Solving the one dimensional vertical suspended sediment mixing equation with arbitrary eddy diffusivity profiles using temporal normalized physics-informed neural networks

Article *in* Physics of Fluids · January 2024

DOI: 10.1063/5.0179223

Shaotong Zhang
Ocean University of China

Xi'An Li
Ceyear Technologies Co., Ltd.

Zixi Zhao
Ocean University of China

Jinran Wu
Australian Catholic University

Accepted to Phys. Fluids 10.1063/5.0179223

Solving the one dimensional vertical suspended sediment mixing equation with arbitrary eddy diffusivity profiles using temporal normalized physics-informed neural networks

Shaotong Zhang* (张少同),[1] Jiaxin Deng (邓加新),[2] Xi'an Li (李西安),[3] Zixi Zhao (赵子茜),[1] Jinran Wu (吴金冉),[4] Weide Li (李维德),[2] You-Gan Wang (王友乾),[5] and Dong-Sheng Jeng (郑东生)[6]

[1] *Frontiers Science Center for Deep Ocean Multispheres and Earth System; Key Lab of Submarine Geosciences and Prospecting Techniques, MOE; College of Marine Geosciences, Ocean University of China, Qingdao 266100, China*
[2] *School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, China, PR China*
[3] *Ceyear Technologies Co., Ltd, Qingdao 266555, China*
[4] *Institute for Learning Sciences & Teacher Education, Australian Catholic University, Brisbane, QLD 4001, Australia*
[5] *School of Mathematics and Physics, The University of Queensland, St Lucia 4067, Australia*
[6] *School of Engineering and Built Environments, Griffith University Gold Coast Campus, QLD 4222, Australia*

(*Electronic mail: shaotong.zhang@ouc.edu.cn)

(Dated: 27 December 2023)

Analytical solutions are practical tools in ocean engineering, but their derivation is often constrained by the complexities of the real world. This underscores the necessity for alternative approaches. In this study, the potential of Physics-Informed Neural Networks (PINN) for solving the one-dimensional vertical suspended sediment mixing (settling-diffusion) equation which involves simplified and arbitrary vertical D_s profiles is explored. A new approach of temporal Normalized Physics-Informed Neural Networks (T-NPINN) which normalizes the time component is proposed, and it achieves a remarkable accuracy (Mean Square Error of 10^{-5} and Relative Error Loss of 10^{-4}). T-NPINN also proves its ability to handle the challenges posed by long-duration spatiotemporal models, which is a formidable task for conventional PINN methods. Besides, the T-NPINN is free of the limitations of numerical methods, e.g., the susceptibility to inaccuracies stemming from the discretization and approximations intrinsic to their algorithms, particularly evident within intricate and dynamic oceanic environments. The demonstrated accuracy and versatility of T-NPINN make it a compelling complement to numerical techniques, effectively bridging the gap between analytical and numerical approaches and enriching the toolkit available for oceanic research and engineering.

I. INTRODUCTION

Analytical solutions are practical tools in ocean engineering applications. Analytical solutions to the one-dimensional vertical (1DV) suspended sediment settling-diffusion equation are useful tools for modeling suspended sediment concentration profiles (SSC). In recent years, it has also been widely used to optimize key sediment transport parameters from measured SSC data[1,2].

Existing analytical solutions to the 1DV equation are derived under different assumptions of eddy diffusivity (D_s) profiles along the depth. For example, with constant vertical D_s and settling velocity w_s profiles, the exponential model was given as[3]:

$$C(z,t) = C(0,t)\mathrm{e}^{-\frac{w_s z}{D_s}} \tag{1}$$

where $C(z,t)$ is the suspended sediment concentration at anytime t and any elevation in the water column z, with z=0 representing the sea floor and positive z upwards. $C(0,t)$ is the bottom reference concentration.

With linear vertical D_s profile, the power-law model was given as,

$$C(z,t) = C(0,t)\left(\frac{z}{z_r}\right)^{w_s/ku_{*\max}} \tag{2}$$

where k=0.4 is the von Karman constant. z_r is the reference elevation at which $C(0,t)$ is given. $u_{*\max}$ is the maximum shear friction velocity.

With parabolic vertica D_s profile, the Rouse model[4] was given as:

$$C(z,t) = C(0,t)\left[\frac{z_r/h(1-z/h)}{z/h(1-z_r/h)}\right]^{\delta_T w_s/ku_*} \tag{3}$$

where h is the water depth, $\delta_T = 0.7$ is the Prandtl-Schmidt number[5].

Prandle [6] proposed an analytical solution for sediment concentration time series associated with each erosional event of magnitude M as follows:

$$C(z,t) = \frac{M}{(4\pi D_s t)^{1/2}}\left[e^{-\frac{(z+w_s t)^2}{4D_s t}} + e^{-\frac{(2h+w_s t-z)^2}{4D_s t}}\right] \tag{4}$$

where M is the magnitude of each erosional event, thus the observed SSC time series represents a time integration of all the preceding "events".

Based on the principle of the convolution of impulse response function and the pick-up rates, a new analytical solution to the 1DV governing equation with bottom flux boundary conditions is given by Zhang *et al.* [1] as follows:

$$C(z,t) = \int_{-\infty}^{t}\left[\frac{\mathrm{e}^{-\frac{(z-w_s[t-t'])^2}{4D_s[t-t']}}}{\sqrt{\pi D_s[t-t']}} + \frac{w_s}{2D_s}\operatorname{erfc}\left(\frac{z-w_s[t-t']}{2\sqrt{D_s[t-t']}}\right)\right] p(t')\,\mathrm{d}t'. \tag{5}$$

physically this solution considers all the bottom input, i.e., pick-up rates $p(t')$ in history $(t-t')$ ago and integrates all their remaining effects at the present moment (t), and erfc is the error function.

However, in many complex cases, analytical solutions are not available, so solving these Partial Differential Equations (PDEs) numerically using approximation methods is necessary. Numerical methods such as finite element method (FEM)[7–9], finite difference method (FDM)[10–12] and finite volume method (FVM)[13] have gained favorable achievement. However, numerical methods require discretization and approximations that may lead to inaccuracies, struggle to capture intricate details, long-term behavior, or rapidly changing conditions like oceanic systems. Besides, numerical methods rely on mesh discretization for a given interest domain. However, in engineering applications sometimes only local data is expected rather than the entire domain[14,15]. Therefore, meshless methods that use a set of configuration points without grids have been developed[16–18]. Deep neural networks (DNN), an efficient mesh-free method without discretization for a given interest domain, have drawn more and more attention. Based on their extraordinary universal approximation capacity[19], DNN can numerically solve the ordinary and partial differential equations as well as the inverse problems for complex geometrical domain and high-dimensional cases[20–25]. While these methods are easy to implement and straightforward, their accuracy may deteriorate or not converge for a few configuration points.

With the rapid development of computer science and technology[23,26], Physics-Informed Neural Networks (PINN) dating back to the early 1990s again attracted widespread attention of researchers and have made remarkable achievements for approximating the solution of PDEs by embracing the physical laws with neural networks[27,28]. This method skillfully incorporates the residual of governing equations and the discrepancy of boundary/initial constraints, then formulates a cost function that can be optimized easily via the automatic differentiation in DNN. This makes PINN particularly effective in scenarios where numerical methods encounter challenges posed by intricate geometries, sparse data, or evolving dynamics, underscoring its significance as an innovative and promising tool in ocean science. So far, PINN has been used in solving hydrodynamic equations[29–32], but not yet in suspended sediment transport equations.

In the present paper, the potential of PINN in solving the 1DV suspended sediment settling-diffusion equation with arbitrary vertical D_s profiles is explored. An improved Temporal Nomalized Physics-Informed Neural Networks (T-NPINN) is proposed and compared with analytical solutions, numerical methods as well as traditional PINN and modified PINNs with different normalization strategies. The great potential of T-NPINN in solving the 1DV suspended sediment settling-diffusion equation in complex conditions is demonstrated. T-NPINN has the potential to be an important complement to cases where analytical solutions are not available and numerical models are not applicable, or even a more practical approach compared to numerical models.

Accepted to Phys. Fluids 10.1063/5.0179223

II. METHODOLOGY

A. An new analytical solution for $C(z,t)$ with depth constant D_s

A new analytical solution for $C(z,t)$ to the 1DV governing equation (Eq. (6)) driven by a step increase in bottom reference concentration $C(0,t)$ is obtained in this section. This analytical solution, as an exact solution, is used to test the performance of PINN under the same conditions, as the first step of this article. This scenario describes the response of $C(z,t)$ to a prescribed bottom concentration that appears instantly and is kept thereafter (Figure 1(a)). The physical significance of taking a constant vertical D_s is for the shallow water region where wind generates turbulence at the air-water interface and ocean flow generates turbulence at the water-seabed interface.

1. Governing equation

The 1DV sediment mixing governing equation is given as:

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial z}\left[D_s(z)\frac{\partial C}{\partial z}\right] - w_s\frac{\partial C}{\partial z} \tag{6}$$

where D_s is denoted as a function of z, because in the following examples, the assumption that D_s is a constant is released. Therefore, a universal expression is provided here. The eddy diffusivity tensor should stay inside the first derivative of the Laplacian term. Taking it out will change the effective settling velocity against the one measured[33], though these corrections are typically small, and often neglected[34]. w_s is a constant sedimentation flow or in general an incompressible one. But this method can be easily generalized to compressible flows, in case the velocity is brought in.

