

# 基于热传导方程的自适应损失物理信息神经网络算法研究

赵洪玉

青岛大学数学与统计学院, 山东 青岛

收稿日期: 2025年2月19日; 录用日期: 2025年3月12日; 发布日期: 2025年3月18日

## 摘要

在热传导方程的研究中, 物理信息神经网络(PINN)的应用已初显成效, 其损失函数由多个损失项的加权和组成, 这些损失项的加权组合对PINN的有效训练具有关键作用。为此, 我们引入了一个基于高斯概率模型的损失项定义, 通过噪声参数来描述每个损失项的权重, 并提出了一种基于极大似然估计原理的自适应损失函数方法, 该方法通过不断更新每个训练周期中的噪声参数, 实现损失权重的自动分配。采用自适应物理信息神经网络(SalPINN)对一维瞬态热传导方程进行求解, 并与传统PINN方法对比, 结果显示SalPINN在模拟热传导方程方面表现出更高的精确性和有效性。

## 关键词

热传导方程, 物理信息神经网络, 自适应损失平衡法

# Study on Self-Adaptive Loss Physical Information Neural Network Algorithm Based on Heat Transfer Equation

Hongyu Zhao

School of Mathematics and Statistics, Qingdao University, Qingdao Shandong

Received: Feb. 19<sup>th</sup>, 2025; accepted: Mar. 12<sup>th</sup>, 2025; published: Mar. 18<sup>th</sup>, 2025

## Abstract

In the field of research into heat transfer equations, the application of physical information neural network (PINN) has achieved some results. The loss function of PINN consists of a weighted sum of

multiple loss terms, and the weighted combination of these loss terms plays an important role in PINN's effective training. Therefore, we construct a loss term definition based on a Gaussian probability model, where the introduction of noise parameters is used to describe the weight of each loss term. We propose a self-adaptive loss function method based on the maximum likelihood estimation principle to automatically assign loss weights by constantly updating noise parameters in each training cycle. Then, we use self-adaptive loss physical information neural network (SalPINN) to solve the one-dimensional transient heat transfer equation, and compare it with the traditional PINN method, and the results show that SalPINN is more accurate and effective in simulating the heat transfer equation.

## Keywords

Heat Transfer Equation, Physical Information Neural Network, Adaptive Loss Balance Method

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

随着问题复杂性的增加,数值计算模型在处理时间和硬件资源上的消耗日益显著。这些问题通常以偏微分方程(PDEs)的形式出现,作为模拟物理、生物学等领域现象的控制方程。尽管数值求解器是求解这些PDEs的便捷工具,但其对计算资源的高要求促使人们寻求替代模型以降低成本。机器学习(ML),尤其是深度神经网络(DNN),得益于Python编程语言中流行ML库的进步,已成为代理建模的重要选择。DNN作为一种机器学习模型,经过适当训练后,可作为数学问题的参数化解决方案。然而,DNN训练通常需要庞大的数据样本集,这在代理建模中意味着需要收集被代理模型模拟驱动的数据。当模型计算成本高昂时,生成足够的数据变得不切实际。因此,能够在较少数据下实现收敛的DNN模型备受关注,其中物理信息神经网络(PINN)便是一种有效方法[1]。PINN本质上是DNN,但其损失函数采用了不同的表达方式,通过软约束将PDE作为唯一解,而非依赖大数据来确定关系。PINN能够在更短时间内构建网络解决PDEs问题,且所需数据量少于标准DNN。这一优势已在多个基准问题上得到验证,如Burger方程[2]、Schrödinger方程[3]、非线性扩散率方程和Biot方程等[4]。PINN的变体,即生物信息神经网络(BINN)[5],也被用于模拟来自从头开始分析实验的稀疏真实世界数据的反应-扩散方程。通过利用智能数据和针对基准测试问题的最新应用,PINN有潜力缓解数值求解PDE时的计算需求。

本文在普通物理信息神经网络的基础上,创新性地加入自适应损失加权法[6],自适应损失加权法在每次训练迭代开始时动态调整单个损失项的相对重要性,确保在没有单个损失分量占主导地位的情况下实现平衡的优化。这种方法促进了收敛速度的提高和模型准确性的提高。

