

Physics-informed neural networks for the Reynolds-Averaged Navier–Stokes modeling of Rayleigh–Taylor turbulent mixing

Meng-Juan Xiao^a, Teng-Chao Yu^a, You-Sheng Zhang^{a,b}, Heng Yong^{a,*}

^a Institute of Applied Physics and Computational Mathematics, Beijing 100094, China

^b Center for Applied Physics and Technology, HEDPS, and College of Engineering, Peking University, Beijing 100871, China

ARTICLE INFO

Keywords:
PINNs
K–L model
Rayleigh–Taylor
Turbulent mixing

ABSTRACT

Recently, Physics-informed neural networks (PINNs) have proven to be an efficient machine-learning method for solving partial differential equations. However, this method can be quite challenging when solving complex problems with shock/material discontinuities or multi-scale features, such as turbulence. In this paper, we propose an improved PINNs framework for solving the Reynolds-averaged Navier–Stokes (RANS) equations for turbulent mixing induced by the Rayleigh–Taylor (RT) instability. The RANS model is based on the closure form of the K–L model. However, the transport equations of the turbulent kinetic energy K and turbulent length scale L are not included and are instead predicted directly by neural networks, thus resulting in an inverse problem. Several modifications are made to the original PINNs to improve its applicability to RT turbulent mixing and accelerate the training optimization process. We first examine the applicability of the PINNs for solving multi-material Euler equations without considering turbulence. Then, PINNs is applied to the RT turbulent mixing problem using training data from the traditional K–L model. The results confirm the ability of the PINNs to predict the entire spatiotemporal field using limited training data. Next, we further train the PINNs using data from the implicit large eddy simulation (ILES), which yields a PINN-based turbulence model that performs better than the traditional K–L model. These results shed light on further applications of PINNs for complex problems, particularly those with limited measurement data and unknown physical models.

1. Introduction

Physics-informed neural networks (PINNs) is a class of deep learning algorithms that embed physical information by considering the loss of partial differential equations (PDEs), initial conditions, boundary conditions and training data if there is missing physics in the PDEs. It was first introduced by Raissi et al. [1] and has been demonstrated to be an easy-to-use method for solving forward and inverse problems related to different types of PDEs. An increasing number of applications cover various issues [2–8]. Recently, this method has been applied to more challenging problems, particularly those with shock/material discontinuities or multi-scale features, such as turbulence. For problems with shock/material discontinuities, several efforts have been made to solve hydrodynamic shock-tube problems, using techniques such as weighted PINNs with domain extension [9], gradient-weighted PINNs [10], and combined sequential-to-sequential and gradient-weighted PINNs [11]. To solve the turbulence problem, the applicability of PINNs for solving the Navier–Stokes equations directly [12] or solving the Reynolds-Averaged Navier–Stokes (RANS) equations of turbulent boundary layers [13] has been discussed.

In this study, we apply PINNs to a new problem, turbulent mixing induced by the Rayleigh–Taylor (RT) instability, because this problem has far-reaching consequences in many natural (e.g., supernovae [14]) and engineering (e.g., inertial confinement fusion [15]) flows [16,17]. This occurs when the perturbed interface of two materials is subjected to acceleration in the direction opposite to the mean density gradient [18,19] and the induced interpenetration and mixing of the materials finally evolve into a turbulent mixing state. Driven by its fundamental and practical importance, numerous studies [16,17,20] have been conducted experimentally or numerically by modeling using different strategies [21–30]. Different effects, including mixing transitions, initial conditions and accelerations, and numerical methods, on RT turbulent mixing have been widely investigated [31–39]. Specifically, for this multi-material problem, the mean quantities, such as the density, total energy and mass fraction, are initially discontinuous and then gradually diffuse (within a certain range) as turbulence develops. However, the turbulent quantities, such as the turbulent kinetic energy and turbulent length scale (extensively discussed in [20]), are initially small quantities as perturbations and then increase by several orders of magnitude, which can be quite challenging for PINNs to capture.

* Corresponding author.

E-mail address: yong_heng@iapcm.ac.cn (H. Yong).