2. Boundary and initial conditions

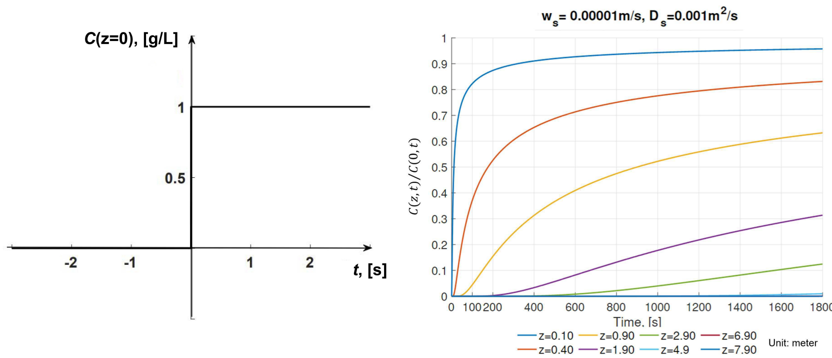
The bottom boundary condition of solving the governing equation is the bottom concentration $C(0,t)$ varies as a step function. The step function is the manifestation of the input quantity jumps from 0 to 1 in a very short period (Figure 1(a)). The step function and impulse function which is the derivative of the step function allow for a convenient description of typical input into a system. Correspondingly, the step response function and impulse response function are convenient descriptions for the output of the dynamic systems. Here, the step function bottom boundary condition physically means that the time series of the bottom concentration at the bed varies as a step function: $C(0,t) = 0$ for t <0; $C(0,t) = 1$ for t >0; and the initial condition is $c(z,t) = 0$ for everywhere.

3. The analytical solution

The analytical solution for $C(z,t)$ driven by a step function bottom concentration input is just the step response function for a system governed by Eq. (6). In other terms, if the SSC at the bed varies as a step function and the suspended sediment concentrations in the overlying column are governed by Eq. (6), one finds the analytical solution for $C(z,t)$ at any time and elevation is:

$$C(z,t) = \frac{C(0,t)}{2}\left\{erfc(\frac{z+w_st}{2\sqrt{D_st}}) + e^{-\frac{w_sz}{D_s}}erfc(\frac{z-w_st}{2\sqrt{D_st}})\right\}. \tag{7}$$

where $C(0,t) = 1$ is the concentration at the bottom. This solution behaves as the example in Figure 1(b), i.e., asymptotic to a steady state for $t \to \infty$. It can be seen that this analytical solution is reasonable because the closer to the bottom (the smaller z), the earlier and more intense the concentration responses.



(a) Step function (bottom input condition)

(b) Step response function (responses at any elevation)

FIG. 1. Diagram of (a) the Step Function (bottom boundary condition) and (b) the Step Response Function (analytical solution for $C(z,t)$). Note: According to the definition of a step function, the y-axis label of panel (a) is concentration in g/L; in panel (b) the y-axis label of the step response function should also be concentration $C(z,t)$ in g/L; here in panel (b) the y-axis label is a ratio because there is $C(0,t)$ on the right side of the equal sign in Eq. (7). When displaying the diagram, $C(0,t)$ is divided to the left side of the equal sign.

B. Description of the traditional Physics-Informed Neural Networks (PINN)

Physics Informed Neural Networks (PINNs) represent a cutting-edge revolutionary approach that seamlessly merges the capabilities of neural networks with the governing equations of physical systems. Developed specifically to address complex PDEs and other intricate physical phenomena, PINN has emerged as a versatile and powerful tool with broad applications across scientific research and engineering disciplines. The foundational structure of PINN, customized for solving PDEs, is illustrated in Figure 2. Within the PINN framework, randomly generated coordinates are input to a fully connected neural network. 50000 random iterations were made in the present work which is crucial and the core reason behind this work, that is, one should use a neural network rather than a finite difference scheme. This network efficiently computes both the temporal and spatial differential terms of the output values. These computed terms are then integrated into the governing equations, allowing for the derivation of a loss function. The ultimate goal is to minimize this loss function through training, to obtain a highly accurate and efficient solution to the underlying PDEs. This unique fusion of machine learning and physics-based modeling equips PINN to excel in tackling complex and nonlinear problems, which may pose challenges for traditional numerical techniques.

Consider a system of parameterized PDEs given by:

$$
\begin{aligned}
&\mathbb{N}_{\boldsymbol{\lambda}}[\hat{u}(\boldsymbol{x},t)] = \hat{f}(\boldsymbol{x},t), && \boldsymbol{x} \in \Omega, t \in (t_0,T] \\
&\mathbb{B}\hat{u}(\boldsymbol{x},t) = \hat{g}(\boldsymbol{x},t), && \boldsymbol{x} \in \partial\Omega, t \in [t_0,T] \\
&\hat{u}(\boldsymbol{x},t_0) = \hat{h}(\boldsymbol{x}), && \boldsymbol{x} \in \Omega
\end{aligned}
\tag{8}
$$

where $\mathbb{N}_{\boldsymbol{\lambda}}$ represents the linear or nonlinear differential operator with parameters $\boldsymbol{\lambda}$, and $\mathbb{B}$ denotes the boundary operators such as Dirichlet, Neumann, and Robin. Ω and $\partial\Omega$ refer to the region of interest and its boundary. The region here can be one-dimensional, two-dimensional, or even multidimensional, but the context targeted in this article is one-dimensional; x is the spatial distance which is elevation z in the present case. In the general PINN approach, a DNN model can be substituted as the solution to PDEs (Eq. (8)). The optimal solution can then be obtained by minimizing the following loss function:

$$
L = Loss_{PDE} + \omega_1 Loss_{IC} + \omega_2 Loss_{BC} \tag{9}
$$

Accepted to Phys. Fluids 10.1063/5.0179223

with

$$\begin{aligned} Loss_{PDE} &= \frac{1}{N_P}\sum_{i=1}^{N_P}\left|\, \mathbb{N}_{\boldsymbol{\lambda}}[\hat{u}_{NN}(\boldsymbol{x}^i,t^i)] - \hat{f}(\boldsymbol{x}^i,t^i)\right|^2 \\ Loss_{BC} &= \frac{1}{N_B}\sum_{i=1}^{N_B}\left|\mathbb{B}\hat{u}_{NN}\left(\boldsymbol{x}^i,t\right) - \hat{g}(\boldsymbol{x}^i,t)\right|^2 \\ Loss_{IC} &= \frac{1}{N_I}\sum_{i=1}^{N_I}\left|\hat{u}_{NN}(\boldsymbol{x}^i,t_0) - \hat{h}(\boldsymbol{x}^i)\right|^2 \end{aligned} \tag{10}$$

where $\omega_1>0$ and $\omega_2>0$ are the loss weighting coefficients for distinct boundaries that are set as fixed values manually in implementation according to experiences. $Loss_{PDE}$, $Loss_{IC}$, and $Loss_{BC}$ represent the residuals of the governing equations, the loss associated with the given initial conditions, and the loss related to the prescribed boundary conditions, respectively. Thus, solving a partial differential equation is transformed into an optimization problem for a neural network.

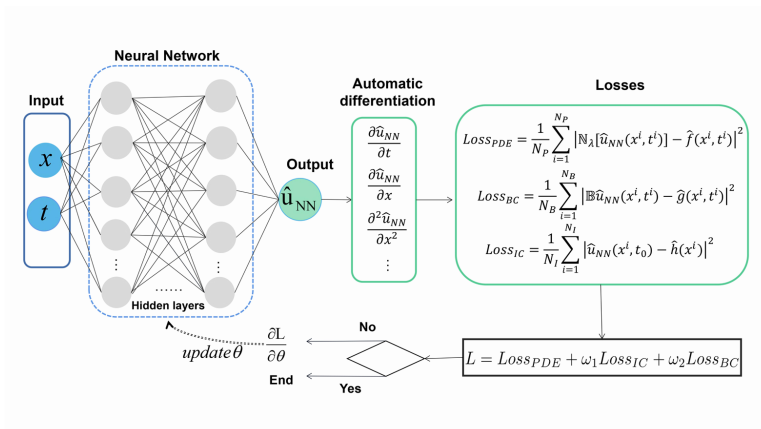


FIG. 2. Schematic diagram of the Physics-Informed Neural Network (PINN) structure.

The PINN method is different from traditional finite difference methods that discretize the domain equidistantly. PINN randomly samples the solution area, not equidistant, and resamples every iteration. Therefore, when the number of iterations is sufficient, enough points are sampled thus the accuracy is high. This article iterates 50000 times. In the present work, we know the exact expression of the initial and boundary conditions, and this allows us to perform a bootstrap over the whole continuous set. In other situations, one might not know the expressions but have a discrete measurement set. In such cases, one would need to do some interpolation on the boundary conditions first or perform some sort of bagging else.

C. Normalized PINN with different temporal or spatial normalization strategies

In this section, we present the innovative concept of a Normalized Physics-Informed Neural Network (NPINN), which incorporates temporal and/or spatial normalization strategies. This pioneering approach aims to address challenges associated with solving Settling-Diffusion Equations (SDEs) conveniently in ocean engineering.

