

专题: 非线性系统理论及其前沿应用

基于梯度优化物理信息神经网络 求解复杂非线性问题*

田十方 李彪†

(宁波大学数学与统计学院, 宁波 315211)

(2022年12月14日收到; 2023年1月5日收到修改稿)

近年来, 物理信息神经网络 (PINNs) 因其仅通过少量数据就能快速获得高精度的数据驱动解而受到越来越多的关注. 然而, 尽管该模型在部分非线性问题中有着很好的结果, 但它还是有一些不足的地方, 如它的不平衡的反向传播梯度计算导致模型训练期间梯度值剧烈振荡, 这容易导致预测精度不稳定. 基于此, 本文通过梯度统计平衡了模型训练期间损失函数中不同项之间的相互作用, 提出了一种梯度优化物理信息神经网络 (GOPINNs), 该网络结构对梯度波动更具鲁棒性. 然后以 Camassa-Holm (CH) 方程、导数非线性薛定谔方程为例, 利用 GOPINNs 模拟了 CH 方程的 peakon 解和导数非线性薛定谔方程的有理波解、怪波解. 数值结果表明, GOPINNs 可以有效地平滑计算过程中损失函数的梯度, 并获得了比原始 PINNs 精度更高的解. 总之, 本文的工作为优化神经网络的学习性能提供了新的见解, 并在求解复杂的 CH 方程和导数非线性薛定谔方程时用时更少, 节约了超过三分之一的计算时间, 并且将预测精度提高了将近 10 倍.

关键词: 物理信息神经网络, 梯度优化, Camassa-Holm 方程, 导数非线性薛定谔方程, peakon 解, 怪波解

PACS: 02.30.Ik, 02.60.-x, 05.45.-a

DOI: 10.7498/aps.72.20222381

1 引言

现代的科学、各种科技以及工程领域中的很多模型均可用偏微分方程来描述, 很多近代自然科学发展的基本方程其实也是偏微分方程, 所以人们总是用偏微分方程来描述、解释或预测各种自然界中的现象. 如非线性问题, 是最常见的一个以偏微分方程为基础数学模型的问题, 它广泛存在于物理工程、流体力学、化学力学等研究领域.

由于此类问题通常难以直接求得精确解, 所以采用数值方法来求得近似解尤为重要. 高精度的数值计算方法也引起了越来越多学者的关注. 求解此类问题的数值计算方法很多, 如有限差分法、有限

元法、有限体积法^[1-4]等. 但是, 数值解法的数据量之大、计算时间之长导致求得近似解较为困难. 而计算机具有高效性与实用性, 于是, 如何将计算机与数学更好地结合, 成为近些年来人们一直在研究的问题. 随着可用数据和计算资源的爆炸性增长, 机器学习和数据分析的最新进展在不同的科学学科中产生了革命性的结果, 包括图像识别、认知科学和基因组学^[5-8]等. 然而, 在分析复杂的物理、生物或工程系统的过程中, 数据采集的成本往往过高, 不可避免地面临在部分信息下得出结论和做出决策的挑战. 在这种小数据区域中, 绝大多数最先进的机器学习技术 (如深度/卷积/递归神经网络)^[9,10] 缺乏鲁棒性, 无法提供任何收敛保证. 乍看, 训练一个深度学习算法以从几个输入和输出数据

* 国家自然科学基金 (批准号: 12175111, 12235007) 和宁波大学王宽诚幸福基金资助的课题.

† 通信作者. E-mail: libiao@nbu.edu.cn

开始, 到最后可以准确识别非线性映射的任务, 这是不成熟的. 为了更好地解决问题, 在许多与物理和生物系统建模相关的案例中, 存在着大量的先验知识, 而这些知识目前尚未用于现代机器学习实践中.

随着计算科学的快速发展, 深度学习在许多领域取得了巨大成功, 包括计算机视觉 (CV)、自然语言处理 (NLP)、推荐系统、蛋白质结构预测等^[11–14]. 这些成功背后有一个重要的原因: 神经网络模型是复杂函数的良好逼近器. 近几十年来, 非线性偏微分方程 (NPDE) 的数值解一直是数学物理领域的一个热门话题. 物理信息神经网络 (PINNs) 被提出后, 其相关模型一直被应用于求解各种 NPDE, 如未知解的函数逼近^[15]和数据驱动发现^[16,17]. 值得注意的是, 一些学者在此领域做了大量有意义的工作, 利用神经网络的这一特性, 已经提出了许多数据驱动方法来求解 NPDE^[17–20], 其中 Raissi 和 Karniadakis^[17] 提出的 PINNs 模型在求解 NPDE 时, 以其高预测精度和良好的泛化能力而突出. 该模型先设计了损失函数, 并通过不断地减小损失函数使之趋近于 0, 使其受到潜在 NPDE 和边界条件的约束, 来完成方程解的近似. 也就是说, 它把一般 NPDE 描述的物理定理, 作为监督学习的约束条件. 随后, 出现了许多基于 PINNs 的改进的深度学习框架, 以提高其鲁棒性以及用于其他领域的泛化能力. 如 Raissi 等^[20] 提出了一种基于最小化的前向-后向随机神经网络模型来求解耦合的前向随机微分方程; Jagtap 等^[21] 提出了符合各种守恒定律的守恒物理信息神经网络 (cPINNs), 解决了多个子域上的问题, 并确保了子域边界上的通量连续性; Revanth 和 Susanta^[22] 随后提出了向后兼容的 PINNs, 它可以有效地解决大域或多尺度问题. Chen 和他的团队^[23–28] 对一些经典的数学物理方程进行了研究, 如 Sine-Gordon 方程、非线性薛定谔方程和导数非线性薛定谔方程, 获得了这些方程的数据驱动的呼吸子解、怪波解和其他孤子解; 并且在 PINNs 模型的基础上, 提出了一种两阶段 PINNs^[29], 对一类非线性物理方程进行了模拟, 获得了这些方程的局域波解. Ling 等^[30] 使用改进的 PINNs 模型对耦合非线性薛定谔方程进行模拟, 并获得了数据驱动的向量孤子解. He 和 Wang^[31] 使用 PINNs 模型对非线性薛定谔方程在扰动初始条件下的 Peregrine 孤子解进行了模拟. Wang 和 Yan 等^[32–35] 使用改进的 PINNs 模型研究了相关

方程的数据驱动解和参数发现, 并且通过改进的 PINNs 模型对 Camassa-Holm (CH), Degasperis-Procesi (DP) 和 Novikov 等方程的 peakon 解和周期解进行了预测^[36]. Bai 等^[37] 使用改进的 PINNs 模型求解了 Huxley 方程. Wu 等^[38] 通过改进的 PINNs 模型预测了双折射光纤中矢量光孤子的动力学过程和模型参数. Li 等^[39] 使用 PINNs 模型模拟了带有广义 PT 对称 Scarf-II 势的 NLS 方程; 并在 PINNs 模型的基础上提出了梯度优化物理信息神经网络模型^[40]和混合训练物理信息神经网络模型^[41], 这两种模型把模拟解的精度进一步提高. Dai 等^[42,43] 通过一种改进的 PINNs 模型模拟了包括非线性薛定谔方程、Korteweg-de Vries (KdV) 方程和 modified Korteweg-de Vries (KdV) 方程在内的孤子解. Yuan 等^[44] 通过 PINNs 模型解决了非线性积分微分方程的正逆问题. Zeng 等^[45] 通过一种自适应神经网络模型模拟了高维偏微分方程问题.