<https://doi.org/10.1016/j.compfluid.2023.106025>

Received 19 November 2022; Received in revised form 26 July 2023; Accepted 7 August 2023

Available online 9 August 2023

0045-7930/© 2023 Elsevier Ltd. All rights reserved.

In this paper, we propose an improved PINNs framework to reflect the characteristics of the RT turbulent mixing problem. We first examine the applicability of PINNs for solving multi-material Euler equations without considering turbulence. Then, PINNs is then applied to solve the RANS equations for RT turbulent mixing, as the RANS models remain the most viable approach for the solution of practical problems [17]. Present RANS model is based on the closure form of the K-L model, as previous [24,40–43], and which our recent works have shown its advantages in describing turbulent mixing flows [44–47]. However, in this study, the transport equations of the turbulent kinetic energy K and turbulent length scale L are not included and are instead predicted directly by neural networks, thus resulting in an inverse problem. The performance of PINNs using training data from both the traditional K-L model and the implicit large eddy simulation (ILES) of Youngs et al. [23,24] is presented. The results show that the entire spatiotemporal field can be reasonably predicted using limited training data.

The remainder of this paper is organized as follows. The PINNs architecture for solving the RT turbulent mixing problem and RANS equations are presented in Section 2. The results and limitation of this method for extrapolation situations are discussed in Section 3. Finally, conclusions are presented in Section 4.

2. Methodology

2.1. Problem setting

The target problem is the classical RT mixing problem. Two one dimensional (1D) RT cases with density ratios of heavy fluid ρ_H to light fluid ρ_L of 3:1 and 20:1 ($\rho_H = 3$ or 20 g/cm³ and $\rho_L = 1$ g/cm³), are set similarly to [24], for comparison with the ILES results of Youngs et al. [23]. The heavy and light fluids are placed in the computational domains of $[-8, 0]$ cm and $[0, 20]$ cm, respectively. The Atwood numbers $A = (\rho_H - \rho_L)/(\rho_H + \rho_L)$ for these two cases are 0.5 (3:1) and 0.9 (20:1). For each case, a constant acceleration g is imposed along the $+x$ direction, with the value set to satisfy $Ag = 1$ cm/s². The heavy and light fluids are both ideal gases with an adiabatic exponent $\gamma = 1.4$ and molecular weight $M = 0.0288$ kg/mol. The computational end time is 10s.

2.1.1. Governing equations of the K-L model

In the present study, the governing equations are based on the closure form of the K-L model, which was recently developed in a series of works [44–47]. A complete 1D form of our latest version is as follows: The Favre-averaged conservation equations for the mixed density $\bar{\rho}$, velocity \bar{u} , energy $\bar{E} \equiv \bar{e} + \bar{u}\bar{u}/2$ (\bar{e} denotes the internal energy), mass species $\bar{Y}_a \equiv \bar{\rho}_a/\bar{\rho}$ of media a , fluctuating/turbulent kinetic energy (TKE) \bar{K}_f , and turbulent length scale \bar{L} of the K-L model (neglecting molecular viscosity) are:

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial \bar{\rho}\bar{u}}{\partial x} = 0 \quad (1)$$

$$\frac{\partial \bar{\rho}\bar{u}}{\partial t} + \frac{\partial \bar{\rho}\bar{u}\bar{u}}{\partial x} + \frac{\partial \bar{\rho}}{\partial x} - \bar{\rho}\bar{g} = \frac{\partial \bar{\tau}_{11}}{\partial x} \quad (2)$$

$$\frac{\partial \bar{\rho}\bar{E}}{\partial t} + \frac{\partial \bar{u}(\bar{\rho}\bar{E} + \bar{p})}{\partial x} - \bar{\rho}\bar{u}\bar{g} = \frac{\partial}{\partial x} \left(\frac{\mu_t}{N_h} \frac{\partial \bar{h}}{\partial x_j} \right) + \bar{u} \frac{\partial \bar{\tau}_{11}}{\partial x} - S_{K_f} \quad (3)$$