Temporal Normalized PINN (T-NPINN). The first pivotal augmentation introduced in NPINN involves a normalized temporal component. By subjecting the time variable to a normalization process, we adeptly mitigate challenges attributed

Accepted to Phys. Fluids 10.1063/5.0179223

to protracted temporal dependencies, i.e.,

$$\tilde{t} = \frac{t}{T}. \tag{11}$$

Then the governed equation of Eq. (8) can be expressed by

$$\mathbb{N}_{\boldsymbol{\lambda}}[\hat{u}(\boldsymbol{x},\tilde{t})] = \hat{f}(\boldsymbol{x},\tilde{t}), \quad \boldsymbol{x} \in \Omega, \tilde{t} \in (t_0/T, 1]. \tag{12}$$

This strategic intervention empowers the neural network to adeptly encapsulate and model the intricate temporal intricacies inherent to SDEs. Notably, this temporal normalization holds the potential to profoundly enhance the fidelity of SDE representation across expansive time intervals. This methodological refinement not only ensures the accuracy of the neural network's predictions but also contributes to the overall efficiency of SDE solution strategies in ocean engineering domains.

Spatial Normalized PINN (S-NPINN). Additionally, we introduce the Spatial Normalized PINN (S-NPINN), isolating and analyzing the spatial intricacies, allowing for an in-depth exploration of their influence on the SDE solution process. By further scaling the spatial variable into the unit interval $[0,1]^d$ where d is the dimension, as this article focuses on a one-dimensional vertical problem, $d = 1$, i.e.,

$$\tilde{\boldsymbol{x}} = \frac{\boldsymbol{x} - \min\Omega}{\max\Omega - \min\Omega}. \tag{13}$$

Temporal and Spatial Normalized PINN (ST-NPINN). The last strategy is the Temporal and Spatial Normalized PINN (ST-NPINN). Hence, the governed equation of Eq. (8) can be written in terms of the scaled and characteristic quantities:

$$\mathbb{N}_{\boldsymbol{\lambda}}[\hat{u}(\tilde{\boldsymbol{x}},\tilde{t})] = \hat{f}(\tilde{\boldsymbol{x}},\tilde{t}), \quad \tilde{\boldsymbol{x}} \in [0,1]^d, \tilde{t} \in (t_0/T, 1]. \tag{14}$$

The corresponding three types of normalized 1DV governing equations are described as follows:

$$\frac{1}{T}\frac{\partial C}{\partial \tilde{t}} = \frac{\partial}{\partial z}\left[D_s(z)\frac{\partial C}{\partial z}\right] - W_s\frac{\partial C}{\partial z} \tag{15}$$

and

$$\frac{\partial C}{\partial t} = \frac{1}{X^2}\frac{\partial}{\partial \tilde{z}}\left[D_s(\tilde{z})\frac{\partial C}{\partial \tilde{z}}\right] - W_s\frac{W_s}{X}\frac{\partial C}{\partial \tilde{z}} \tag{16}$$

with the length of the domain $X = \max\Omega - \min\Omega$, in the present case X is the whole water depth h, and

$$\frac{1}{T}\frac{\partial C}{\partial \tilde{t}} = \frac{1}{X^2}\frac{\partial}{\partial \tilde{z}}\left[D_s(\tilde{z})\frac{\partial C}{\partial \tilde{z}}\right] - \frac{W_s}{X}\frac{\partial C}{\partial \tilde{z}}. \tag{17}$$

This combined temporal and spatial normalization paradigm enriches the accuracy, efficiency, and interpretability of SDE solutions, providing a novel avenue for advancing the understanding and application of PINN methodologies in ocean engineering research.

III. PINN TRAINING

The basic common setup of the PINNs used in the present paper is summarized in this section.

A. Model setup

Four candidate PINN models are tested in the present paper.

- *The traditional PINN without normalization*: The introduction of this base model has been detailed in Section II B. Here, the solver employs the standard Physics-Informed Neural Network (PINN) model.
- *Three improved NPINNs with different normalization strategies*: We conducted experiments comparing different normalization strategies. Specifically, we explored three strategies: (1) Temporal Normalized PINN (T-NPINN), (2) Spatial Normalized PINN (S-NPINN), and (3) Spatial-Temporal Normalized PINN (ST-NPINN).

B. Training Setup

- *Network configuration:* All four models discussed in the previous sections share a consistent network configuration, ensuring a fair comparison. For the training phase, we employ a 5-layer fully connected neural network architecture with layer sizes of (100, 150, 80, 80, 50), utilizing the sine activation function for all hidden layers and a linear output layer for the final predictions. To maintain a balanced optimization process, we set $\omega_1 = \omega_2 = 1$ defined as in Eq. (9) according to common experience. The training process utilizes the Adam optimizer[35] with an exponential learning rate strategy. The initial learning rate is set at 0.001, and it decays at 0.0005 every 100 training epochs. All models are trained for 50,000 epochs.

- *Sampling strategy:* In each training epoch, we adopt Latin Hypercube Sampling (LHS) to intelligently select training points across the normalized temporal and spatial domains. Specifically, the number of points taken on the control equation $N_R = 10,000$ (collocation points), the number of points taken on the boundary condition $N_B = 6,000$ (boundary points), and the number of points taken on the initial condition $N_I = 3,000$ (initial points) in each epoch. In other words, we have taken 10000 points for each spacetime within the boundary, 6000 points for each spacetime under boundary conditions, and 3000 points for each spacetime under initial conditions. This ensures the models are well-trained and capable of capturing intricate spatiotemporal patterns. During the testing phase, we employ uniform sampling to gather testing data within the domain Ω (or its boundary $\partial\Omega$).

- *Criterion selection:* To assess the accuracy of the models, we employ the Mean Square Error (MSE) and the Relative Error Loss (REL) as evaluation metrics. These metrics are defined as follows:

$$MSE = \frac{1}{N'} \sum_{i=1}^{N'} \left(\tilde{u}(x^i,t^i) - u^*(x^i,t^i)\right)^2 \tag{18}$$

  and

$$REL = \frac{\sum_{i=1}^{N'} \left(\tilde{u}(x^i,t^i) - u^*(x^i,t^i)\right)^2}{\sum_{i=1}^{N'} \left(u^*(x^i,t^i)\right)^2} \tag{19}$$

  where $\tilde{u}(x^i,t^i)$ denotes the approximate solution predicted by the neural network, $u^*(x^i,t^i)$ represents the exact/reference solution, and $\{(x^i,t^i)\}_{i=1}^{N'}$ constitutes the set of testing points. The value of N' denotes the total number of testing points.

- *Visualization:* We logged the MSE and REL of the test set at intervals of 1000 steps and depicted their trends graphically. Concurrently, we conducted a comparative analysis by visualizing the 3D plots generated by NPINN's prediction test set alongside the ground truth images. Additionally, we graphically represented the point-wise absolute errors between the prediction outcomes of the four models and the corresponding actual values.

- *Computing power resources:* Our implementation uses PyTorch (version 1.12.1) on a workstation with 256 GB of RAM and a single NVIDIA GeForce GTX 2080Ti GPU with 12 GB of memory. These computational resources enable efficient and accurate training of the NPINN model and facilitate insightful analysis of its performance. The time taken per iteration with the computing power resources is 0.006119 seconds.

The details of the neural network architectures are summarized in Table I.

IV. VALIDATION OF PINNS IN LINEAR SETTLING–DIFFUSION EQUATION

In this section, we first train the four candidate models to address the 1DV equation with a depth constant D_s (linear Settling–Diffusion Equation). To assess their efficacy, we juxtapose their predictions against the benchmark analytical solution in Section II A.

This section presents a comprehensive analysis of the outcomes obtained from the four distinct models. A comparative evaluation is conducted through visual representations (Figure 3), error analysis (Figure 4), and convergence studies (Figure 5) which offer a detailed insight into the performance of the models.

TABLE I. Network Configuration and Training Parameters

Parameter	**Value**
Architecture	5-layer fully connected network
Hidden Layer Sizes	(100, 150, 80, 80, 50)
Activation Functions	Sine (hidden layers), Linear (output layer)
Training Point Selection	Latin Hypercube Sampling (LHS)
Epochs	50000
Learning Rate	Exponential Decay
Initial Learning Rate	0.001
Learning Rate Decay Rate	0.0005 (every 100 epochs)
Collocation points	10000
Boundary points	6000
Initial points	3000

The error heatmap of T-NPINN demonstrates a broader and more distributed pattern in contrast to the other three models, particularly evident in the extended range of the blue area. This distinct feature signifies that T-NPINN exhibits enhanced training accuracy. The normalization of time variables, a defining characteristic of T-NPINN, is attributed to this improvement, effectively accommodating intricate spatiotemporal dynamics. An intriguing observation is the descending trend of point-wise errors, illustrating an ascending order of accuracy: PINN, S-NPINN, ST-NPINN, and T-NPINN (Figure 4).

Convergence analysis, as depicted in Figures 5(a) and 5(b), reveals that T-NPINN achieves a faster and more stable convergence rate than the other PINN models. This expedited convergence can be attributed to the incorporation of time normalization within the T-NPINN architecture, which has been detailed in Section II C. The inherent ability of this normalization technique to alleviate potential long-term dependencies and temporal evolution within neural networks contributes significantly to T-NPINN's superior performance.