基于梯度均衡优化^[46]的思想, 本文提出一种梯度优化的物理信息神经网络 (GOPINNs). 具体而言, 通过对原始 PINNs 模型进行梯度统计, 在模型训练期间平衡损失函数中不同项之间的相互作用, 优化了梯度下降更新过程的方法, 并改变其完全连接的前馈神经网络结构. 对于方法的改进有两个主要动机: 1) 在模型训练期间, 使用反向传播梯度统计^[47]自动调整损失函数项系数, 以平衡损失函数项之间的相互作用; 2) 该想法来源于最近频繁使用于 CV 和 NLP 研究的神经网络注意力机制^[48], 在传统神经网络中添加了两个变压层 (transformer) 以更新隐藏层, 并使用跨层传播来减少传播误差. 总之, 我们可以平滑神经网络隐藏层上的梯度统计, 以使新的神经网络结构具有更好的稳定性和预测精度. 本文以 CH 方程的 peakon 解和导数薛定谔方程的有理波解和怪波解为例, 通过比较 PINNs 模型和 GOPINNs 模型的数值结果, 验证了新模型的良好学习性能.

第 2 节介绍了 PINNs 模型, 并通过梯度分析, 说明了 PINNs 模型的不足, 并且提出了 GOPINNs 模型, 以及对该方法如何进行梯度调整进行了介绍; 第 3 节以 CH 方程的 peakon 解、导数非线性薛定谔方程的有理波解和导数非线性薛定谔方程的怪波解为例, 通过比较两模型间的相对 L_2 误差, 评估了 PINNs 和 GOPINNs 两种模型的性能; 最后, 第 4 节给出了结论和讨论.

2 深度学习方法

2.1 PINNs 方法

首先, 简单介绍一下 PINNs 模型, 这是一个深度学习框架, 旨在获得一般形式的 NPDE 的潜在解 $\hat{q}(x, t)$ ^[39]. 通过 PINNs 模型求解的非线性偏微分方程如下:

$$\begin{cases} q_t + N[q] = 0, & (x, t) \in \Omega \times T, \\ q(x, 0) = I(x), & x \in \Omega, \\ q(x, t) = B(x, t), & (x, t) \in \partial\Omega \times T, \end{cases} \quad (1)$$

这里, T 和 Ω 分别代表该方程的时域和空域, $\partial\Omega$ 代表方程在空间方向上的边界区域. $N[q]$ 是关于 $q(x, t)$ 的非线性算子和线性算子的组合, $I(x, t)$ 和 $B(x, t)$ 分别是方程的初值条件与边界条件. 另外, 为了后续计算损失函数, 令 $f = q_t + N[q]$, 通过不断优化 f 来得到预测解 $\hat{q}(x, t)$.

要通过神经网络对原方程进行模拟, 从而模拟出关于原方程的预测解 $\hat{q}(x, t)$. 用 PINNs 来求解带有初始条件和边界条件的非线性偏微分方程需要有一个损失函数 MSE, 并且通过随机梯度下降法 (SGD) 来处理损失函数, 对于损失函数 MSE:

$$\text{MSE} = \sum_{i=1}^{N_B} \lambda_i \text{MSB} + \sum_{j=1}^{N_I} \lambda_j \text{MSI} + \text{MSf}, \quad (2)$$

其中 MSB 为边界条件损失函数, MSI 为初值损失函数, MSf 是方程残差. λ_i 和 λ_j 是根据求得的梯度来确定的系数, 用来平衡损失函数使其达到最佳效果. 值得一提的是, 在 PINNs 模型中, λ_i 和 λ_j 的值均为 1.

另外, 将损失函数的每一部分定义为

$$\begin{aligned} \text{MSB} &= \frac{1}{N_B} \sum_{l=1}^{N_B} [q(x_B^l, t_B^l) - B^l]^2, \\ \text{MSI} &= \frac{1}{N_I} \sum_{l=1}^{N_I} [q(x_I^l, t_I^l) - I^l]^2, \end{aligned} \quad (3)$$

$$\text{MSf} = \frac{1}{N_R} \sum_{l=1}^{N_R} [f(x_R^l, t_R^l) - R^l]^2, \quad (4)$$

其中 x_I^l, t_I^l, N_I 和 x_B^l, t_B^l, N_B 分别是方程初值数据集和边界条件数据集; x_R^l, t_R^l, N_R 是方程残差数据集. 这些设计的最终目标是构建深度神经网络, 使得损失函数尽可能接近 0.

2.2 GOPINNs 方法

尽管有一些积极的结果, 但 PINNs 模型在逼近潜在解 $\hat{q}(x, t)$ 时仍存在一些意想不到的困难. 有些方程模拟出来的解不尽如人意, 精度不高, 甚至根本无法模拟出来. 于是, 如何在已有的 PINNs 模型的基础上进行改进, 使得新模型可以模拟出精度更高的解, 成为一个重要的研究方向. 近期也出现了不少优秀的模型, 我们从 Glorot 和 Bengio^[49] 的有趣工作中获得了灵感, 这项工作是在训练过程中监测模型的每个隐藏层中神经网络参数的反向传播梯度波动. 我们提出的梯度优化算法的主要思想是通过迭代出来的损失值以及求得的梯度来不断调整 λ_i 和 λ_j 的值使其达到最佳效果, 从而通过调整梯度来进一步提高机器学习模型模拟求解的精度. 需要注意的是, 我们没有跟踪总损失函数 MSE 的梯度, 跟踪的是深度神经网络的每个隐藏层中的初值函数 MSI 的梯度、边界条件函数 MSB 的梯度和方程残差 MSf 的梯度.

下面介绍在梯度优化模型中具体怎么进行梯度优化. 首先, 分别给边界损失函数和初值损失函数的系数 λ_i 和 λ_j 一个初值 α 和 β , 为了方便后续进行梯度调整, 设一个学习率 rate. 通过对每一次迭代中损失函数 MSI, MSB 和 MSf 进行求梯度操作, 分别取 MSf 梯度的最大值, 记作 g_r , 满足条件 $g_r = \max \text{grad}_{\text{ref}}$; 取 MSI 和 MSB 梯度的平均值, 记作 g_I 和 g_B , 分别满足条件 $g_I = \text{meangrad}_I$ 和 $g_B = \text{meangrad}_B$. 通过以下两个过程:

$$\begin{cases} \lambda_i = g_r / g_B, \\ \lambda_j = g_r / g_I, \end{cases} \quad (5)$$

$$\begin{cases} \lambda_i = \lambda_i \times (1 - \text{rate}) + \text{rate} \times \lambda_i, \\ \lambda_j = \lambda_j \times (1 - \text{rate}) + \text{rate} \times \lambda_j, \end{cases} \quad (6)$$

对损失函数 MSI 和 MSB 的系数 λ_i 和 λ_j 进行更新, 以达到梯度优化的目的.

另外, 值得一提的是, 因为原始的 PINNs 模型是通过 Matlab 求得的方程解, 并将数据保存到 .mat 文件中, 然后通过 Python 读取 .mat 文件, 才开始了机器学习、模拟求解的过程, 这个过程中不免会出现电脑自带的误差, 这部分误差可能会影响最终的精确度. 为了避免这种情况, 我们将真解也写进了 Python 中, 真解和预测解均通过 Python 求得, 这样有效地减小了电脑自带误差.

