure A.1: Multi-head attention.

output $\{g_i^a\}$ as

$$\{g_i^a\} = A\left(\left\{g_i^f\right\}\right) = \text{Concat}\left(\text{head}_1, \ldots, \text{head}_\text{h}\right) W^O, \tag{A.2}$$

where

$$\text{head}_i = \text{Attention}(Q_i, K_i, V_i) = \text{Attention}\left(\left\{g_i^f\right\} W_i^Q, \left\{g_i^f\right\} W_i^K, \left\{g_i^f\right\} W_i^V\right),$$

$W_i^Q \in \mathbb{R}^{N_h \times N_h}$, $W_i^K \in \mathbb{R}^{N_h \times N_h}$, $W_i^V \in \mathbb{R}^{N_h \times N_h}$, $W^O \in \mathbb{R}^{(N_h \times h) \times N_h}$, $i \in 1, \ldots, h$.

References

[1] O. Akdemir, A. Lagendijk, and W. L. Vos, Breakdown of light transport models in photonic scattering slabs with strong absorption and anisotropy, *Phys. Rev. A*, **105**(3):033517, 2022.

[2] Y. An, X. Yan, W. Lu, H. Qian, and Z. Zhang, An improved bayesian approach linked to a surrogate model for identifying groundwater pollution sources, *Hydrogeol. J.*, **30**(2):601–616, 2022.

[3] A. Anshuman and T. I. Eldho, Entity aware sequence to sequence learning using lstms for estimation of groundwater contamination release history and transport parameters. *J. Hydrol.*, **608**:127662, 2022.

[4] V. C. Badham, E. W. Larsen, and G. C. Pomraning, Asymptotic analysis of radiative transfer problems, *J. Quant. Spectrosc. Radiat. Transf.*, **29**:285, 1983.

[5] A. R. Barron, Universal approximation bounds for superpositions of a sigmoidal function, *IEEE Trans. Inf. Theory*, **39**(3):930–945, 1993.

[6] T. Brown et al., Language models are few-shot learners, *Adv. Neural Inf. Process Syst.*, **33**:1877–1901, 2020.

[7] G. A. Carpenter and S. J. Grossberg, Cybernetic approach to pattern recognition and learning, *Internat. J. Systems Sci.*, **18**(7):493–509, 1987.

[8] D. S. Clark et al., Three-dimensional modeling and hydrodynamic scaling of national ignition facility implosions, *Phys. Plasmas*, **26**(5):050601, 2019.

[9] P. G. Constantine, A. Doostan, Q. Wang, and G. Iaccarino, A surrogate accelerated bayesian inverse analysis of the hyshot ii flight data, *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 2011.

[10] J. D. Densmore and E. W. Larsen, Asymptotic equilibrium diffusion analysis of time-dependent Monte Carlo methods for grey radiative transfer, *J. Comput. Phys.*, **199**(1):175–204, 2004.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv:1810.04805*, 2018.

[12] W. E, C. Ma, and L. Wu, A priori estimates of the population risk for two-layer neural networks, *arXiv:1810.06397*, 2018.

[13] W. E, C. Ma, and L. Wu, The Barron space and the flow-induced function spaces for neural network models, *Constr. Approx.*, **55**(1):369–406, 2022.

[14] B. Efron and R. Tibshirani, Improvements on cross-validation: The .632+ bootstrap method. *J. Am. Stat. Assoc.*, **92**(438):548–560, 1997.

[15] H. Fang, C. Gong, C. Li, X. Li, H. Su, and L. Gu, A surrogate model based nested optimization framework for inverse problem considering interval uncertainty, *Struct. Multidiscip. Optim.*, **58** (3):869–883, 2018.

[16] X. Glorot and Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 249–256. JMLR Workshop and Conference Proceedings, 2010.

[17] A. Graves, S. Fernández, and J. Schmidhuber, Bidirectional lstm networks for improved phoneme classification and recognition. In: Artificial Neural Networks: Formal Models and Their Applications – ICANN 2005, Lecture Notes in Computer Science, Vol. 3697, Springer, 2005.

[18] S. Heidenreich, H. Gross, and M. Bär, Bayesian approach to determine critical dimensions from scatterometric measurements, *Metrologia*, **55**(6):S201, 2018.