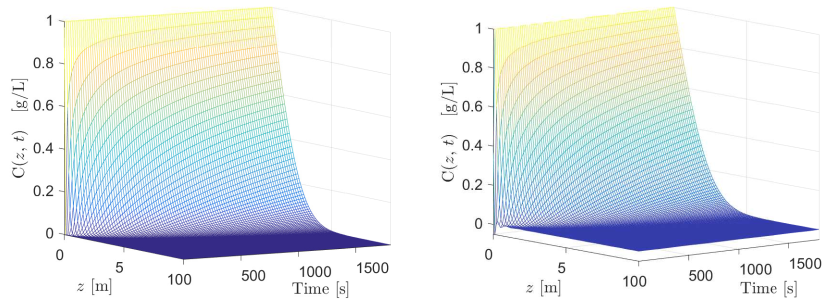
The quantitative assessment outlined in Table II underscores T-NPINN's prowess. The MSE for T-NPINN exhibits a remarkable reduction than the other PINN models, reflecting its ability to approximate solutions more accurately. Similarly, REL for T-NPINN is substantially diminished, underscoring its exceptional proficiency in tackling intricate and long-duration spatiotemporal Settling-Diffusion equations.

In summary, including the time variable normalization within T-NPINN yields substantial improvements in training accuracy and convergence speed, especially when addressing intricate spatiotemporal Settling-diffusion equations. The normalization process inherently enhances the network's precision, with T-NPINN exhibiting the highest accuracy when solely normalizing the time variable.

The small gap between the T-NPINN and the analytic solution demonstrates the ability of the T-NPINN to approximate the exact solution, which validates the ability of T-NPINN in this case.

TABLE II. Performance of different models in constant D_s scenario

	MSE	REL
PINN	3.082×10^{-1}	4.879
S-NPINN	1.006×10^{-2}	1.592×10^{-1}
T-NPINN	4.078×10^{-5}	6.455×10^{-4}
ST-NPINN	1.360×10^{-4}	2.168×10^{-3}
FDM	4.000×10^{-6}	5.900×10^{-5}



(a) Analytical Solution (Eq. (7))

(b) T-NPINN Solution

FIG. 3. Comparation between the (a) analytical solution, and (b) T-NPINN solution with constant $C(0,t) = 1$ g/L at the bottom and depth constant D_s.

V. APPLICATION OF PINNS IN NON-LINEAR SETTLING–DIFFUSION EQUATION

The validated PINN models from Section IV are used for predicting the SSC distribution in more complicated scenarios, i.e., the linear, parabolic, and arbitrary depth distribution of D_s where analytical solutions are unavailable.

A. Scenario 1: Linear depth distribution of D_s

1. PINN settings

In this scenario, we investigate the behavior of the proposed method when the eddy diffusivity D_s follows a linear depth distribution, i.e., $D_s(z) = 0.0001z$. The spatial and temporal domains, as well as the boundary conditions, remain the same as those in Section IV. Since no analytical solution is available for comparison, we employ the finite difference method (FDM) as a benchmark, which has shown good performance in the constant D_s scenario as shown in Table II (The MSE and REL of T-NPINN and FDM are at the similarly highest level). All the model training settings are identical to Section IV (Table I).

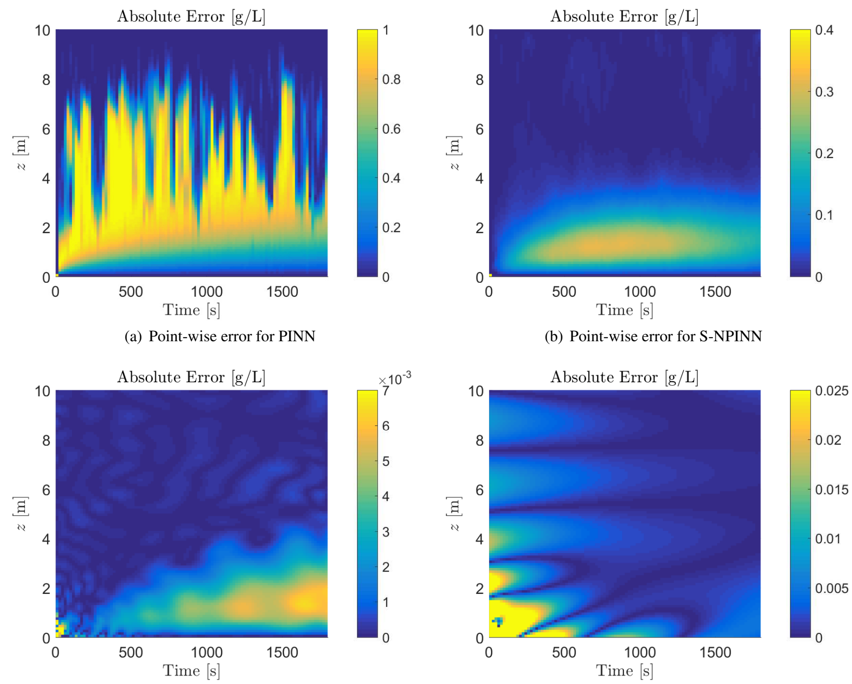
2. Results

A comparative evaluation is conducted through visual representations (Figure 6). The point-wise error of both PINN and NPINN models is illustrated in Figure 7, considering the linear depth distribution of D_s. Additionally, Figure 8 shows the MSE and REL curves during training for all the models. The key results are summarized in Table III and detailed discussions will be made in Section VI.

B. Scenario 2: Parabolic depth distribution of D_s

1. PINN settings

Taking the D_s in Eq. (6) as $D_s(z) = 0.001(z/5-1)^2$, i.e., the eddy diffusivity $D_s(z)$ is a parabolic function of z. The range of spatial and time domain and the boundary conditions are identical to those in Section IV. Since there is no



(a) Point-wise error for PINN

(b) Point-wise error for S-NPINN

(c) Point-wise error for T-NPINN

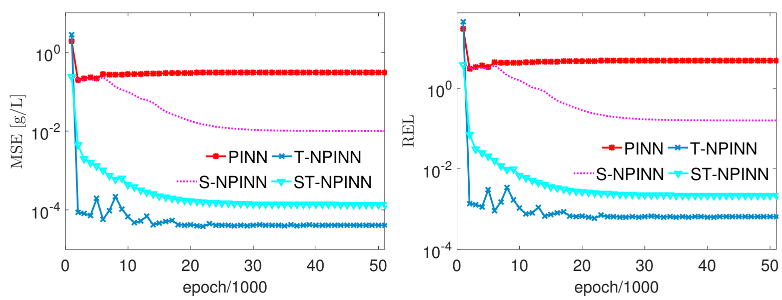
(d) Point-wise error for ST-NPINN

FIG. 4. Point-wise error of 4 candidate models with constant $C(0,t) = 1$ g/L at the bottom and depth constant D_s. (a) For PINN. (b) For S-NPINN. (c) For T-NPINN. (d) For ST-NPINN.

theoretical solution to compare, the FDM is again used to compare. All the model training settings are identical to Section IV (Table I).

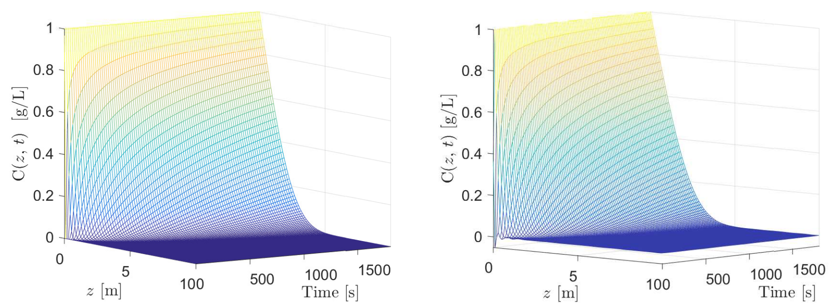
2. Results

A comparative evaluation is conducted through visual representations (Figure 9). The point-wise error of both PINN and NPINN models is illustrated in Figure 10, considering the parabolic depth distribution of D_s. Additionally, Figure 11 shows the MSE and REL curves during training for all the models. The key results are summarized in Table III and detailed discussions will be made in Section VI.



(a) Mean square error of the 4 candidate PINN models (b) Relative error of the 4 candidate PINN models

FIG. 5. The test errors curve during training with constant $C(0,t) = 1$ g/L at the bottom and depth constant D_s. (a) Mean square error of the 4 candidate PINN models. (b) Relative error of the 4 candidate PINN models.



(a) FDM Solution (b) T-NPINN Solution

FIG. 6. Comparison between the (a) FDM and (b) T-NPINN solutions with constant bottom $C(0,t) = 1$ g/L and linear depth distribution of D_s.