下面介绍 GOPINNs 的全连接网络架构, 网络结构优化是深度学习中的一个重要研究思想, 在求解 NPDE 时通常缺乏物理知识神经网络的一般逼近定理, 因此标准的全连接结构是否能够提供足够灵活的表示来推断更复杂的 NPDE 是我们需要关注的问题. 受到 CV 和 NLP 中广泛使用的神经网络注意机制的启发, 我们对标准的全连接网络架构进行了简单的修改, 并提出了一种新的网络架构, 并通过数值结果表明 (第 3 节), 新提出的体系结构的推理性能优于原始 PINNs 方法. 为了防止梯度爆炸现象的产生, 我们调整传统的全连接神经网络, 引入两个变压层 U 和 V 来平滑 NPDE 的扩散项, 也就是所谓的跨层传播, 然后根据前馈传播规则使用逐点乘法运算来更新隐藏层. 如图 1 所示, 其中 Input 和 Output 代表输入层和输出层; U 和 V 是添加的两个变压层; Hidden1, Hidden2, ..., HiddenN 表示隐藏层, 双曲正切函数 tanh 作为隐藏层的激活函数. $I(x, t)$ 为初值条件, $B(x, t)$ 为边界条件, ε 是一个可控的超参数, maxiter 为迭代的最大步骤, 以上两个参数被用来当作循环继续的条件. MSf 是方程残差, MSB 是边界条件损失, MSI 是初值损失; 另外 g_r, g_B, g_I 分别是方程残差、边界损失和初值损失的梯度, λ_i 和 λ_j 分别是初值损失和边界损失的梯度, 最后通过不断迭代循环, 得到最终预测解 $\hat{q}(x, t)$. 在这里, $\hat{q}(x, t)$ 来自于输出层 Output 中 q 的迭代优化.

3 数值实验

3.1 Camassa-Holm 方程

本节以 Camassa-Holm 方程为例进行数值实验, 比较 PINNs 和 GOPINNs 的性能. Camassa-Holm 方程是一类十分重要而又特别的新型浅水波方程, 1993 年 Camassa 和 Holm^[50] 将其作为浅水波方程提出. 具体方程格式如下:

$$\begin{cases} q_t - q_{xxt} + 3qq_x - 2q_xq_{xx} - qq_{xxx} = 0, & (x, t) \in \Omega \times T, \\ q(x, 0) = I(x), & x \in \Omega, \\ q(x, t) = B(x, t), & (x, t) \in \partial\Omega \times T, \end{cases} \quad (7)$$

其中 $T \in [0, 1], X \in [-1, 1], q(x, t)$ 是含有变量 x 和 t 的函数, x 和 t 分别是空间变量和时间变量. 在此以 CH 方程的 peakon 解 $ce^{-|x-ct|}$ ^[32] 为例, peakon 解的格式为

$$\begin{cases} q(x, t) = ce^{-|x-ct|}, & (x, t) \in \Omega \times T, \\ q(x, 0) = ce^{-|x|}, & x \in \Omega, \\ q(-1, t) = ce^{-|1+ct|}, \\ q(1, t) = ce^{-|1-ct|}, & (x, t) \in \partial\Omega \times T. \end{cases} \quad (8)$$

为了更好地观察 peakon 解的分布, 图 2 给出了了解的三维图.

首先使用 PINNs 模型在具有 6 个隐藏层和每个隐藏层 50 个神经元的深度神经网络上对该方程的 peakon 解进行模拟, 使用双曲正切 (tanh) 作为

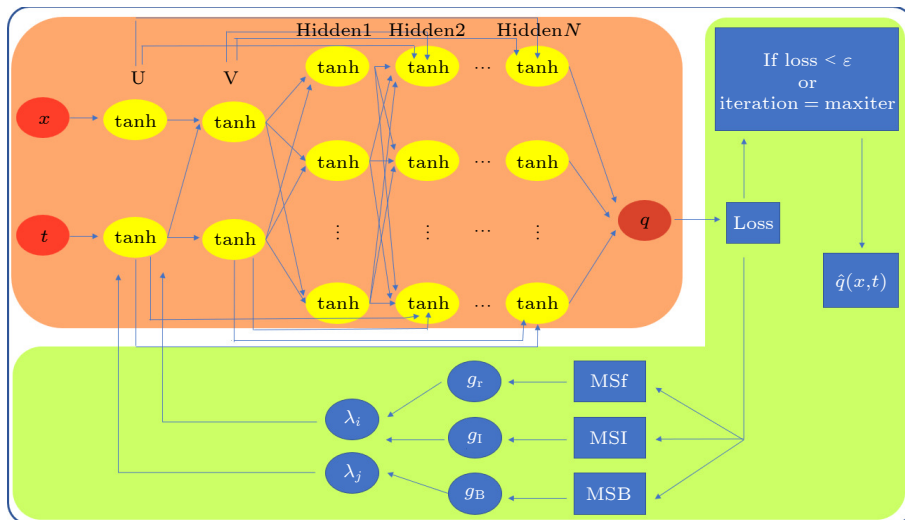


图 1 GOPINNs 模型

Fig. 1. GOPINNs model.

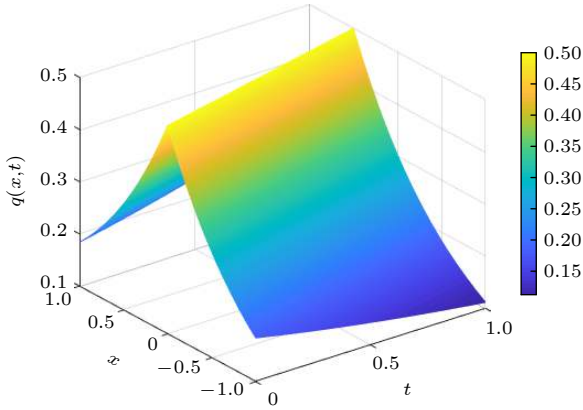


图 2 当 $c = 0.5$ 时, CH 方程的 peakon 解 ((8) 式) 的三维图
Fig. 2. Three-dimensional diagram of peakon solution (Eq. (8)) of the CH equation when $c = 0.5$.

非线性激活函数. 使用 Adam 优化器进行 40000 次迭代后, 数值预测结果、绝对误差、每一层隐藏层梯度分布和损失函数值如图 3—图 5 所示.

从图 3(c) 可以看出, 使用 PINNs 模型模拟的误差较大, 相对 L_2 误差为 5.48×10^{-2} . 图 4 中的梯度值分别表示每个隐藏层中的初始 g_I , 边界损失 g_B 和方程残差 g_r (这几个梯度项在第 2 节被定义), 它们集中在零点附近, 并形成尖峰, 这可能是梯度失衡的原因. 众所周知, 当梯度波动很大时, 神经网络将很容易获得满足 NPDE 的任何解. 因此, 本文训练的模型应该严格返回残差尽可能接近零的 NPDE 的解, 否则很容易返回错误的预测.

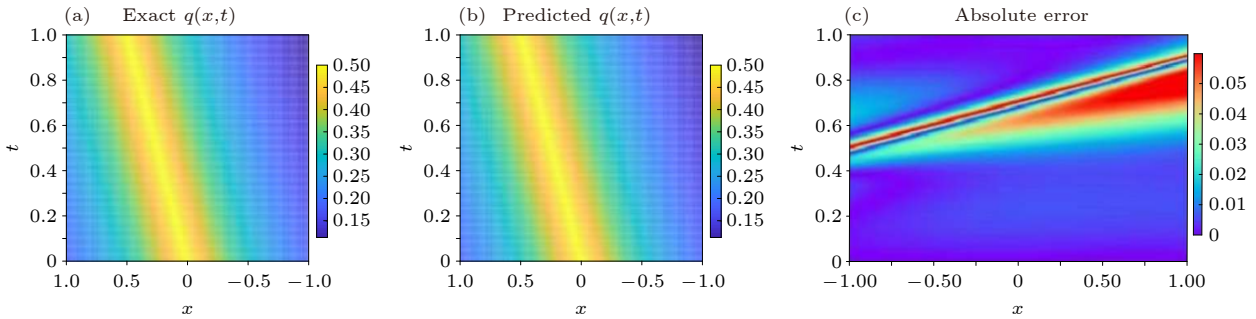


图 3 (a) CH 方程的 peakon 解 ((8) 式); (b) 应用 PINNs 模型模拟的预测解; (c) PINNs 模型的绝对误差
Fig. 3. (a) Peakon solution of CH equation (Eq. (8)); (b) prediction solution simulated by PINNs model; (c) absolute error of PINNs model.