文章第二节简要叙述了热力学方程;第三节给出PINN的一般模型、基于数值PINN的自适应损失物理信息神经网络(SalPINN)的基本原理;第四节展开数值实验,使用两种算法分别对一个一维瞬态热传导方程进行求解,并进行比较,以验证方法的有效性;第五节总结了方法的优缺点,并提出了潜在的应用前景。

## 2. 热力学方程

在本文中,我们主要考虑以下一维瞬态热传导方程:

$$\frac{\partial u(x,t)}{\partial t} - a \frac{\partial^2 u(x,t)}{\partial x^2} = f(x,t), \quad L_l < x < L_r, \quad t_0 < t, \quad (1)$$

式(1)中,  $a$  是热扩散系数, 与材料的热传导系数和比热容有关,  $u(x,t)$  表示温度, 描述了热量在一维物体中如何随时间和空间变化而传播,  $f(x,t)$  代表热源项, 它表示单位时间内单位体积内产生的热量,  $L_l$  表示空间坐标的左边界,  $L_r$  表示空间坐标的右边界,  $t_0$  是初始时刻。在本文中,  $f(x,t)$  定义为:

$$f(x,t) = g(x,t), \quad 0 < x < L, \quad t_0 < t, \quad (2)$$

式(2)中,  $g(x,t)$  为已知的函数。我们主要考虑第一类边界条件 Dirichlet 边界条件:

$$\begin{cases} u(x, t_0) = \Gamma(x, t_0), & L_l < x < L_r, \\ u(x_b, t) = N(x_b, t), & t_0 < t, \quad x_b \in \partial\Omega, \end{cases} \quad (3)$$

式(3)中,  $\Gamma(x)$  和  $N(t)$  均为已知函数,  $\partial\Omega$  表示空间坐标边界。

### 3. 算法基本原理

#### 3.1. 物理信息神经网络

物理信息神经网络作为一种数据驱动的方法已被广泛应用于求解一般非线性偏微分方程, 如 Burgers 方程、泊松方程和薛定谔方程。本文使用 PINN 来研究热传导问题, 为了解决式(1)~(3)中的热传导方程, 我们构建了多层前馈神经网络, 用以分别逼近目标解。在其最简单的形式中, 我们考虑深度为  $M$  的前馈全连接神经网络, 它们的输入为训练点的坐标  $(x, t)$ , 假设第  $m$  层的输出为  $\hat{u}^{[m]}$ , 神经网络可以定义为:

$$\begin{aligned} \text{input layer:} & \quad \hat{u}^{[m]} = x, \\ \text{hidden layer:} & \quad \hat{u}^{[m]} = \sigma(W^{[m]}\hat{u}^{[m-1]} + b^{[m]}), \\ \text{for } m = 2, 3, \dots, M-1, & \\ \text{output layer:} & \quad \hat{u}^{[M]} = (W^{[M]}\hat{u}^{[M-1]} + b^{[M]}), \end{aligned} \quad (4)$$

式(4)中,  $\sigma$  为激活函数[7], 包括 sigmoid、relu 和 tanh [8]。  $W^{[m]}$  和  $b^{[m]}$  分别表示第  $m$  层的权重和偏差。所有的权重矩阵和偏置向量都可以用一个参数集合  $\theta = \{W^{[m]}, b^{[m]}\}_{1 \leq m \leq M}$  表示。

在本文的 PINN 框架中, 训练点的坐标由三部分组成: 计算区域内部训练点  $\{x_f^j, t_f^j\}_{j=1}^{N_f}$ 、初始训练点  $\{x_i^j, t_i^j\}_{j=1}^{N_i}$ 、边界训练点  $\{x_b^j, t_b^j\}_{j=1}^{N_b}$ ,  $N_f$ 、 $N_i$  和  $N_b$  分别是计算区域内部训练点、初始训练点和边界训练点的个数。在下面的方程中, 所有偏导数都是通过自动微分(AD) [9]得到的。PINN 将物理的先验知识集成到损失函数中, 以增强数据的信息量, 在本文中, 损失函数被定义为:

$$Loss_{total} = Loss_f + Loss_i + Loss_b, \quad (5)$$

式(5)中,  $Loss_f$  表示偏微分方程的损失,  $Loss_i$  表示初始条件的损失,  $Loss_b$  表示边界条件的损失, 考虑 Dirichlet 边界条件[10], 假设神经网络输出的近似解为  $\hat{u}$ , 则式(5)中的子损失可表示为:

$$Loss_f = \sum_{j=1}^{N_f} \left| \frac{\partial \hat{u}(x_f^j, t_f^j)}{\partial t} - a \frac{\partial^2 \hat{u}(x_f^j, t_f^j)}{\partial x^2} - g(x_f^j, t_f^j) \right|^2, \quad (6)$$

$$Loss_i = \sum_{j=1}^{N_i} \left| \hat{u}(x_i^j, t_i^j) - \Gamma(x_i^j, t_i^j) \right|^2, \quad (7)$$

$$Loss_b = \sum_{j=1}^{N_b} \left| \hat{u}(x_b^j, t_b^j) - N(x_b^j, t_b^j) \right|^2. \quad (8)$$

由于自动微分技术能高效计算方程残差，上述损失函数在训练过程中无需额外标签数据，实现了无监督学习。该设计融合了监督与无监督学习，确保损失函数既能最小化已知数据误差，又能满足偏微分方程蕴含的物理约束。在输入必要的时空数据后，我们采用全连接神经网络逼近解函数，并利用自动微分技术计算残差以构建损失函数。通过梯度下降等优化算法调整神经网络权重，最终训练出符合偏微分方程的神经网络参数，实现对解函数的逼近。算法 1 总结了普通 PINNs 算法的参数限定和运行步骤。

#### 算法 1 普通物理信息神经网络(PINN)

**参数限定:** 迭代次数  $S$ ，学习率  $lr$ 。

**目标:** 找出参数  $\theta^*$  的最佳模型。

**开始**

**步骤 1:** 设置一个具有初始参数  $\theta$  的物理信息神经网络  $\hat{u}(x, t; \theta)$ 。

**步骤 2:** 指定训练点  $N_t = \{N_f, N_b, N_i\}$ 。

**步骤 3:** 使用梯度下降算法在  $S$  次迭代中将参数  $\theta$  更新为:

**循环** 从  $s = 1$  到  $s = S$

(a) 根据式(6)~(8)定义损失函数  $Loss_{total}(\varepsilon; \theta; N_t)$  (5)。

(b) 通过 Adam 更新参数  $\theta$  :

$$\theta_{s+1} \leftarrow Adam(Loss_{total}(\varepsilon_s; \theta_s; N); lr)$$

**结束**

### 3.2. 自适应损失物理信息神经网络

我们的想法受到一篇论文的启发，该论文提出了一种原则策略来权衡场景几何中的多个损失函数，以及语义多任务深度学习问题[5]。损失函数是基于均匀不确定性的高斯似然最大化来定义的。对于求解热传导方程的回归任务，似然定义为高斯分布，其均值由代理模型输出给出：

$$p(y | \hat{u}(x, t; \theta)) = N(\hat{u}(x, t; \theta), \varepsilon^2), \quad (9)$$

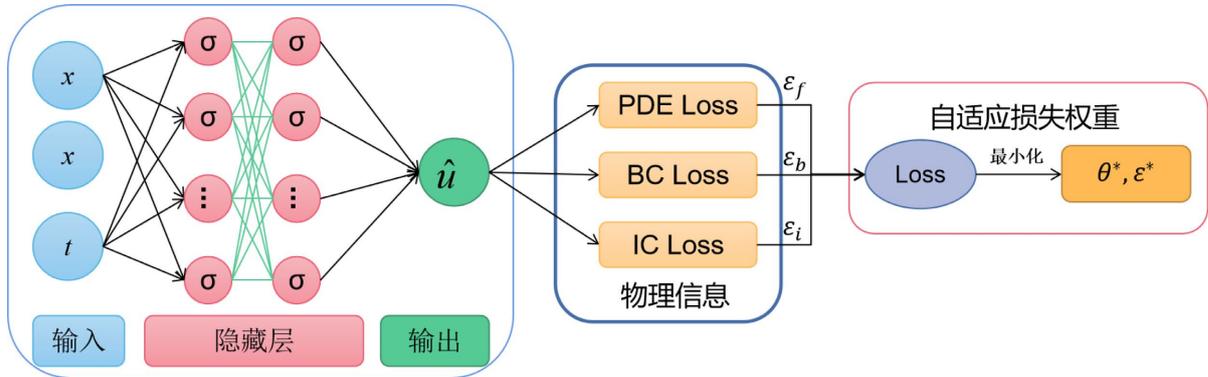
考虑输出服从高斯分布。作为神经网络权值衰减的一部分，式(9)中噪声标量  $\varepsilon$  通常是固定的， $\hat{u}(x, t; \theta)$  为神经网络的输出解。为了捕获依赖数据的任意不确定性，我们将根据最大似然推理调整观测噪声参数。基于最小化目标，模型的负对数似然为：