C. Scenario 3: Arbitrary (exponential) depth distribution of D_s

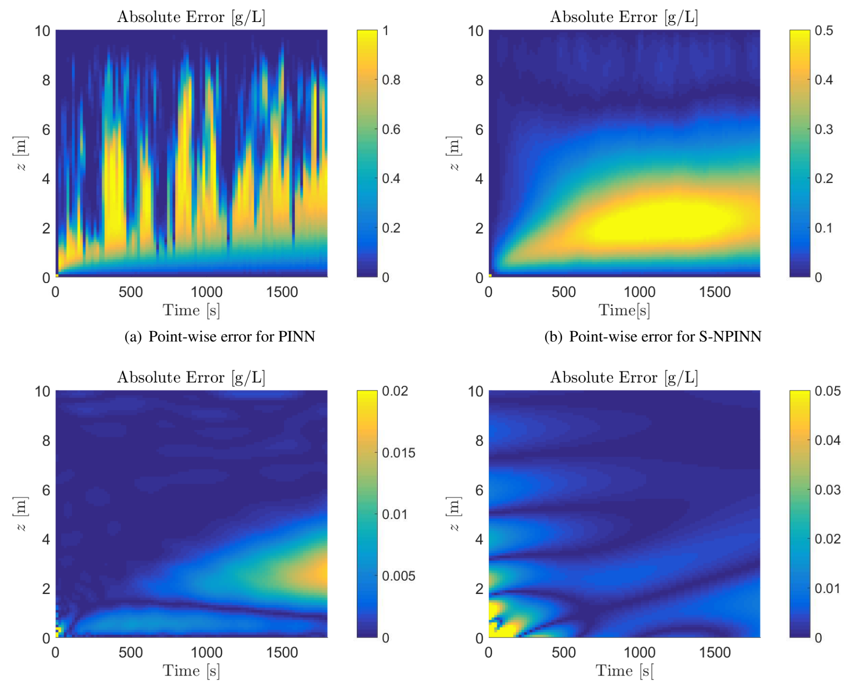
1. PINN settings

Taking the D_s in Eq. (6) as $D_s(z) = 0.001\exp(-z)$, i.e., the eddy diffusivity $D_s(z)$ is a exponential function of z. The range of spatial and time domains and the boundary conditions are identical to those in Section IV. We also take the FDM solution as the benchmark. All the model training settings are identical to Section IV (Table I).

2. Results

A comparative evaluation is conducted through visual representations (Figure 12). The point-wise error of both PINN and NPINN models is illustrated in Figure 13, considering the arbitrary depth distribution of D_s. Additionally, Figure 14 shows the MSE and REL curves during training for all the models. The key results are summarized in Table III and detailed

Accepted to Phys. Fluids 10.1063/5.0179223



(a) Point-wise error for PINN

(b) Point-wise error for S-NPINN

(c) Point-wise error for T-NPINN

(d) Point-wise error for ST-NPINN

FIG. 7. Point-wise error of the 4 candidate models with constant bottom $C(0,t) = 1$ g/L and linear depth distribution of D_s. (a) For PINN. (b) For S-NPINN. (c) For T-NPINN. (d) For ST-NPINN.

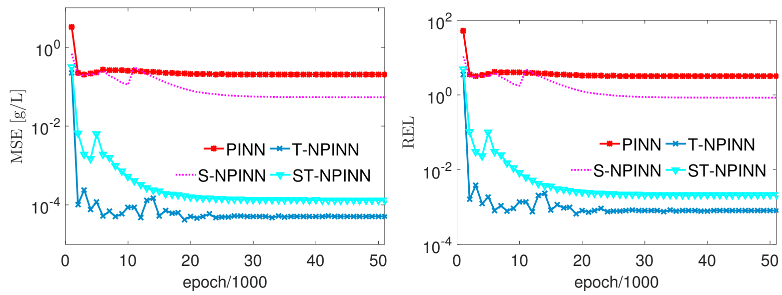
discussions will be made in Section VI.

VI. DISCUSSION

A. The optimal PINN model

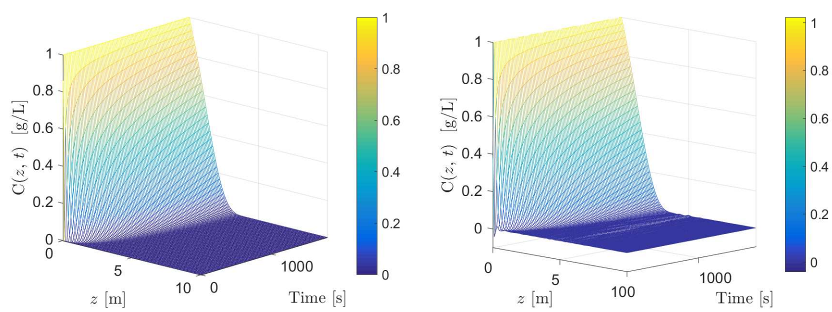
The quantitative assessment outlined in Table III underscores T-NPINN's prowess. The MSE for T-NPINN exhibits a remarkable reduction in compared to the traditional PINN, reflecting its ability to approximate solutions more accurately. Similarly, the REL for T-NPINN is substantially diminished, underscoring its exceptional proficiency in tackling intricate and long-duration spatiotemporal Settling-Diffusion equations.

From the visual representations, error analysis, and convergence plots, it can be seen that T-NPINN demonstrates a broader and more distributed pattern in contrast to the other three models. An intriguing observation is the descending trend of point-wise errors, illustrating an ascending order of accuracy: PINN, S-NPINN, ST-NPINN, and T-NPINN. The time normalization, a defining characteristic of T-NPINN, is the most efficient improvement, effectively accommodating intricate spatiotemporal dynamics.



(a) Mean square error of the 4 candidate PINN models (b) Relative error of the 4 candidate PINN models

FIG. 8. The test errors curve during training the 4 candidate models with constant bottom $C(0,t) = 1$ g/L and linear depth distribution of D_s. (a) Mean square error of the 4 candidate PINN models. (b) Relative error of the 4 candidate PINN models.



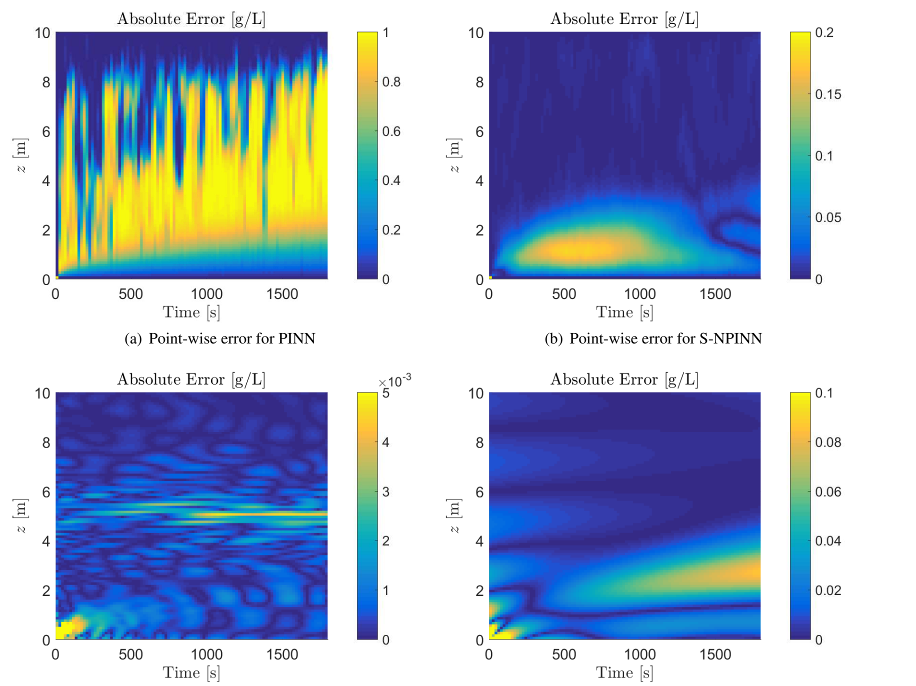
(a) FDM Solution (b) T-NPINN Solution

FIG. 9. Comparison between the (a) FDM and (b) T-NPINN solutions with constant bottom $C(0,t) = 1$ g/L and parabolic depth distribution of $D_s(z)$.

T-NPINN achieves a faster and more stable convergence rate than the conventional PINN model. This expedited convergence can be attributed to the incorporation of time normalization within the T-NPINN architecture which has been detailed in Section II C. The inherent ability of this normalization technique to alleviate potential long-term dependencies and temporal evolution within neural networks contributes significantly to T-NPINN's superior performance.

Although the spatial (S-NPINN) and spatiotemporal normalizations (ST-NPINN) don't seem to be the best approaches, they are maintained in this paper, because we were trying to test and find the optimal normalization strategies. At present, PINN calculations are generally carried out on a normalized domain, with PINN calculations being carried out on a maximum range of [0-4][36]. The present work conducted PINN calculations for 1800 s by normalizing the independent variables, so it is necessary to explore different normalization schemes, and the result is that T-NPINN performs the best. We believe this is also useful information that other more complex solutions are unnecessary.

In summary, introducing the time variable normalization within T-NPINN yields substantial improvements in training accuracy and convergence speed, especially when addressing intricate spatiotemporal SDEs. The normalization process inherently enhances the network's precision, with T-NPINN exhibiting the highest accuracy when solely normalizing the time variable.



(a) Point-wise error for PINN (b) Point-wise error for S-NPINN

(c) Point-wise error for T-NPINN (d) Point-wise error for ST-NPINN

FIG. 10. Point-wise error of the 4 candidate models with constant bottom $C(0,t) = 1$ g/L and parabolic $D_s(z)$. (a) For PINN. (b) For S-NPINN. (c) For T-NPINN. (d) For ST-NPINN.