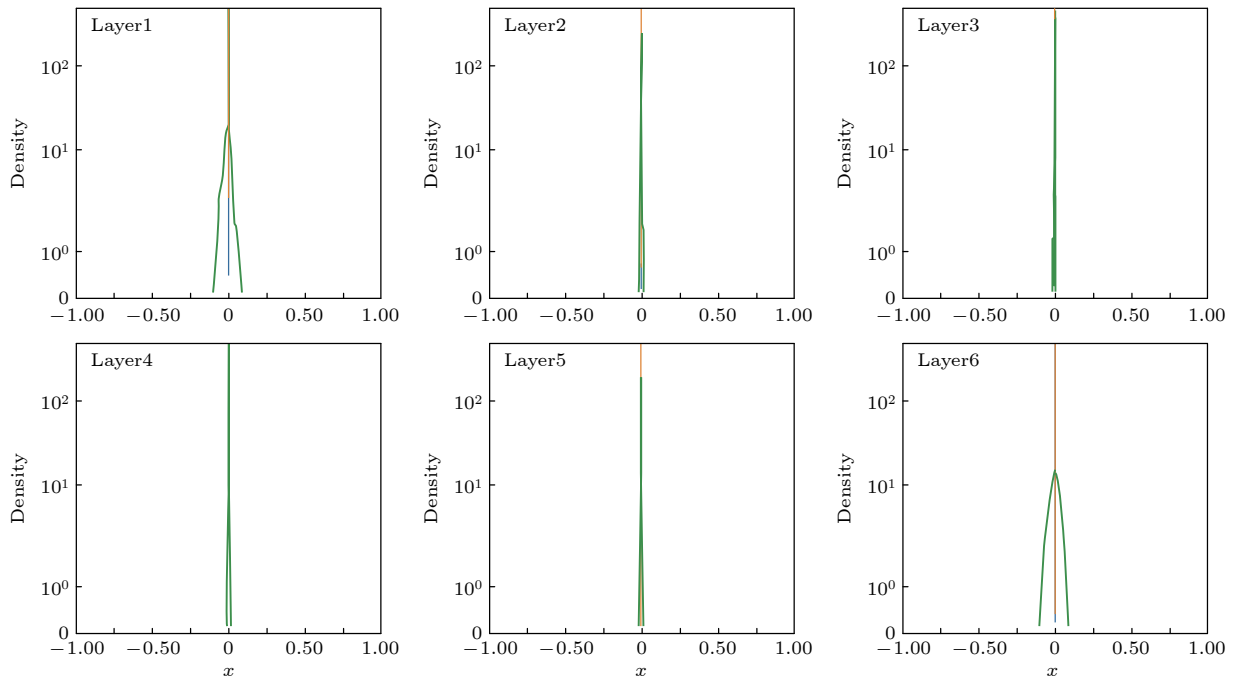


图 4 应用 PINNs 模型时, 每一个隐藏层的梯度分布, 其中绿色、蓝色、黄色曲线分别对应 g_r , g_I 和 g_B
Fig. 4. When PINNs model is applied, the gradient distribution of each hidden layer is g_r , g_I and g_B by color (green, blue, yellow).

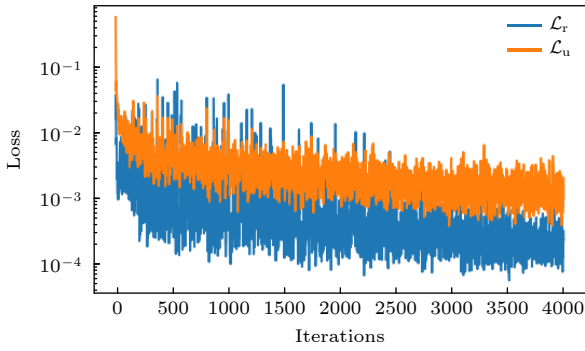


图 5 应用 PINNs 模型时, 整个迭代过程中损失函数 MSE 的值, 其中 \mathcal{L}_r 表示 NPDE 的残差的平均值, \mathcal{L}_u 表示 NPDE 在初始值和边界上的误差的平均值

Fig. 5. When the PINNs model is applied, the value of the function MSE is lost throughout the iteration, where \mathcal{L}_r represents the average value of the residual error of NPDE, and \mathcal{L}_u represents the average value of the error of NPDE at the initial value and the boundary.

损失函数 MSE 随不同迭代的变化如图 5 所示, 其中 \mathcal{L}_r 表示 NPDE 的残差的平均值, \mathcal{L}_u 表示 NPDE 在初始值和边界上的误差的平均值. 显然可以看到, \mathcal{L}_r 和 \mathcal{L}_u 的值在整个迭代过程中非常不稳定, 这也从侧面解释了不稳定、变化幅度较大的梯度会导致 PINNs 模型的预测精度较差. 为了一致性, 这里依然设置了具有 6 个隐藏层和每个隐藏层 50 个神经元的深度神经网络, 并使用双曲正切作为非线性激活函数. 使用 Adam 优化器通过 SGD 进行 40000 次迭代后, 新提出的 GOPINNs 模型的数值预测结果以及绝对误差如图 6 所示. 很明显, 所提出的新的训练方案可以适当地平衡初始和边界之间的相互作用, 并将预测解的相对 L_2 误差变成 7.43×10^{-3} , 与 PINNs 相比, 几乎降低一个数量级. 此处, 引出一个模型优化率的概念, 其定义为

$$\frac{[\text{PINNs}(L_2\text{error}) - \text{GOPINNs}(L_2\text{error})]}{\text{PINNs}(L_2\text{error})}. \quad (9)$$

所以, 提出的 GOPINNs 模型优化率为 86.4%.

将图 6 与图 3 中的原始 PINNs 模型相比可以发现, 在函数的边界和波峰部分的绝对误差有效地减少了. 通过调整传统的全连接前馈神经网络结构, 新模型 (GOPINNs) 的预测精度比 PINNs 方法的预测精度要高.

使用 GOPINNs 模型求解方程 (5) 的 peakon 解的其他数值结果如图 7 和图 8 所示. 比较图 4 与图 7 可以发现, GOPINNs 模型的梯度分布有了显著改善, 梯度随着隐藏层的更新逐渐平稳, 不会那么陡峭, 梯度值也变小了. 此现象也从另一方面说明了梯度优化模型是通过调整梯度让梯度平滑稳定, 来使预测解的精度提高. 另外, 从图 8 不难发现, GOPINNs 模型迭代过程中损失函数 MSE 中的 \mathcal{L}_r 和 \mathcal{L}_u 的值在整个迭代过程中较为稳定, 变化波动不大, 也体现了梯度优化模型的优良性能.