$$\begin{aligned} -\log p(y | \hat{u}(x, t; \theta)) &\propto \frac{1}{2\varepsilon^2} \|y - \hat{u}(x, t; \theta)\|^2 + \log \varepsilon \\ &= \frac{1}{2\varepsilon^2} L_1(\theta) + \log \varepsilon, \end{aligned} \quad (10)$$

式(10)中， $L_1(\theta)$  表示输出变量的损失，我们将(6)~(8)中的损失项代入，则可以得到：

$$\begin{aligned} Loss_{total}(\varepsilon; \theta; N_t) &= \frac{1}{2\varepsilon_f^2} Loss_f(\theta; N_f) + \frac{1}{2\varepsilon_b^2} Loss_b(\theta; N_b) \\ &\quad + \frac{1}{2\varepsilon_i^2} Loss_i(\theta; N_i) + \log \varepsilon_f \varepsilon_b \varepsilon_i, \end{aligned} \quad (11)$$

式(11)中， $\varepsilon = \{\varepsilon_f, \varepsilon_b, \varepsilon_i\}$  表示每个损耗项的噪声参数，我们的目标是通过使用基于梯度的优化器，如随机梯度下降(SGD)、Adam 和 L-BFGS [11]最小化损失  $Loss_{total}(\varepsilon; \theta; N_t)$  来找到最佳模型权重  $\theta^*$  和噪声标量  $\varepsilon^*$ 。本文的自适应损失神经网络框架如图 1 所示。算法 2 总结了 PINN 的自适应损失平衡算法的运行流程和步骤。



**Figure 1.** Self-adaptive loss physical information neural network (SalPINN) framework  
**图 1.** 自适应物理信息神经网络(SalPINN)框架

### 算法 2 自适应物理信息神经网络(SalPINN)

参数限定: 迭代次数  $S$ , 学习率  $l_r$ , 初始化噪声项  $\varepsilon = \{\varepsilon_f, \varepsilon_b, \varepsilon_i\}$ 。

目标: 找出参数为  $\theta^*$  和噪声标量  $\varepsilon^*$  的最佳模型。

开始

步骤 1: 设置一个具有初始参数  $\theta$  的物理信息神经网络  $\hat{u}(x, t; \theta)$ 。

步骤 2: 指定训练点  $N_t = \{N_f, N_b, N_i\}$ 。

步骤 3: 构造一个高斯概率模型, 其均值由 PINNs 的输出和噪声集合  $\varepsilon$  给出。

步骤 4: 使用梯度下降算法在  $S$  次迭代中将参数  $\theta$  和  $\varepsilon$  更新为:

循环 从  $s = 1$  到  $s = S$

(a) 根据极大似然估计(10)定义加权损失函数  $Loss_{total}(\varepsilon; \theta; N_t)$  (11)。

(b) 通过 Adam 调整噪声参数  $\varepsilon$ , 满足约束的概率最大化:

$$\varepsilon_{s+1} \leftarrow Adam(Loss_{total}(\varepsilon_s; \theta_s; N); l_r)$$

(c) 通过 Adam 更新参数  $\theta$ :