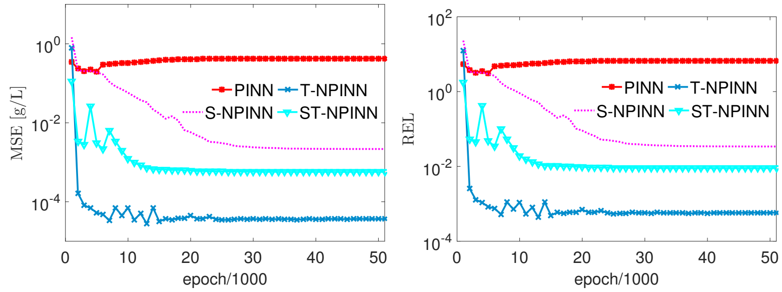
B. Limitations and future works

The current simulation of the working conditions for the bottom boundary for the step function input, the future will develop the bottom boundary for the impulse function input[1,37], to solve the more realistic common working conditions.

In the present case of SSC in the ocean, the solutions are smooth and quite regular throughout the domain. SSC profiles along the water depth over the seabed normally do not show any significant peaks because the mixing of suspended sediments is influenced by gravity and its profile generally increases towards the bottom[1]. Besides, marine sedimentologists care more about the average suspended sediment concentration profile in estimating the sediment transport fluxes[38,39]. Our model starts with simple settling velocity fields, and the application range is relatively narrower.

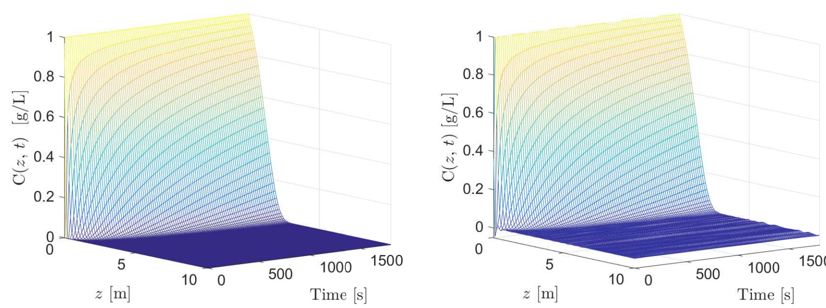
However, the mass transport problem is not only related to settling or mixing in the ocean[40–42], the equation we try to solve appears in other fields such as atmosphere, charged particle transport, etc. In other cases like porous media, well-behaved enough concentration fields might not hold anymore. Solutions may have strong oscillations or one or more peaks. This is a limitation for accurate modeling in the ocean and needs further investigation. It is planned that for more realistic cases with oscillations, peaks, and impulses, the modeling can be tried through a multi-scale neural network (MscaleDNN)[43,44]. MsacleDNN is an excellent model for multi-scale problems with oscillations, peaks, and impulses by converting the high-frequency component into a low-frequency space[45–47].

Besides, eddy diffusivity might be stiff even for good-looking carrier flows (e.g. cell flows)[48,49]. with (locally) anoma-



(a) Mean square error of the 4 candidate PINN models (b) Relative error of the 4 candidate PINN models

FIG. 11. The test errors curve during training the 4 candidate models with constant bottom $C(0,t) = 1$ g/L and parabolic depth distribution of D_s. (a) Mean square error of the 4 candidate PINN models. (b) Relative error of the 4 candidate PINN models.



(a) FDM Solution (b) T-NPINN Solution

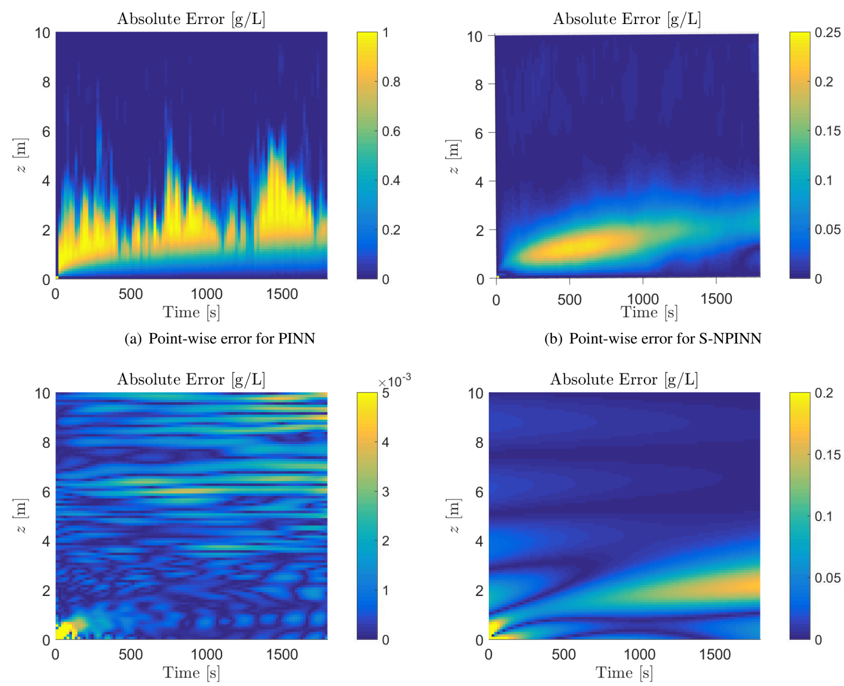
FIG. 12. Comparation between the (a) FDM and (b) T-NPINN solutions with constant bottom $C(0,t) = 1$ g/L and arbitrary (exponential) depth distribution of D_s.

lous diffusion, future works will be done through robust loss functions and Fourier feature network with domain decomposition technique (DFFN). The robust loss function will suppress effectively the adverse effect of abnormal value for the neural network. DFFN model will focus on the local domain with abnormal values and improve the performance of neural networks[50–52].

Finally, many approaches have been proposed to tackle transport problems by encapsulating small and mesoscale transport by carrier flow in the eddy diffusivity, especially because the carrier flow can be turbulent/stochastic and so the settling velocity might have these components too when analyzed at small enough scales[53]. Further investigations into the different scales are also a meaningful generalization.

VII. CONCLUSIONS

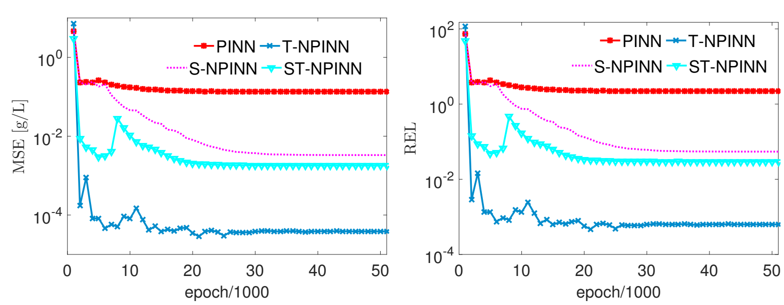
Analytical solutions are practical tools in ocean engineering applications. However, their derivation is often constrained by the complexities of the real world. This underscores the necessity for alternative approaches. In this study, the potential



(a) Point-wise error for PINN (b) Point-wise error for S-NPINN

(c) Point-wise error for T-NPINN (d) Point-wise error for ST-NPINN

FIG. 13. Point-wise error of the 4 candidate models with constant bottom $C(0,t) = 1$ g/L and arbitrary (exponential) $D_s(z)$. (a) For PINN. (b) For S-NPINN. (c) For T-NPINN. (d) For ST-NPINN.



(a) Mean square error of the 4 candidate PINN models (b) Relative error of the 4 candidate PINN models

FIG. 14. The test errors curve during training the 4 candidate models with constant bottom $C(0,t) = 1$ g/L and arbitrary (exponential) depth distribution of D_s. (a) Mean square error of the 4 candidate PINN models. (b) Relative error of the 4 candidate PINN models.

TABLE III. Summary of the model performance in different scenarios

Model	Scenario	$D_s(z)$	MSE	REL
PINN	1	$0.0001z$	2.05×10^{-1}	3.21
	2	$0.001(z/5-1)^2$	4.25×10^{-1}	6.69
	3	$0.001\exp(-z)$	1.35×10^{-1}	2.223
S-NPINN	1	$0.0001z$	5.39×10^{-2}	8.45×10^{-1}
	2	$0.001(z/5-1)^2$	2.16×10^{-3}	3.41×10^{-2}
	3	$0.001\exp(-z)$	3.31×10^{-3}	5.43×10^{-2}
ST-NPINN	1	$0.0001z$	1.32×10^{-4}	2.07×10^{-3}
	2	$0.001(z/5-1)^2$	5.69×10^{-4}	8.96×10^{-3}
	3	$0.001\exp(-z)$	1.79×10^{-3}	2.93×10^{-2}
T-NPINN	1	$0.0001z$	5.11×10^{-5}	8.00×10^{-4}
	2	$0.001(z/5-1)^2$	3.66×10^{-5}	5.76×10^{-4}
	3	$0.001\exp(-z)$	3.83×10^{-5}	6.28×10^{-4}

of Physics-Informed Neural Networks (PINN) in approaching the solution of the one-dimensional suspended sediment settling-diffusion equation with arbitrary D_s profiles is explored. A new approach of Temporal Normalized Physics-Informed Neural Networks (T-NPINN) which normalizes the time component is proposed and achieves a remarkable accuracy level (Mean Square Error of 10^{-5} and the relative error loss of 10^{-4}). Detailed conclusions can be summarized as follows:

(1) An analytical solution to the 1DV governing equation is derived. The analytical expression assumes that the eddy diffusivity D_s and settling velocity w_s are uniform along the depth and the bottom reference concentration is a step function.