3.2 导数非线性薛定谔方程

为了验证所提模型的可行性, 用 GOPINNs 模型对导数非线性薛定谔方程的不同解进行数值实验. NPDE 多数来源于物理学、化学、生命科学等领域, 是经过抽象化后的数学模型, 具有非常鲜明的物理意义. NPDE 可以分为可积分方程和不可积分方程两类, 其中可积的 NPDE [51] 因为其孤子解的特性而成为非线性科学研究方向的热门问题. 导数非线性薛定谔 (DNLS) 方程 [52,53] 是严格可积的 NPDE 之一, 是用来描述等离子体中的 Alfvén 波的数学模型. 它同样也可以用来描述单模光纤中的亚皮秒或者飞秒脉冲波, 外加磁场下 (反) 铁磁

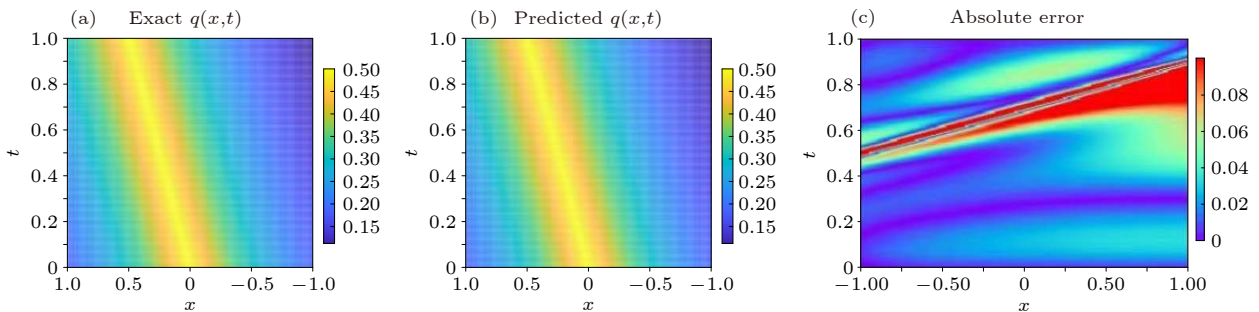


图 6 (a) CH 方程的 peakon 解 ((8) 式); (b) 应用 GOPINNs 模型模拟时的预测解; (c) GOPINNs 模型的绝对误差

Fig. 6. (a) Peakon solution of CH equation (Eq. (8)); (b) predicted solution when using GOPINNs model; (c) absolute error of GOPINNs model.

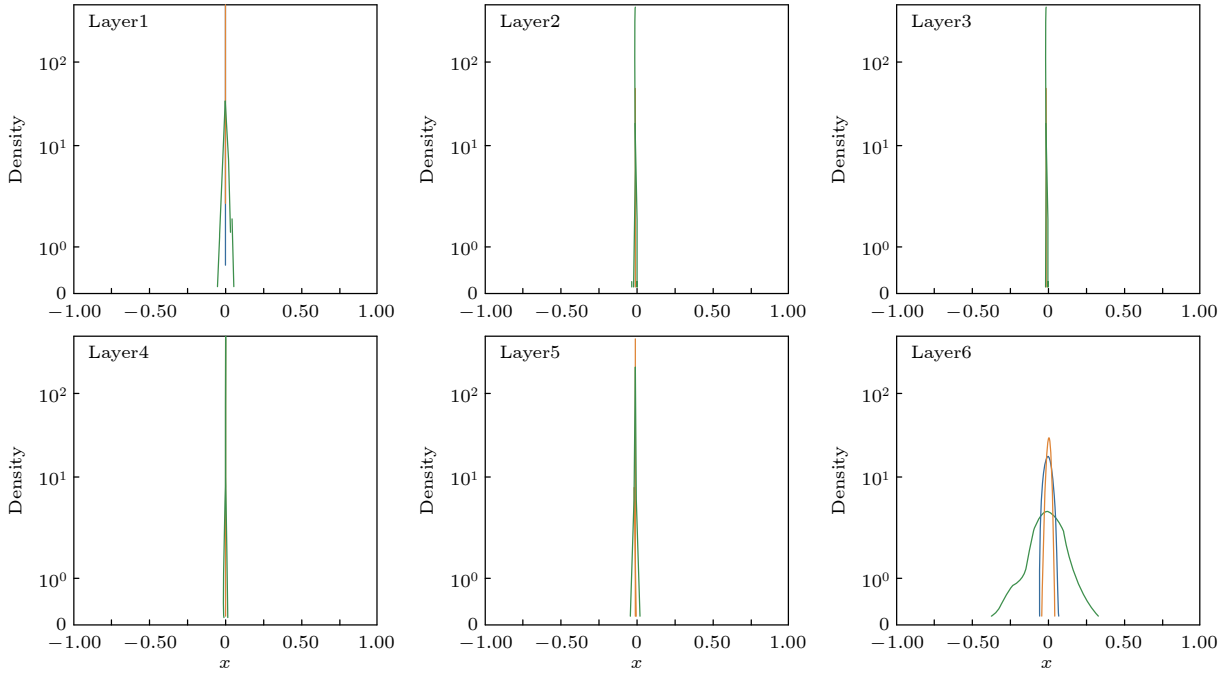


图 7 应用 GOPINNs 模型时, 每一个隐藏层的梯度分布, 其中绿色、蓝色、黄色曲线分别对应 g_f, g_l, g_B

Fig. 7. When GOPINNs model is applied, the gradient distribution of each hidden layer is g_f, g_l, g_B by color (green, blue, yellow).

媒质或电介质中的弱非线性电磁波. 求解 DNLS 方程孤子解的方法主要有 3 种: 反散射变换法 (IST)、Darboux 变换法 (DT) 以及 Hirota 双线性导数法. 其中 Xu 等^[53]通过 Darboux 变换法获得了 DNLS 方程的有理波解和怪波解, 此类方程具体形式如下:

$$\begin{cases} iqt - q_{xx} + i(q^2 q^*)_x = 0, & (x, t) \in \Omega \times T, \\ q(x, t_0) = I(x), & x \in \Omega, \\ q(x, t) = B(x, t), & (x, t) \in \partial\Omega \times T, \end{cases} \quad (10)$$

其中 $T \in [-4, 4]$, $X \in [-4, 4]$; x 和 t 分别是空间变量和时间变量; $q(x, t)$ 是含有实部 $u(x, t)$ 和虚部 $v(x, t)$ 的复合函数, 记作 $q(x, t) = u(x, t) + iv(x, t)$. 另外, 定义 $q(x, t)$ 的复共轭函数, 记作 $q(x, t) = u(x, t) - iv(x, t)$.

3.2.1 导数非线性薛定谔方程的有理波解

先以导数薛定谔方程的有理波解^[53]为例, 具体解的形式如下:

$$\begin{cases} q(x, t) = 4\beta_1 \exp[2i\beta_1^2(-x + 2\beta_1^2 t)] \frac{[4i\beta_1^2(4\beta_1^2 t - x) - 1]^3}{[16\beta_1^4(4\beta_1^2 t - x)^2 + 1]^2}, \\ q(x, -4) = 4\beta_1 \exp[2i\beta_1^2(-x - 8\beta_1^2)] \frac{[4i\beta_1^2(-16\beta_1^2 - x) - 1]^3}{[16\beta_1^4(-16\beta_1^2 - x)^2 + 1]^2}, \\ q(-4, t) = 4\beta_1 \exp[2i\beta_1^2(4 + 2\beta_1^2 t)] \frac{[4i\beta_1^2(4\beta_1^2 t + 4) - 1]^3}{[16\beta_1^4(4\beta_1^2 t + 4)^2 + 1]^2}, \\ q(4, t) = 4\beta_1 \exp[2i\beta_1^2(-4 + 2\beta_1^2 t)] \frac{[4i\beta_1^2(4\beta_1^2 t - 4) - 1]^3}{[16\beta_1^4(4\beta_1^2 t - 4)^2 + 1]^2}, \end{cases} \quad (11)$$

其中, $\beta_1 = 0.5$.

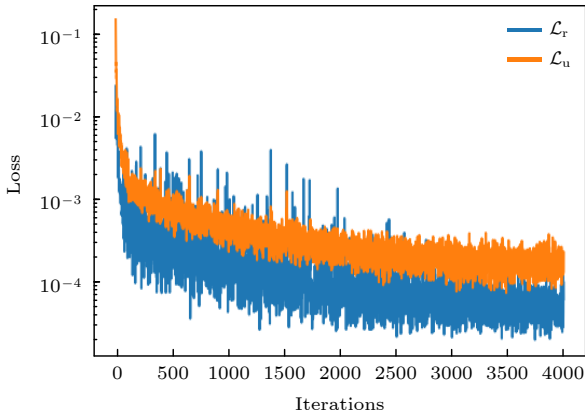


图 8 应用 GOPINNs 模型时, 整个迭代过程中损失函数 MSE 的值, 其中 \mathcal{L}_r 表示 NPDE 的残差的平均值, \mathcal{L}_u 表示 NPDE 在初始值和边界上的误差的平均值

Fig. 8. When the GOPINNs model is applied, the value of the function MSE is lost during the whole iteration, where \mathcal{L}_r represents the average value of residual error of NPDE, and \mathcal{L}_u represents the average value of the error of NPDE at the initial value and the boundary.