$$\frac{\partial \bar{\rho}\bar{Y}_a}{\partial t} + \frac{\partial \bar{\rho}\bar{u}\bar{Y}_a}{\partial x} = \frac{\partial}{\partial x} \left(\frac{\mu_t}{N_Y} \frac{\partial \bar{Y}_a}{\partial x} \right) \quad (4)$$

$$\frac{\partial \bar{\rho}\bar{K}_f}{\partial t} + \frac{\partial \bar{\rho}\bar{u}\bar{K}_f}{\partial x} = \bar{\tau}_{11} \frac{\partial \bar{u}}{\partial x} + \frac{\partial}{\partial x} \left(\frac{\mu_t}{N_k} \frac{\partial \bar{K}_f}{\partial x} \right) + S_{K_f} \quad (5)$$

$$\frac{\partial \bar{\rho}\bar{L}}{\partial t} + \frac{\partial \bar{\rho}\bar{u}\bar{L}}{\partial x} = \frac{\partial}{\partial x} \left(\frac{\mu_t}{N_L} \frac{\partial \bar{L}}{\partial x} \right) + C_L \bar{\rho} \sqrt{2\bar{K}_f} + C_C \bar{\rho} \bar{L} \frac{\partial \bar{u}}{\partial x} \quad (6)$$

where g is a volume force (e.g., gravitation) and $\bar{h} \equiv \bar{e} + \bar{p}/\bar{\rho}$. The Reynolds decomposition, $f \equiv \bar{f} + f'$, and Favre decomposition, $f \equiv$

Table 1

ALL the model coefficients for the present K-L model.

C_C	C_P	C_A	C_B	C_D	C_μ	C_L	N_h	N_k	N_L	N_Y
1/3	2/3	11.2	0.76	0.2	1.19	0.19	0.35	0.43	0.04	0.35

$\bar{f} + f''$, are used to derive the equations, where \bar{f} , \bar{f} , f' , f'' denote the Reynolds averaged, Favre averaged $\bar{f} \equiv \bar{\rho}\bar{f}/\bar{\rho}$, Reynolds fluctuation, and Favre fluctuation components of f , respectively. The equation array is solved by coupling with the equation of state (EOS) $\bar{p}M = \bar{\rho}R\bar{T}$ of a perfect gas, where M is the molar mass, R is the gas constant, and \bar{T} is the Favre averaged static temperature corresponding to the static pressure. According to [48], the assumptions of iso-temperature (i.e., $T = T_1 = \dots = T_a$) and partial-pressure (i.e., $p = \sum p_a$) are used to calculate the EOS of the mixture, and the species-linearly-weighted assumption (i.e., $f = \sum Y_a f_a$) is adopted to calculate the fluid properties of the mixture.

The terms on the right-hand side of Eqs. (1)–(6) are the unclosed terms after the ensemble average, which are modeled using the mean-field under various assumptions. A standard form of the turbulent eddy viscosity is described by the TKE \bar{K}_f and turbulent length scale \bar{L} [40] as follows:

$$\mu_t = C_\mu \bar{\rho} \bar{L} \sqrt{2\bar{K}_f}. \quad (7)$$

The Reynolds stress $\bar{\tau}_{11} \equiv -\bar{\rho}\overline{u'u'}$ is modeled based on the Boussinesq eddy viscosity assumption

$$\bar{\tau}_{11} = \frac{4}{3} \mu_t \frac{\partial \bar{u}}{\partial x} - C_P \bar{\rho} \bar{K}_f \quad (8)$$

The source term S_{K_f} is given by the buoyancy-drag model as follows:

$$S_{K_f} = \bar{\rho} \sqrt{2\bar{K}_f} [C_B C_A \frac{\bar{L}}{\bar{\rho}} \frac{\partial \bar{\rho}}{\partial x} \left(-\frac{1}{\bar{\rho}} \frac{\partial \bar{p}}{\partial x} \right) - 2C_D \bar{K}_f / \bar{L}] \quad (9)$$

where the first part is the buoyancy production and the second part denotes the dissipation.