(2) The potential of T-NPINN in solving the 1DV suspended sediment convection-diffusion equation with depth uniform D_s profiles is validated by comparing it to the analytic solution. The T-NPINN outperforms other PINN normalization strategies. It achieves a remarkable accuracy level (Mean Square Error of 10^{-5} and the relative error loss of 10^{-4}), validating the efficiency of the T-NPINN method.

(3) The validated T-NPINN is then used in more complex cases of linear, parabolic, and arbitrary (exponential) D_s profiles when analytical expressions are unavailable. It keeps a remarkable accuracy level (Mean Square Error of 10^{-5} and the relative error loss of 10^{-4}), proving its ability for more complex scenarios when analytical solutions are not available.

The present study reveals the potential of T-NPINN, it bridges the gap between analytical and numerical approaches, and it emerges as an indispensable tool for addressing scenarios where numerical models encounter limitations, thereby enriching the toolkit available for oceanic research and engineering.

APPENDIX

To visualize more closely how well the T-NPINN solution matches the numerical solution, the comparison profile of FDM and T-NPINN at different depths are shown in Figure 15. T-NPINN can fit the concentration values at different depths very close to FDM, which also means that our proposed T-NPINN method can be applied to nonlinear SDE when D_s is linearly distributed.

From Figure 16, we can see that when solving the parabolic situation where the distribution of D_s is more complex, T-NPINN can maintain good stability and almost completely coincide with the FDM solution with higher accuracy in terms of image fitting.

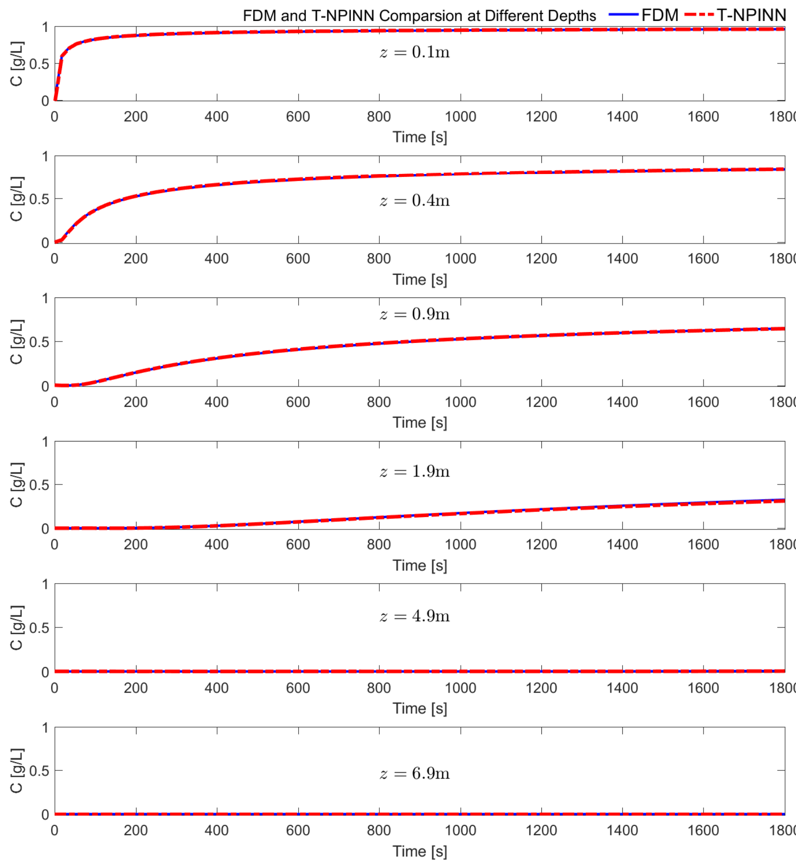


FIG. 15. Comparison of FDM and T-NPINN at Different Depths with a linear depth distribution of D_s

ACKNOWLEDGEMENTS

This work was supported by the Natural Science Foundation of China [grant numbers 42276215, 42130113], the "Chunhui" Program Collaborative Scientific Research Project (202202004), and the Start-up Funding for Young Talent Project of Ocean University of China (Shaotong Zhang).

DECLARATION OF INTERESTS

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

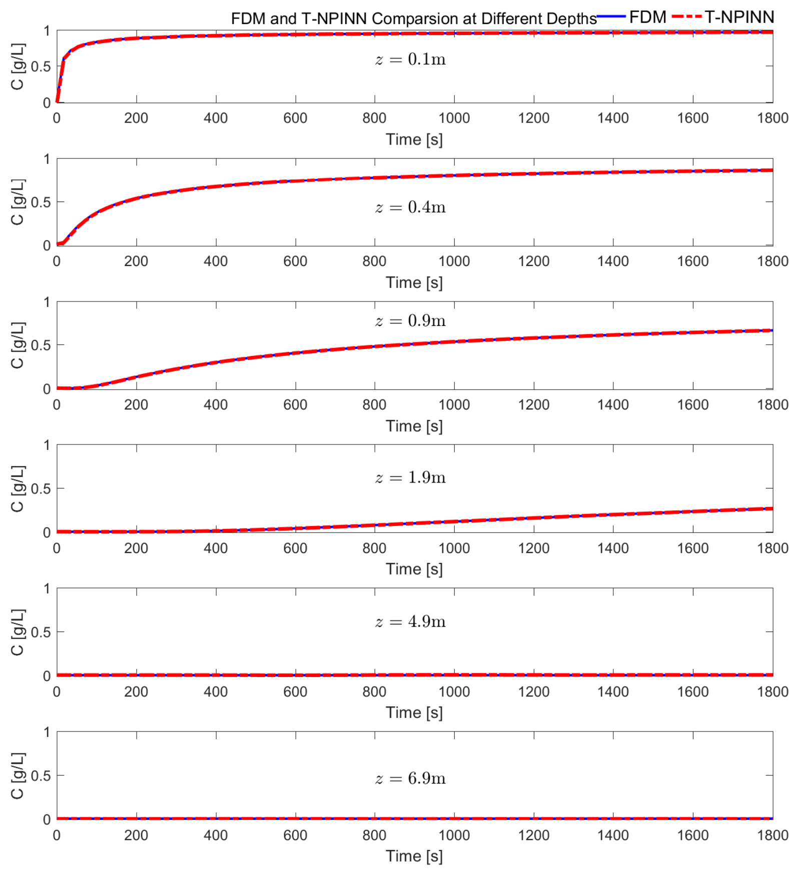


FIG. 16. Comparison of FDM and T-NPINN at Different Depths with a parabolic depth distribution of D_s

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Shaotong Zhang: Conceptualization, Investigation, Formal analysis, Writing - Original Draft, Project administration, Funding acquisition. Jiaxin Deng: Methodology, Formal analysis, Writing - Original Draft. Xi'An Li: Conceptualization, Methodology, Investigation, Writing - Review and Editing. Zixi Zhao: Writing - Review and Editing. Jinran Wu: Writing - Review and Editing. Weide Li: Writing - Review and Editing. You-Gan Wang: Writing - Review and Editing. Dong-Sheng Jeng: Writing - Review and Editing.

[1]S. Zhang, P. Nielsen, P. Perrochet, B. Xu, Y. Jia, and M. Wen, "Derivation of settling velocity, eddy diffusivity and pick-up rate from field-measured suspended sediment concentration profiles in the horizontally uniform but vertically unsteady scenario," Applied Ocean Research **107**, 102485 (2021).
[2]W. C. Edge, M. D. Rayson, N. L. Jones, and G. N. Ivey, "In situ estimation of erosion model parameters using an advection-diffusion model and bayesian inversion," Journal of Advances in Modeling Earth Systems **15**, e2022MS003500 (2023).