为了保持数值实验一致性, 使用 PINNs 和 GOPINNs 模型在具有 6 个隐藏层和每个隐藏层 50 个神经元的深度神经网络对方程 (6) 的有理波解进行模拟. 值得注意的是, 由于 $q(x, t)$ 是含有实部 $u(x, t)$ 和虚部 $v(x, t)$ 的复合函数, 所以最终模拟出的解应该是 $|q(x, t)|$, 满足条件 $|q(x, t)| = \sqrt{(u(x, t))^2 + (v(x, t))^2}$. 另外, 为了模拟出预测解, 在损失函数 MSE 中做了修改, 把函数残差 $f(x, t)$ 分为 $f_u(x, t)$ 和 $f_v(x, t)$, 分别是方程残差的虚数部分和实数部分, 满足条件 $f(x, t) = f_u(x, t) - if_v(x, t)$. 使用 Adam 优化器进行了 40000 次迭代后, 两种模型的数值预测结果以及时间消耗如表 1 所列, 图 9 给出了 GOPINNs 模型的预测结果和绝对误差. 根据上述结果可以发现, 换方程之后, GOPINNs 模型的精确度还是要优于 PINNs 的精确度, 而且 GOPINNs 模型模拟方程所需时间与 PINNs 模型相比, 节约了超过三分之一的时间. 提出的 GOPINNs 模型在精确度和所用时间两方面上都要优于原始模型, 这点很有意义.

3.2.2 导数非线性薛定谔方程的一阶怪波解

本小节以导数非线性薛定谔方程的怪波解 [53] 为例, 怪波是海洋动力学和数值分析中的重要研究课题. 由于其不稳定性和不可预测性, 研究怪波及其应用具有重要意义. 导数非线性薛定谔方程的一阶怪波解的形式如下:

表 1 两种模型对导数非线性薛定谔方程的有理波解 ((11) 式) 的数值预测结果

Table 1. Numerical prediction results of rational wave solution of derivative nonlinear Schrödinger equation (Eq. (11)) by two models.

神经网络信息	PINNs模型的相对 L_2 误差	GOPINNs模型的相对 L_2 误差	两模型模拟求解的时间消耗/s
6层隐藏层 50个神经元	0.237	4.56×10^{-2}	24564 15674.4

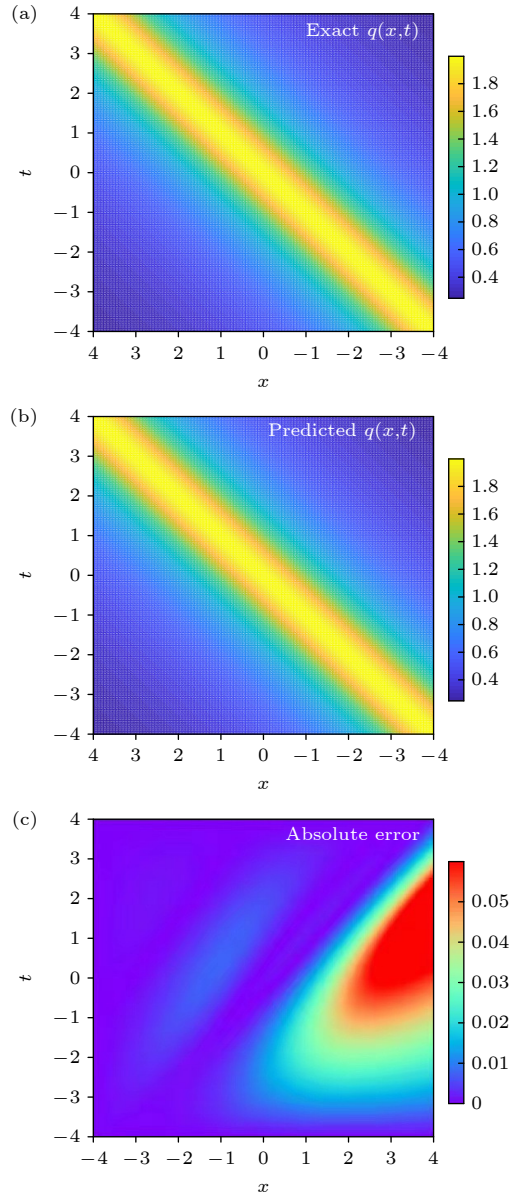


图 9 (a) 导数非线性薛定谔方程的有理波解 ((11) 式); (b) 使用 GOPINNs 模型模拟的预测解; (c) GOPINNs 模型的绝对误差

Fig. 9. (a) Rational wave solution of the derivative nonlinear Schrödinger equation (Eq. (11)); (b) predicted solution simulated by the GOPINNs model; (c) absolute error of the GOPINNs model.

$$\begin{cases} q_{\text{rogue}}(x, t) = \frac{r_1 r_2 r_3}{r_4 r_5}(x, t), \\ q_{\text{rogue}}(x, -4) = \frac{r_1 r_2 r_3}{r_4 r_5}(x, -4), \\ q_{\text{rogue}}(-4, t) = \frac{r_1 r_2 r_3}{r_4 r_5}(-4, t), \\ q(4, t) = \frac{r_1 r_2 r_3}{r_4 r_5}(4, t), \end{cases} \quad (12)$$

$$\begin{aligned} r_1 &= 2e^{[2i(\alpha_1^2 + \beta_1^2)(2t\alpha_1^2 + x - 2t\beta_1^2)]}, \\ r_2 &= \beta_1 [16\beta_1^2 \alpha_1^2 (4t\alpha_1^2 + x)^2 + 16\beta_1^4 (4t\beta_1^2 - x)^2 \\ &\quad + 8i\beta_1^2 (x + 4t\alpha_1^2 - 8t\beta_1^2)^2 + 1], \\ r_3 &= 2[16\beta_1^2 \alpha_1^2 (4t\alpha_1^2 + x)^2 + 16\beta_1^4 (4t\beta_1^2 - x)^2 \\ &\quad - 8\alpha_1 \beta_1^2 (x + 4t\alpha_1^2 - 8t\beta_1^2)^2 + 1] \\ &\quad \times [-\alpha_1 + 16\beta_1(\beta_1^4 - \alpha_1^4)t - 4\beta_1(\beta_1^2 + \alpha_1^2)x \\ &\quad + 16i\alpha_1 \beta_1^2 (\beta_1^2 + \alpha_1^2)t - i\beta_1] \\ &\quad - [16\beta_1^2 \alpha_1^2 (4t\alpha_1^2 + x)^2 + 16\beta_1^4 (4t\beta_1^2 - x)^2 \\ &\quad + 8i\beta_1^2 (x + 4t\alpha_1^2 - 8t\beta_1^2)^2 + 1] \\ &\quad \times [\alpha_1 + 16\beta_1(\beta_1^4 - \alpha_1^4)t - 4\beta_1(\beta_1^2 + \alpha_1^2)x \\ &\quad + 16i\alpha_1 \beta_1^2 (\beta_1^2 + \alpha_1^2)t + i\beta_1], \\ r_4 &= \alpha_1 + 16\beta_1(\beta_1^4 - \alpha_1^4)t - 4\beta_1(\beta_1^2 + \alpha_1^2)x \\ &\quad + 16i\alpha_1 \beta_1^2 (\beta_1^2 + \alpha_1^2)t + i\beta_1, \\ r_5 &= [-16\beta_1^2 \alpha_1^2 (4t\alpha_1^2 + x)^2 - 16\beta_1^4 (4t\beta_1^2 - x)^2 \\ &\quad + 8i\beta_1^2 (x + 4t\alpha_1^2 - 8t\beta_1^2)^2 - 1]^2, \end{aligned} \quad (13)$$

其中, $\alpha_1 = 0.5, \beta_1 = 0.5$.