$$\theta_{s+1} \leftarrow Adam(Loss_{total}(\varepsilon_s; \theta_s; N); l_r)$$

结束

## 4. 数值实验

在本节中, 为了验证上述方法的可行性, 设计了一个数值实验算例对算法 1 和算法 2 的性能进行验证, 并且本文算法 1 和算法 2 的基础网络框架是相同的。

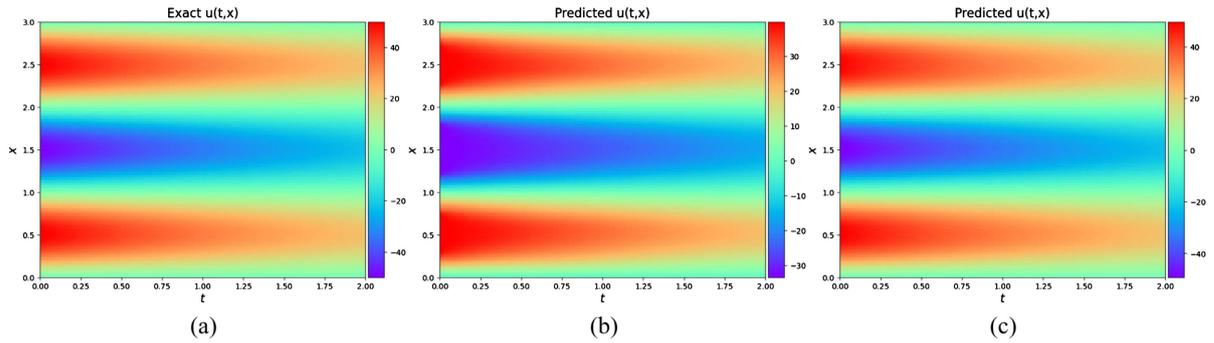
热传导方程是一个描述热量在物体内部或区域内的传播规律以及温度如何随时间变化的物理模型, 本节实验我们对一维瞬态热传导方程进行求解, 该偏微分方程为:

$$\begin{cases} u_t = au_{xx}, & 0 < x < L, \quad 0 < t < t_1, \\ u|_{x=0} = 0, \\ u|_{x=L} = 0, \\ u|_{t=0} = 50 \sin \pi x, & 0 < x < L, \end{cases} \quad (12)$$

式(12)中,  $a = 0.04$ ,  $L = 3$ ,  $t_1 = 10$ , 精确解为  $u(x, t) = 50e^{-a^2\pi^2t} \sin \pi x$ 。实验中的物理信息神经网络和自适应损失物理信息神经网络的拓扑结构均为全连接网络, 输入层有 2 个神经元, 输出层有 1 个神经元, 隐藏层有 6 层, 每层 40 个神经元, 隐藏层间以双曲正切函数  $\tanh$  为激活函数。神经网络使用 Adam 优化

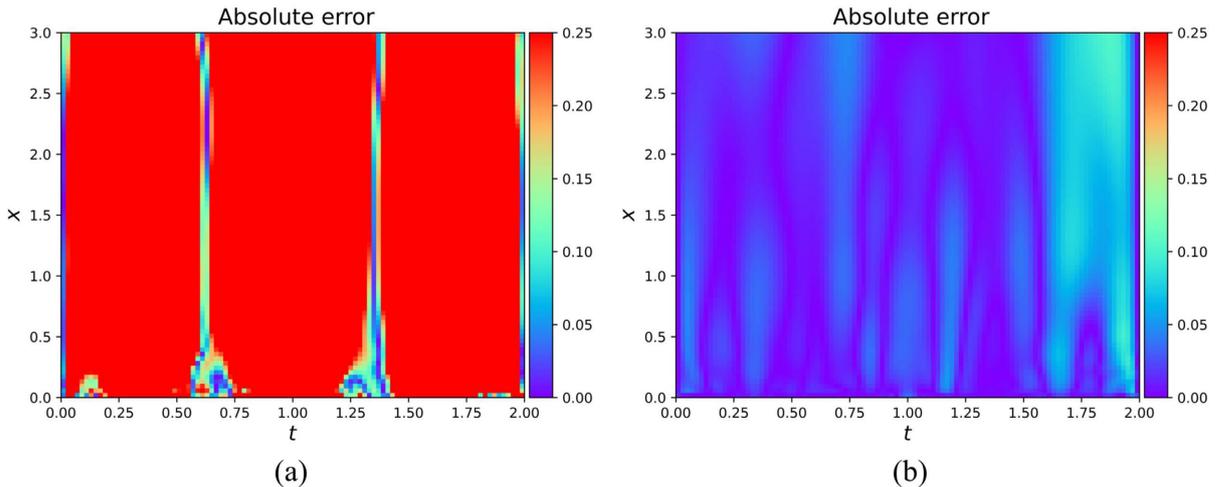
器以 0.001 为学习率迭代 7500 次，训练点设置为  $N_f = 600$ ， $N_i = 40$ ， $N_b = 100$ 。所有实验均在 GPU-GTX4060 上进行，编程语言采用 Python。本文判断热传导方程的误差定义为：

$$\text{L2 error} = \frac{\sqrt{\sum_{i=1}^N |\hat{u}(\mathbf{x}_i, t_i) - u(\mathbf{x}_i, t_i)|^2}}{\sqrt{\sum_{i=1}^N |u(\mathbf{x}_i, t_i)|^2}} \quad (13)$$