[3] P. Nielsen, "Field measurements of time-averaged suspended sediment concentrations under waves," Coastal Engineering **8**, 51–72 (1984).
[4] R. Soulsby, *Dynamics of Marine Sands: A Manual for Practical Applications* (Thomas Telford, 1997).
[5] T. Vijverberg, J. C. Winterwerp, S. G. J. Aarninkhof, and H. Drost, "Fine sediment dynamics in a shallow lake and implication for design of hydraulic works," Ocean Dynamics **61**, 187–202 (2011).
[6] D. Prandle, "Tidal characteristics of suspended sediment concentrations," Journal of Hydraulic Engineering **123**, 341–350 (1997).
[7] E. Burman and A. Ern, "Continuous interior penalty hp-finite element methods for advection and advection-diffusion equations," Mathematics of Computation , 1119–1140 (2007).
[8] Y. Zheng, C. Li, and Z. Zhao, "A note on the finite element method for the space-fractional advection diffusion equation," Computers & Mathematics with Applications **59**, 1718–1726 (2010).
[9] S. Dhawan, S. Kapoor, and S. Kumar, "Numerical method for advection diffusion equation using FEM and B-splines," Journal of Computational Science **3**, 429–437 (2012).
[10] M. Sari, G. Gürarslan, and A. Zeytinoğlu, "High-order finite difference schemes for solving the advection-diffusion equation," Mathematical and Computational Applications **15**, 449–460 (2010).
[11] F. U. Prieto, J. J. B. Muñoz, and L. G. Corvinos, "Application of the generalized finite difference method to solve the advection–diffusion equation," Journal of Computational and Applied Mathematics **235**, 1849–1855 (2011).
[12] R. Cao, S. Wang, G. Xu, and X. Lv, "Characteristics of liquefied soil motion in wavy environment," Physics of Fluids **31** (2019).
[13] C. Ollivier-Gooch and M. Van Altena, "A high-order-accurate unstructured mesh finite-volume scheme for the advection–diffusion equation," Journal of Computational Physics **181**, 729–752 (2002).
[14] S. Zhang, J. Wu, Y. Jia, Y.-G. Wang, Y. Zhang, and Q. Duan, "A temporal lasso regression model for the emergency forecasting of the suspended sediment concentrations in coastal oceans: Accuracy and interpretability," Engineering Applications of Artificial Intelligence **100**, 104206 (2021).
[15] S. Zhang, J. Wu, Y.-G. Wang, D.-S. Jeng, and G. Li, "A physics-informed statistical learning framework for forecasting local suspended sediment concentrations in marine environment," Water Research **218**, 118518 (2022).
[16] I. Boztosun and A. Charafi, "An analysis of the linear advection–diffusion equation using mesh-free and mesh-dependent methods," Engineering Analysis with Boundary Elements **26**, 889–895 (2002).
[17] M. Askari and H. Adibi, "Numerical solution of advection–diffusion equation using meshless method of lines," Iranian Journal of Science and Technology, Transactions A: Science **41**, 457–464 (2017).
[18] J. Zhang, F. Wang, S. Nadeem, and M. Sun, "Simulation of linear and nonlinear advection-diffusion problems by the direct radial basis function collocation method," International Communications in Heat and Mass Transfer **130**, 105775 (2022).
[19] A. Hauptmann and B. Cox, "Deep learning in photoacoustic tomography: current approaches and future directions," Journal of Biomedical Optics **25**, 112903–112903 (2020).
[20] B. Yu *et al.*, "The deep ritz method: a deep learning-based numerical algorithm for solving variational problems," Communications in Mathematics and Statistics **6**, 1–12 (2018).
[21] J. Sirignano and K. Spiliopoulos, "Dgm: A deep learning algorithm for solving partial differential equations," Journal of Computational Physics **375**, 1339–1364 (2018).
[22] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "Neural ordinary differential equations," Advances in Neural Information Processing Systems **31**, 6572–6583 (2018).
[23] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "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 (2019).
[24] Y. Zang, G. Bao, X. Ye, and H. Zhou, "Weak adversarial networks for high-dimensional partial differential equations," Journal of Computational Physics **411**, 109409 (2020).
[25] L. Lyu, Z. Zhang, M. Chen, and J. Chen, "MIM: A deep mixed residual method for solving high-order partial differential equations," Journal of Computational Physics **452**, 110930 (2022).
[26] M. Dissanayake and N. Phan-Thien, "Neural-network-based approximations for solving partial differential equations," Communications in Numerical Methods in Engineering **10**, 195–201 (1994).
[27] N. Chang, Y. Huai, T. Liu, X. Chen, and Y. Jin, "Multi-physical predictions in electro-osmotic micromixer by auto-encoder physics-informed neural networks," Physics of Fluids **35** (2023).
[28] Y. Guo, X. Cao, J. Song, H. Leng, and K. Peng, "An efficient framework for solving forward and inverse problems of nonlinear partial differential equations via enhanced physics-informed neural network based on adaptive learning," Physics of Fluids **35** (2023).
[29] H. Eivazi, M. Tahani, P. Schlatter, and R. Vinuesa, "Physics-informed neural networks for solving reynolds-averaged navier–stokes equations," Physics of Fluids **34** (2022).
[30] N. Wang, Q. Chen, and Z. Chen, "Reconstruction of nearshore wave fields based on physics-informed neural networks," Coastal Engineering **176**, 104167 (2022).
[31] H. Tang, Y. Liao, H. Yang, and L. Xie, "A transfer learning-physics informed neural network (tl-pinn) for vortex-induced vibration," Ocean Engineering **266**, 113101 (2022).
[32] Q. Jiang, X. Wang, M. Yu, M. Tang, B. Zhan, and S. Dong, "Study on pile driving and sound propagation in shallow water using physics-informed neural network," Ocean Engineering **281**, 114684 (2023).
[33] A. Mazzino, S. Musacchio, and A. Vulpiani, "Multiple-scale analysis and renormalization for preasymptotic scalar transport," Physical Review E **71**, 011113 (2005).
[34] S. Boi, A. Mazzino, and G. Lacorata, "Explicit expressions for eddy-diffusivity fields and effective large-scale advection in turbulent transport," Journal of Fluid Mechanics **795**, 524–548 (2016).
[35] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," International Conference on Learning Representations (ICLR) (2015).
[36] R. Mattey and S. Ghosh, "A novel sequential method to train physics informed neural networks for allen cahn and cahn hilliard equations," Computer Methods in Applied Mechanics and Engineering **390**, 114474 (2022).

[37]S. Zhang, P. Nielsen, P. Perrochet, and Y. Jia, "Multiscale superposition and decomposition of field-measured suspended sediment concentrations: Implications for extending 1dv models to coastal oceans with advected fine sediments," Journal of Geophysical Research: Oceans **126**, e2020JC016474 (2021).

[38]S. Zhang, Y. Zhang, J. Xu, L. Guo, G. Li, Y. Jia, L. Qiao, J. Wu, M. Wen, and C. Zhu, "In situ observations of hydro-sediment dynamics on the abandoned diaokou lobe of the yellow river delta: Erosion mechanism and rate," Estuarine, Coastal and Shelf Science **277**, 108065 (2022).

[39]Y. Liu, P.-A. Garambois, A. Terfous, and A. Ghenaim, "Experimental investigations and three-dimensional computational fluid dynamics modeling of sediment transport in tanks influenced by cavities," Physics of Fluids **35** (2023).

[40]Z. Tian, Y. Jia, J. Chen, J. P. Liu, S. Zhang, C. Ji, X. Liu, H. Shan, X. Shi, and J. Tian, "Internal solitary waves induced deep-water nepheloid layers and seafloor geomorphic changes on the continental slope of the northern south china sea," Physics of Fluids **33** (2021).

[41]M. Sanjou and Y. Sugihara, "Experimental study on characteristics of turbulence and sediment transport produced by wind-induced water waves," Physics of Fluids **35** (2023).

[42]M. A. Aleebrahim and M. Jamali, "Laboratory study of instability-driven mixing of fluid mud under surface wave motion," Physics of Fluids **35** (2023).

[43]X.-A. Li, Z.-Q. J. Xu, and L. Zhang, "A multi-scale dnn algorithm for nonlinear elliptic equations with multiple scales," Communications in Computational Physics **28**, 1886–1906 (2020).

[44]Z. Liu, W. Cai, and Z.-Q. J. Xu, "Multi-scale deep neural network (mscalednn) for solving poisson-boltzmann equation in complex domains," Communications in Computational Physics **28**, 1970–2001 (2020).

[45]S. Wang, H. Wang, and P. Perdikaris, "On the eigenvector bias of fourier feature networks: From regression to solving multi-scale pdes with physics-informed neural networks," Computer Methods in Applied Mechanics and Engineering **384**, 113938 (2021).

[46]B. Wang, "Multi-scale deep neural network (mscalednn) methods for oscillatory stokes flows in complex domains," Communications in Computational Physics **28**, 2139–2157 (2020).

[47]X.-A. Li, Z.-Q. J. Xu, and L. Zhang, "Subspace decomposition based dnn algorithm for elliptic type multi-scale pdes," Journal of Computational Physics **488**, 112242 (2023).

[48]F. Cecconi, A. Puglisi, A. Sarracino, and A. Vulpiani, "Anomalous mobility of a driven active particle in a steady laminar flow," Journal of Physics: Condensed Matter **30**, 264002 (2018).

[49]S. Boi, A. Mazzino, and P. Muratore-Ginanneschi, "Eddy diffusivities of inertial particles in random gaussian flows," Physical Review Fluids **2**, 014602 (2017).

[50]Y. Zhu, "Domain decomposition preconditioners for elliptic equations with jump coefficients," Numerical Linear Algebra with Applications **15**, 271–289 (2008).

[51]K. Wang, W. Zhu, and P. Zhong, "Robust support vector regression with generalized loss function and applications," Neural Processing Letters **41**, 89–106 (2015).

[52]K. Li, K. Tang, T. Wu, and Q. Liao, "D3m: A deep domain decomposition method for partial differential equations," IEEE Access **8**, 5283–5294 (2019).

[53]S. Boi, "Exact results on the large-scale stochastic transport of inertial particles including the basset history term," Physics of Fluids **31** (2019).