为了保持数值实验一致性, 继续在具有 6 个隐藏层和每个隐藏层 50 个神经元的深度神经网络上分别应用 PINNs 模型和 GOPINNs 模型, 对方程 (6) 的一阶怪波解进行模拟, 通过 Adam 优化器进行 40000 次迭代后, 两种模型的数值预测结果以及时间消耗如表 2 所列, GOPINNs 模型的预测结果和绝对误差见图 10.

4 结论与总结

根据上述实验结果不难发现, 尽管 PINNs 模型在一些应用中取得了成功, 但它通常难以精确地逼近复杂 NPDE 的解. 本文监督和分析了 PINNs 模型中神经网络与隐藏层的梯度, 发现在通过自动反向传播训练时会发生隐藏层中的梯度失衡问题.

表 2 两种模型对导数非线性薛定谔方程的一阶怪波解 ((12 式)) 的数值预测结果

Table 2. Numerical prediction results of the first order odd wave solution of derivative nonlinear Schrödinger equation (Eq. (12)) by two models.

神经网络信息	PINNs模型的相对 L_2 误差	GOPINNs模型的相对 L_2 误差	两模型模拟方程求解的时间消耗/s
6层隐藏层 50个神经元	0.465	8.16×10^{-2}	36223.2 24345.7

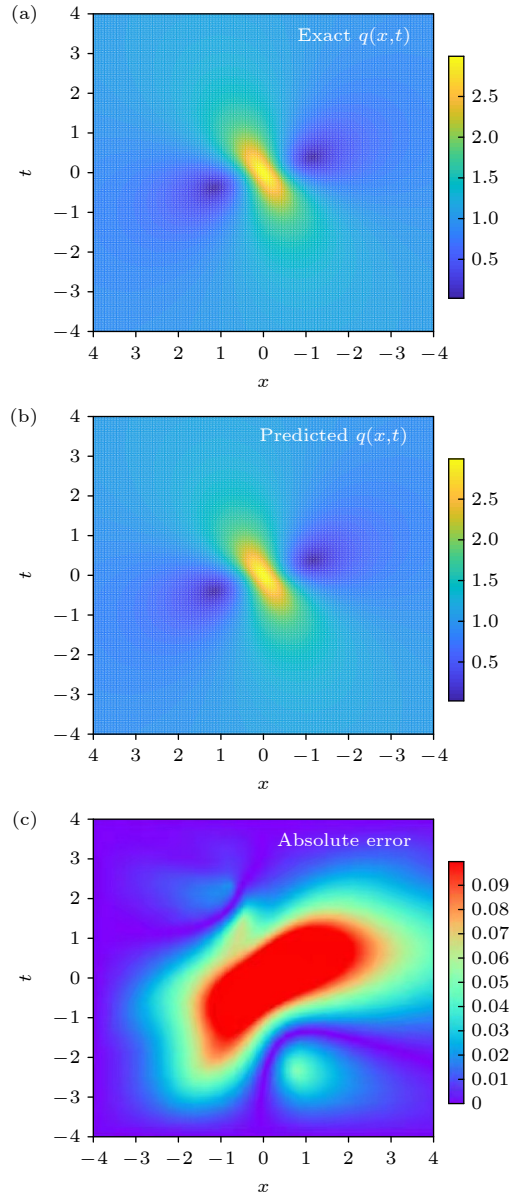


图 10 (a) 导数非线性薛定谔方程的一阶怪波解 ((12 式)); (b) 应用 GOPINNs 模型模拟的预测解; (c) GOPINNs 模型的绝对误差

Fig. 10. (a) First order strange wave solution of derivative nonlinear Schrödinger equation (Eq. (12)); (b) predicted solution simulated by GOPINNs model; (c) absolute error of GOPINNs model.

为了更深入地理解, 定量分析了每一层隐藏层中的梯度动态, 并通过 SGD 算法发现了 PINNs 模型训练的问题. 为了获得更稳定的梯度优化模型, 通过梯度统计平衡模型训练过程中损失函数 MSE 中不同项之间的相互作用, 并提出一种新的神经网络结构, 通过平滑梯度分布来提高泛化和准确性. 以 CH 方程的 peakon 解、导数非线性薛定谔方程的有理波解和怪波解为例进行了验证. 结果表明, 提出的 GOPINNs 模型可以获得精确度比 PINNs 更高的近似解, 预测精度提高将近 10 倍, 并且与 PINNs 模型相比, 节约了超过三分之一的训练时间. 总之, 本文的研究为用机器学习方法来求解非线性偏微分方程提供了新的见解.

尽管最近取得了一些进展, 但必须承认, 我们正处于了解 PINNs 模型局限性的初始阶段, 还有许多问题需要进一步探讨. 1) 给定 NPDE 的梯度波动与相应 PINNs 方法的梯度动力学之间的关系是什么? 2) 如何有效地减少这些梯度波动 (如通过选择不同的损失函数、更有效的神经网络架构等)? 3) 还能做什么继续提高训练期间的泛化和预测精度? 这些有趣的讨论将在今后的工作中进一步探讨.