Therefore, all unclosed terms are described by the mean field, and the 11 model coefficients, i.e., C_A , C_B , C_C , C_D , C_P , C_μ , C_L , N_h , N_k , N_L , and N_Y are listed in Table 1.

2.1.2. Initial conditions and boundary conditions

For the ideal RT mixing, the initial flow field should satisfy the isentropic hydrostatic equilibrium, i.e. $u = 0$ and $p/\rho^\gamma = \text{constant}$. Combining this with the EOS of an ideal gas, we can integrate the momentum equation to derive the initial profiles of the density and the pressure as follows:

$$\bar{\rho}_0(x) = \begin{cases} \bar{\rho}_{0H} \left[1 + \frac{\gamma-1}{\gamma} \frac{\bar{\rho}_{0H}}{\bar{\rho}_{0I}} g(x-x_I) \right]^{\frac{1}{\gamma-1}}, & x < x_I \\ \bar{\rho}_{0L} \left[1 + \frac{\gamma-1}{\gamma} \frac{\bar{\rho}_{0L}}{\bar{\rho}_{0I}} g(x-x_I) \right]^{\frac{1}{\gamma-1}}, & x \geq x_I \end{cases}, \quad (10)$$

$$\bar{p}_0(x) = \begin{cases} \bar{p}_{0I} \left[1 + \frac{\gamma-1}{\gamma} \frac{\bar{\rho}_{0H}}{\bar{\rho}_{0I}} g(x-x_I) \right]^{\frac{\gamma}{\gamma-1}}, & x < x_I \\ \bar{p}_{0I} \left[1 + \frac{\gamma-1}{\gamma} \frac{\bar{\rho}_{0L}}{\bar{\rho}_{0I}} g(x-x_I) \right]^{\frac{\gamma}{\gamma-1}}, & x \geq x_I \end{cases}$$

where $x_I = 0$ is the interface position, the subscript 0 denotes the interface (throughout this paper), $\bar{\rho}_{0I}$ is the interface pressure, $\bar{\rho}_{0H}$ and $\bar{\rho}_{0L}$ denote the densities at $x = 0^-$ (the side of the heavy fluid) and $x = 0^+$ (the side of the light fluid), respectively. The value of $\bar{\rho}_{0I}$ influences the shape of the density profile, and a (comparably) large value is set to approach incompressible limit. The velocity is initialized as zero across the whole field. The mass fraction of the heavy fluid $\bar{Y}_1(x)$ is set to 1 for $x < x_I$ and 0 for $x \geq x_I$. Thus, the initial total energy can be calculated using $\bar{E}_0 = \bar{p}_0/(\bar{\rho}_0(\gamma-1)) + \bar{u}_0\bar{u}_0/2$. \bar{K}_f and \bar{L} are initialized as $\bar{K}_f(0) = 0.004$ and $\bar{L}(0) = 0.001$ inside the perturbed region ($|x| \leq 0.01$) and extremely small values outside, referring to the calculation of the traditional K-L model.

The quantities at the boundaries are set to be constant and equivalent to the boundaries of the initial condition.

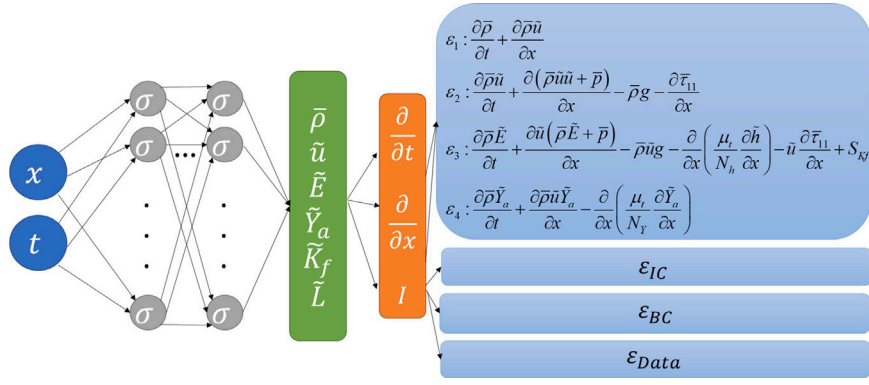


Fig. 1. PINNs framework for solving the RANS equations for a 1D RT turbulent mixing problem.