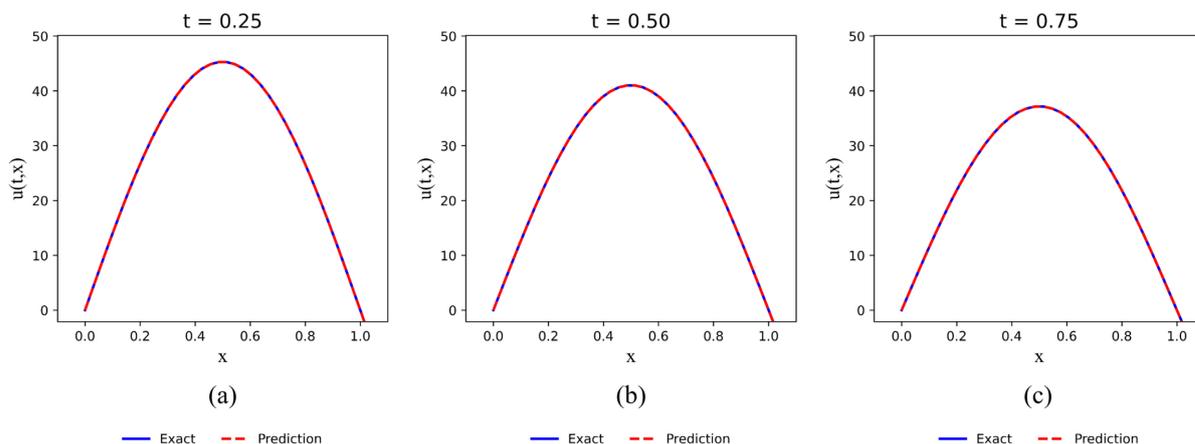
**Figure 2.** (a) Exact solution; (b) Numerical solution of algorithm 1; (c) Numerical solution of algorithm 2  
**图 2.** (a) 精确解; (b) 算法 1 的数值解; (c) 算法 2 的数值解

图 2 展示了算法 1 (PINN)与算法 2 (SaIPINN)在一维瞬态热传导方程求解中的结果。总体而言，两种算法均能模拟热传导过程的基本特性，清晰展现温度随时间的变化。图 2(a)呈现了方程(12)的精确解，而图 2(b)和图 2(c)则分别展示了算法 1 和算法 2 的数值解。通过对比发现，算法 1 在  $x \in (1, 2)$  区域内的数值解与精确解存在较大偏差，而算法 2 在整个区域内的数值解则与精确解基本一致。



**Figure 3.** (a) L2 error of algorithm 1; (b) L2 error of algorithm 2  
**图 3.** (a) 算法 1 的 L2 误差; (b) 算法 2 的 L2 误差

经过深入对比，图 3 展示了算法 1 与算法 2 在时空区域内测试点的 L2 误差。结果显示，算法 1 的误差整体较大，而算法 2 在引入自适应损失平衡机制后，其 L2 误差更为稳定，最高误差值介于 0.10 至 0.15 之间，表明算法 2 的性能显著提升，稳定性增强。



**Figure 4.** Numerical solution of algorithm 2 in  $x \in [0, 1]$  when: (a)  $t = 0.25$ ; (b)  $t = 0.50$ ; (c)  $t = 0.75$ .