参考文献

- [1] Linβ T 2001 *Comput. Math. Math. Phys.* **41** 898
- [2] Vulanović R 1988 *Z. Angew. Math. Mech.* **5** 428
- [3] Vulanović R, Nhan T A 2020 *J. Comput. Appl. Math.* **386** 125495
- [4] Gowrisankar S, Srinivasan N 2019 *Appl. Math. Comput.* **346** 385
- [5] Nie F, Wang H, Song Q, Zhao Y, Shen J, Gong M 2022 *Int. J. Multiphase Flow* **152** 104067
- [6] Legendijk L R, Biemond J, et al. 1988 *International Conference on Acoustics* New York, USA, April 11–14, 1988 p60916260
- [7] Simon H 1980 *Cognitive Science* **4** 33
- [8] Busemeyer J 2015 *Cognition* **135** 43
- [9] Sharma N, Jain V, Mishra A 2018 *Procedia Comput. Sci.* **132** 377
- [10] Gu J X, Wang Z H, Jason K, Ma L y, Amir S, Shuai B, et al. 2018 *Pattern Recognit.* **77** 354
- [11] He K, Zhang X, Ren S, Sun J 2016 *Las Vegas Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* Las Vegas, USA, June 27–30, 2016 p770
- [12] Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, et al 2016 *arXiv: 1609.03499 [cs.SD]*
- [13] Heaton J, Goodfellow I, Bengio Y, Courville A 2018 *Genet Program Evolvable Mach.* **19** 305
- [14] Alipanahi B, Delong A, Weirauch T M, Frey J B 2015 *Nat. Biotechnol.* **33** 831
- [15] Han J, Jentzen A, Weinan E 2018 *Proc. Natl. Acad. Sci.* **115** 8505
- [16] Rudy H S, Brunton L S, Proctor L J, Kutz N 2017 *Sci. Adv.* **3** e1602614
- [17] Raissi M, Karniadakis G E 2018 *J. Comput. Phys.* **357** 125
- [18] Weinan E, Han J Q, Jentzen A 2017 *Commun. Math. Stat.* **5** 349
- [19] Sirignano J, Spiliopoulos K 2018 *J. Comput. Phys.* **375** 1339
- [20] Raissi M, Perdikaris P, Karniadakis G E 2019 *J. Comput. Phys.* **378** 686
- [21] Jagtap A D, Kharazmi E, Karniadakis G E 2020 *Proc. R. Soc. A* **476** 20200334
- [22] Revanth M, Susanta G 2021 *arXiv: 2106.07606 [math.NA]*
- [23] Li J, Chen Y 2020 *Commun. Theor. Phys.* **72** 105005
- [24] Li J, Chen Y 2020 *Commun. Theor. Phys.* **72** 115003
- [25] Li J, Chen Y 2021 *Commun. Theor. Phys.* **73** 015001
- [26] Pu J C, Li J, Chen Y 2021 *Chin. Phys. B* **30** 060202
- [27] Pu J C, Li J, Chen Y 2021 *Nonlinear Dyn.* **105** 1723
- [28] Pu J C, Chen Y 2022 *Chaos, Solitons Fractals* **160** 112182
- [29] Lin S N, Chen Y 2022 *J. Comput. Phys.* **41** 898
- [30] Ling L M, Mo Y F, Zeng D L 2022 *Phys. Lett. A* **421** 127739
- [31] He J S, Wang J L 2022 *Phys. Lett. A* **452** 128432
- [32] Wang L, Yan Z Y 2021 *Phys. Lett. A* **404** 127408
- [33] Wang L, Yan Z Y 2022 *Phys. Lett. A* **450** 128373
- [34] Fang Y, Wu G Z, Wang Y Y, et al. 2021 *Nonlinear Dyn* **105** 603
- [35] Zhou Z J, Yan Z Y 2021 *Phys. Lett. A* **387** 127010
- [36] Wang L, Yan Z Y 2021 *Physica D* **428** 133037
- [37] Bai Y, Chaolu T, Bilige S 2021 *Nonlinear Dyn.* **105** 3439
- [38] Wu G Z, Fang Y, Dai C Q, et al. 2021 *Chaos, Solitons Fractals* **152** 111393
- [39] Li J H, Li B 2021 *Commun. Theor. Phys.* **73** 125001
- [40] Li J H, Chen J C, Li B 2022 *Nonlinear Dyn.* **107** 781
- [41] Li J H, Li B 2022 *Chaos, Solitons Fractals* **164** 112712
- [42] Fang Y, Wu G Z, Dai C Q, et al. 2022 *Chaos, Solitons Fractals* **158** 112118
- [43] Wu G Z, Fang Y, Dai C Q, et al. 2022 *Chaos, Solitons Fractals* **159** 112143
- [44] Yuan L, Ni Y Q, Deng X Y, Hao S 2022 *J. Comput. Phys.* **462** 111260
- [45] Zeng S J, Zhang Z, Zou Q S 2022 *J. Comput. Phys.* **463** 111232
- [46] Samadi-koucheksaraee A, Ahmadianfar I, Bozorg-Haddad O, et al. 2019 *Water Resour. Manage.* **33** 603
- [47] Marcucci G, Pierangeli D, Conti C 2020 *Phys. Rev. Lett.* **125** 093901
- [48] Kingma D P, Jimmy B 2014 *arXiv: 1412.6980 [cs.LG]*
- [49] Glorot X, Bengio Y 2010 *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics* Chia Laguna Resort, Sardinia, Italy, March 31, 2010 pp249–256
- [50] Camassa R, Holm D 1993 *Phys. Rev. Lett.* **71** 1661
- [51] Metin G, Atalay K 1998 *J. Math. Phys.* **39** 2103
- [52] Takayuki T, Miki W 1999 *Phys. Lett. A* **257** 53
- [53] Xu S W, He J S, Wang L H 2011 *J. Phys. A: Math. Theor.* **44** 305203

SPECIAL TOPIC—Nonlinear system theory and its frontier applications

Solving complex nonlinear problems based on gradient-optimized physics-informed neural networks*

Tian Shi-Fang Li Biao[†]*(School of Mathematics and Statistics, Ningbo University, Ningbo 315211, China)*

(Received 14 December 2022; revised manuscript received 5 January 2023)

Abstract

In recent years, physics-informed neural networks (PINNs) have attracted more and more attention for their ability to quickly obtain high-precision data-driven solutions with only a small amount of data. However, although this model has good results in some nonlinear problems, it still has some shortcomings. For example, the unbalanced back-propagation gradient calculation results in the intense oscillation of the gradient value during the model training, which is easy to lead to the instability of the prediction accuracy. Based on this, we propose a gradient-optimized physics-informed neural networks (GOPINNs) model in this paper, which proposes a new neural network structure and balances the interaction between different terms in the loss function during model training through gradient statistics, so as to make the new proposed network structure more robust to gradient fluctuations. In this paper, taking Camassa-Holm (CH) equation and DNLS equation as examples, GOPINNs is used to simulate the peakon solution of CH equation, the rational wave solution of DNLS equation and the rogue wave solution of DNLS equation. The numerical results show that the GOPINNs can effectively smooth the gradient of the loss function in the calculation process, and obtain a higher precision solution than the original PINNs. In conclusion, our work provides new insights for optimizing the learning performance of neural networks, and saves more than one third of the time in simulating the complex CH equation and the DNLS equation, and improves the prediction accuracy by nearly ten times.

Keywords: physics-informed neural networks, gradient optimization, Camassa-Holm equation, derivative nonlinear Schrödinger equation, peakon solution, rogue wave solution

PACS: 02.30.Ik, 02.60.-x, 05.45.-a

DOI: 10.7498/aps.72.20222381

* Project supported by the National Natural Science Foundation of China (Grant Nos. 12175111, 12235007) and the K. C. Wong Magna Fund in Ningbo University, China.

[†] Corresponding author. E-mail: libiao@nbu.edu.cn



基于梯度优化物理信息神经网络求解复杂非线性问题

田十方 李彪

Solving complex nonlinear problems based on gradient-optimized physics-informed neural networks

Tian Shi-Fang Li Biao

引用信息 Citation: *Acta Physica Sinica*, 72, 100202 (2023) DOI: 10.7498/aps.72.20222381

在线阅读 View online: <https://doi.org/10.7498/aps.72.20222381>

当期内容 View table of contents: <http://wulixb.iphy.ac.cn>

您可能感兴趣的其他文章

Articles you may be interested in

四阶色散非线性薛定谔方程的明暗孤立波和怪波的形成机制

Study on the generation mechanism of bright and dark solitary waves and rogue wave for a fourth-order dispersive nonlinear Schrödinger equation

物理学报. 2020, 69(1): 010502 <https://doi.org/10.7498/aps.69.20191384>

基于物理信息神经网络的光斑质心计算

Calculation of spot centroid based on physical informed neural networks

物理学报. 2022, 71(20): 200601 <https://doi.org/10.7498/aps.71.20220670>

非线性薛定谔方程的高阶分裂改进光滑粒子动力学算法

Numerical study of nonlinear Schrödinger equation with high-order split-step corrected smoothed particle hydrodynamics method

物理学报. 2019, 68(9): 090203 <https://doi.org/10.7498/aps.68.20190169>

弹性管中的怪波

Rogue waves in fluid-filled elastic tube

物理学报. 2020, 69(1): 014701 <https://doi.org/10.7498/aps.69.20191308>

神经网络超参数优化的删除垃圾神经元策略

Junk-neuron-deletion strategy for hyperparameter optimization of neural networks

物理学报. 2022, 71(16): 160501 <https://doi.org/10.7498/aps.71.20220436>

基于图形处理器加速数值求解三维含时薛定谔方程

Numerical solution of three-dimensional time-dependent Schrödinger equation based on graphic processing unit acceleration

物理学报. 2020, 69(23): 234202 <https://doi.org/10.7498/aps.69.20200700>