2.2. PINNs

The architecture of the PINNs for solving 1D RT turbulent mixing problem is shown in Fig. 1. In the set up, the 1D spatial coordinate x and the temporal coordinate t are the two inputs of a fully-connected neural network (FNN), and the six outputs are the mean density $\bar{\rho}$, velocity \bar{u} , computable total energy \bar{E} , mass fraction \bar{Y}_a , turbulent kinetic energy \bar{K}_f , and turbulent length scale \bar{L} . Automatic differentiation (AD) [49] is used to differentiate the outputs with respect to the inputs and formulate the governing RANS equations. We use the open-source machine-learning software framework PyTorch to develop our PINNs model.

To apply PINNs to solve the RT turbulent mixing problem, specific improvements are required to optimize the PINNs framework and accelerate convergence in the training process as follows:

- Normalizing the physical quantities and the PDEs.
- Adding the softplus activation function in the last layer to ensure positive values of physical quantities, such as the density, turbulent kinetic energy, and turbulent length scale.
- Introducing weights to different parts of the loss function.

2.2.1. Normalization

As mentioned previously, the magnitudes of the physical quantities involved in RT problems vary significantly. Thus, a normalization process is required to prevent the gradient from disappearing and balance the magnitudes of the different terms of the loss function to accelerate convergence. Given the normalized reference values of $x, t, \bar{\rho}, \bar{p}, \bar{Y}, M, R, \bar{K}_f, \bar{L}$, i.e. $x_{ref}, t_{ref}, \rho_{ref}, p_{ref}, Y_{ref}, M_{ref}, R_{ref}, K_{ref}$, and L_{ref} , the normalized reference values of the velocity \bar{u} and total energy \bar{E} can be defined as $u_{ref} = \sqrt{p_{ref}/\rho_{ref}}$ and $E_{ref} = u_{ref}^2$. Thus, the normalized governing equations for PINNs are as follows:

$$\frac{\partial \bar{\rho}^*}{\partial t^*} A_{ref} + \frac{\partial \bar{\rho}^* \bar{u}^*}{\partial x^*} = 0 \quad (11)$$

$$\frac{\partial \bar{\rho}^* \bar{u}^*}{\partial t^*} A_{ref} + \frac{\partial \bar{\rho}^* \bar{u}^* \bar{u}^*}{\partial x^*} + \frac{\partial \bar{p}^*}{\partial x^*} - \bar{\rho}^* \bar{g}^* - \frac{\partial \bar{\tau}_{11}^*}{\partial x^*} = 0 \quad (12)$$

$$\begin{aligned} & \frac{\partial \bar{\rho}^* \bar{E}^*}{\partial t^*} A_{ref} + \frac{\partial \bar{u}^* (\bar{\rho}^* \bar{E}^* + \bar{p}^*)}{\partial x^*} - \bar{\rho}^* \bar{u}^* \bar{g}^* - \\ & \frac{\partial}{\partial x^*} \left(\frac{\mu_t^*}{N_h} \frac{\partial \bar{h}^*}{\partial x^*} \right) - \bar{u}^* \frac{\partial \bar{\tau}_{11}^*}{\partial x^*} + S_{K_f}^* = 0 \end{aligned} \quad (13)$$

$$\frac{\partial \bar{\rho}^* \bar{Y}_a^*}{\partial t^*} A_{ref} + \frac{\partial \bar{\rho}^* \bar{u}^* \bar{Y}_a^*}{\partial x^*} - \frac{\partial}{\partial x^*} \left(\frac{\mu_t^*}{N_Y} \frac{\partial \bar{Y}_a^*}{\partial x^*} \right) = 0 \quad (14)$$

where the normalized quantity ϕ^* is defined as $\phi^* = \phi/\phi_{ref}$. The normalized eddy viscosity, Reynolds stress and source term are expressed as follows:

$$\mu_t^* = M_{a_{ref}} B_{ref} C_\mu \bar{\rho}^* \bar{L}^* \sqrt{2 \bar{K}_f^*} \quad (15)$$

$$\bar{\tau}_{11}^* = \frac{4}{3} \mu_t^* \frac{\partial \bar{u}^*}{\partial x^*} - M_{a_{ref}}^2 C_p \bar{\rho}^* \bar{K}_f^* \quad (16)$$

$$\begin{aligned} S_{K_f}^* &= \bar{\rho}^* \sqrt{2 \bar{K}_f^*} [M_{a_{ref}} B_{ref} C_{BCA} \frac{\bar{L}^*}{\bar{\rho}^*} \frac{\partial \bar{\rho}^*}{\partial x^*} \left(-\frac{1}{\bar{\rho}^*} \frac{\partial \bar{\rho}^*}{\partial x^*} \right) - \\ & 2 \left(\frac{M_{a_{ref}}^3}{B_{ref}} \right) C_D \bar{K}_f^* / \bar{L}^*] \end{aligned} \quad (17)$$

Three dimensionless constants appear in the above equations: $A_{ref} = x_{ref}/(u_{ref} t_{ref})$, $M_{a_{ref}} = \sqrt{K_{ref}}/u_{ref}$, and $B_{ref} = L_{ref}/x_{ref}$.

For the investigated cases, $x_{ref} = 20$, $t_{ref} = 10$, $p_{ref} = 6000$, $Y_{ref} = 1$, $M_{ref} = 0.0288$, $R_{ref} = 8.314$, $K_{ref} = 1$, $L_{ref} = 1$, $\rho_{ref} = 1$ for the low-density ratio case, and $\rho_{ref} = 10$ for the high-density ratio case.

2.2.2. Ensuring positive values of the outputs

Note that five of the six outputs are physically non-negative (except for the velocity). Here, to ensure that the PDEs can be numerically solved (e.g., the denominator cannot be zero and the quantity with the root cannot be negative), at least three quantities, i.e. the density, turbulence kinetic energy and turbulence characteristic scale, should be positive. This can be achieved by adding softplus activation functions to these quantities in the final layer.

2.2.3. Weights of the loss function

For the inverse problem, the total loss is the sum of the PDEs loss L_{PDEs} , initial condition loss L_{IC} , boundary condition loss L_{BC} , and supervised loss of the training dataset $L_{Data}^{profile}$ and $L_{Data}^{integral}$, shown as follows:

$$L = \eta_{PDEs} L_{PDEs} + \eta_{IC} L_{IC} + \eta_{BC} L_{BC} + \eta_{Data}^{profile} L_{Data}^{profile} + \eta_{Data}^{integral} L_{Data}^{integral} \quad (18)$$

$$L_{PDEs} = \sum_{i=1}^4 \frac{1}{N_{PDEs}} \sum_{n=1}^{N_{PDEs}} |\epsilon_i^n|^2 \quad (19)$$

$$L_{IC} = \frac{1}{N_{IC}} \sum_{n=1}^{N_{IC}} |\epsilon_{IC}^n|^2 \quad (20)$$

$$L_{BC} = \frac{1}{N_{BC}} \sum_{n=1}^{N_{BC}} |\epsilon_{BC}^n|^2 \quad (21)$$

$$L_{Data}^{profile} = \frac{1}{N_{Data}^{profile}} \sum_{n=1}^{N_{Data}^{profile}} |U_{PINNs}^{profile} - U_{Data}^{profile}|^2 \quad (22)$$

$$L_{Data}^{integral} = \frac{1}{N_{Data}^{integral}} \sum_{n=1}^{N_{Data}^{integral}} |U_{PINNs}^{integral} - U_{Data}^{integral}|^2 \quad (23)$$

where $\eta_{PDEs}, \eta_{IC}, \eta_{BC}, \eta_{Data}^{profile}, \eta_{Data}^{integral}$ are the weighting factors to balance different terms of the loss function and accelerate convergence