[19] D. Hendrycks and K. Gimpel, Gaussian error linear units (gelus), *arXiv:1606.08415*, 2016.
[20] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9**(8):1735–1780, 1997.
[21] K. Hornik, M. Stinchcombe, and H. White, Multilayer feedforward networks are universal approximators, *Neural. Netw.*, **2**(5):359–366, 1989.
[22] S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, In: *International Conference on Machine Learning*, PMLR, 448–456, 2015.
[23] A. G. Irvine, I. D. Boyd, and N. A. Gentile, Reducing the spatial discretization error of thermal emission in implicit Monte Carlo simulations, *J. Comput. Theor. Transp.*, **45**(1-2):99–122, 2016.
[24] M. I. Jordan, Serial order: A parallel distributed processing approach, *ICS-Report 8604 Institute for Cognitive Science University of California*, **121**:64, 1986.
[25] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, *arXiv:1412.6980*, 2014.
[26] W. Li, P. Song, and Y. Wang, An asymptotic-preserving imex method for nonlinear radiative transfer equation, *J. Sci. Comput.*, **92**(27), 2022.
[27] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, Fourier neural operator for parametric partial differential equations, *arXiv:2010.08895*, 2020.
[28] I. Loshchilov and F. Hutter, Sgdr: Stochastic gradient descent with warm restarts, *arXiv:1608.03983*, 2016.
[29] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators, *Nat. Mach. Intell.*, **3**(3):218–229, 2021.
[30] M. M. Marinak, G. D. Kerbel, N. A. Gentile, O. Jones, D. Munro, S. Pollaine, T. R. Dittrich, and S. W. Haan, Three-dimensional hydra simulations of national ignition facility targets, *Phys. Plasmas*, **8**(5):2275–2280, 2001.
[31] Y. M. Marzouk and H. N. Najm, Dimensionality reduction and polynomial chaos acceleration of bayesian inference in inverse problems, *J. Comput. Phys.*, **228**(6):1862–1902, 2009.
[32] R. G. McClarren and C. D. Hauck, Robust and accurate filtered spherical harmonics expansions for radiative transfer, *J. Comput. Phys.*, **229**:5597–5614, 2010.
[33] L. Mieussens, On the asymptotic preserving property of the unified gas kinetic scheme for the diffusion limit of linear kinetic models, *J. Comput. Phys.*, **253**:138–156, 2013.
[34] C. D. Ott, A. Burrows, L. Dessart, and E. Livne, Two-dimensional multiangle, multigroup neutrino radiation- hydrodynamic simulations of postbounce supernova cores, *Astrophys. J.*, **685**:1069–1088, 2008.
[35] S. Peng et al., Lared-integration code for numerical simulation of the whole process of the indirect-drive laser inertial confinement fusion, *High Power Laser and Particle Beams*, **27**(03):54–60, 2015.
[36] F. H. Pereira, P. H. T. Schimit, and F. E. Bezerra, A deep learning based surrogate model for the parameter identification problem in probabilistic cellular automaton epidemic models, *Comput. Methods Programs Biomed.*, **205**:106078, 2021.
[37] M. Peters, M. Neumann, M. Iyyer, M. Gardner, and L. Zettlemoyer, Deep contextualized word representations, In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long Papers)*, 2227–2237, 2018.
[38] G. Poëtte and X. Valentin, A new implicit Monte-Carlo scheme for photonics (without teleportation error and without tilts), *J. Comput. Phys.*, **412**:109405, 2020.
[39] G. C. Pomraning, The non-equilibrium Marshak wave problem, *J Quant Spectrosc Radiat Transf*, **21**(3):249–261, 1979.
[40] G. C. Pomraning, *The Equations of Radiation Hydrodynamics*, Courier Corporation, 2005.
[41] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning representations by back-propagating errors, *Nature*, 323, 1986.
[42] M. Schuster and K. K. Paliwal, Bidirectional recurrent neural networks, *IEEE Trans. Signal Process.*, **45**(11):2673–2681, 1997.
[43] A. Srivastava, Inverse design and deep learning for phononic crystals, *J. Acoust. Soc. Am.*, **146**(4):2828–2828, 2019.
[44] W. Sun, S. Jiang, and K. Xu, An asymptotic preserving unified gas kinetic scheme for gray radiative transfer equations, *J. Comput. Phys.*, **285**:265–279, 2015.
[45] W. Sun, S. Jiang, and K. Xu, An asymptotic preserving implicit unified gas kinetic scheme for frequency-dependent radiative transfer equations, *Int. J. Numer. Anal. Model.*, **15**, 2018.

[46] S. Vakili and M. S. Gadala, Low cost surrogate model based evolutionary optimization solvers for inverse heat conduction problem, *Int. J. Heat Mass Transf.*, **56**(1-2):263–273, 2013.
[47] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, Attention is all you need, *Adv. Neural Inf. Process Syst.*, **30**, 2017.
[48] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, *Adv. Neural Inf. Process Syst.*, **32**, 2019.