**图 4.** 算法 2 在  $x \in [0, 1]$  时某一时刻的数值解: (a)  $t = 0.25$ ; (b)  $t = 0.50$ ; (c)  $t = 0.75$

为了进一步测试算法 2 的计算性能, 图 4 分别展示了热传导方程在  $t = 0.25$ 、 $t = 0.50$  和  $t = 0.75$  时刻的温度状态, 图中蓝色实线为精确解, 红色虚线为算法 2 的数值解, 从图像我们可以看出算法 2 的预测结果基本与精确解一致, 这说明算法 2 能够精确计算该问题。

## 5. 总结

在本文中, 我们研究了一维瞬态热传导方程背景下, 普通物理信息神经网络(PINN)与自适应损失物理信息神经网络(SalPINN)的性能。PINN 的性能与收敛性易受损失权重选择的影响, 因此自适应地分配适当权重以组合 PINN 中的多个损失函数至关重要。我们提出了一种基于最大化高斯似然及可扩展不确定性参数的自适应损失平衡方法, 旨在同时学习 PINN 中的竞争损失项, 以提高热传导方程的计算精度。

我们通过数值实验验证了其有效性, 将其应用于一维瞬态热传导方程的求解, 数值实验结果表明, 与传统固定权重 PINN 相比, SalPINN 在热传导方程求解中降低了误差, 在空间区域内的解与精确解吻合度显著提高, 提升了稳定性。然而, 当前方法在计算效率上尚有不足, 这将是未来研究的方向。我们计划将 SalPINN 方法拓展至高维度偏微分方程的计算, 并通过优化参数选择及结合其他优化策略, 克服现有局限, 推动该方法在更广泛领域的应用。

## 参考文献

- [1] Raissi, M., Perdikaris, P. and Karniadakis, G.E. (2019) Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations. *Journal of Computational Physics*, **378**, 686-707. <https://doi.org/10.1016/j.jcp.2018.10.045>
- [2] Jagtap, A.D., Kharazmi, E. and Karniadakis, G.E. (2020) Conservative Physics-Informed Neural Networks on Discrete Domains for Conservation Laws: Applications to Forward and Inverse Problems. *Computer Methods in Applied Mechanics and Engineering*, **365**, Article ID: 113028. <https://doi.org/10.1016/j.cma.2020.113028>
- [3] Pu, J., Li, J. and Chen, Y. (2021) Solving Localized Wave Solutions of the Derivative Nonlinear Schrödinger Equation Using an Improved PINN Method. *Nonlinear Dynamics*, **105**, 1723-1739. <https://doi.org/10.1007/s11071-021-06554-5>
- [4] Kadeethum, T., Jørgensen, T.M. and Nick, H.M. (2020) Physics-Informed Neural Networks for Solving Nonlinear Diffusivity and Biot's Equations. *PLOS ONE*, **15**, e0232683. <https://doi.org/10.1371/journal.pone.0232683>
- [5] Lagergren, J.H., Nardini, J.T., Baker, R.E., Simpson, M.J. and Flores, K.B. (2020) Biologically-Informed Neural Networks Guide Mechanistic Modeling from Sparse Experimental Data. *PLOS Computational Biology*, **16**, e1008462. <https://doi.org/10.1371/journal.pcbi.1008462>
- [6] Cipolla, R., Gal, Y. and Kendall, A. (2018) Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry

- and Semantics. 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 18-23 June 2018, 7482-7491. <https://doi.org/10.1109/cvpr.2018.00781>
- [7] 张焕, 张庆, 于纪言. 激活函数的发展综述及其性质分析[J]. 西华大学学报(自然科学版), 2021, 40(4): 1-10.
- [8] Sharma, S., Sharma, S. and Athaiya, A. (2020) Activation Functions in Neural Networks. *International Journal of Engineering Applied Sciences and Technology*, **4**, 310-316. <https://doi.org/10.33564/ijeast.2020.v04i12.054>
- [9] 张海斌, 薛毅. 自动微分的基本思想与实现[J]. 北京工业大学学报, 2005, 31(3): 332-336.
- [10] Givoli, D. (1991) Non-Reflecting Boundary Conditions. *Journal of Computational Physics*, **94**, 1-29. [https://doi.org/10.1016/0021-9991\(91\)90135-8](https://doi.org/10.1016/0021-9991(91)90135-8)
- [11] Abdolrasol, M.G.M., Hussain, S.M.S., Ustun, T.S., Sarker, M.R., Hannan, M.A., Mohamed, R., *et al.* (2021) Artificial Neural Networks Based Optimization Techniques: A Review. *Electronics*, **10**, Article 2689. <https://doi.org/10.3390/electronics10212689>