in the training process. N_{PDEs} , N_{IC} , N_{BC} , $N_{Data}^{profile}$, $N_{Data}^{integral}$ are the sampling points inside the domain for calculating residuals of the PDEs, on the initial line, on the domain boundary and on the training data points for profiles and integrals, respectively. ϵ_i^n , ϵ_{IC}^n and ϵ_{BC}^n are the residual of the i th governing equation, initial condition and boundary condition at point n , respectively. For the inverse problem using training data, U_{PINNs} denotes the prediction of PINNs and U_{Data} is the given data at the corresponding point. The training data are divided into two categories. One is the profile data $U^{profile}$ of the volume fraction (VF) $\frac{Y_i/M_i}{\sum Y_i/M_i}$ (where Y_i and M_i are the mass fraction and the molecular weight of the i th species, respectively) and TKE at the final time, and the other is the integral data $U^{integral}$, i.e. the growth rates of the integral mixing width (MW) W and the maximum value of the TKE (K_{max}).

There are two rules for setting the weights of the different terms. First, as shown by the numerical experiments of Papados [9], a larger weight for the initial/boundary condition loss should be assigned, and setting it 100 times greater than the weight of the PDEs is optimal for training. This is because the solution of the PDEs depends on the initial and boundary conditions, and the initial loss decreases at a lower rate than the PDEs loss, owing to the discontinuity in the initial state. Hence, assigning a large weight to the initial condition loss forces the neural network to correctly fit the initial condition first so that the solution in the entire time interval is not substantially different from the correct one. Second, for the training data loss, different weights are also introduced for different types of data based on the principle that the weight of data from the primary variables is larger than the weight from the integrals or global variables. This is because if the neural network attempts to learn the growth rate of the maximum value of a variable, e.g., the turbulent kinetic energy, it should learn the profile/shape of the variable first to ensure that there is only one maximum. If the profile or primary variable is not learned correctly, the training of the growth rate of its maximum deviates from the desired track or even terminates the training process. Our numerical experiments showed that setting the weight of the profile loss to 10 times greater than that of the integral loss is optimal for training. Therefore, the final weights are set as $\eta_{PDEs} = 0.1$, $\eta_{IC} = 10.0$, $\eta_{BC} = 10.0$, $\eta_{Data}^{profile} = 10.0$, $\eta_{Data}^{integral} = 1.0$.

2.2.4. Implementation

For each test case below, with no additional illustration, the training points are sampled from the computational domain that has been partitioned into 1001 points in the x direction and 5001 points in t . Each neural network has 9 layers with 30 neurons per layer and uses the tangent (tanh) activation function for non-linear layers. For the training optimizer for stochastic gradient descent, we first use the Adam optimizer [50] with a learning rate of 0.01 for a certain number of iterations, and then switch to L-BFGS [51]. The optimization process of the L-BFGS algorithm is stopped automatically based on the increment tolerance.

3. Results

In this section, we employ PINNs to solve the RT mixing problem.

3.1. Validation for a steady multi-material problem

First, we show the application of PINNs for a steady multi-material problem, in which the heavy fluid is supported by the light fluid, but the interface of the two fluids is not perturbed (no baroclinic generation of vorticity). Thus, RT instability does not occur, i.e., no mixing or turbulence occurs, and the flow is frozen at the initial state. For this case, Eqs. (11)–(14) without the turbulent terms ($-\frac{\partial \bar{\epsilon}^*}{\partial x^*}$, $-\frac{\partial}{\partial x^*} \left(\frac{\mu_t^*}{N_h} \frac{\partial \bar{h}^*}{\partial x^*} \right)$ – $\bar{u}^* \frac{\partial \bar{\epsilon}^*}{\partial x^*} + S_{K_f}^*$, $-\frac{\partial}{\partial x^*} \left(\frac{\mu_t^*}{N_h} \frac{\partial \bar{h}^*}{\partial x^*} \right)$), are solved. The initial and boundary conditions are the same as those described in Section 2.1.2. Since the PDEs and initial/boundary conditions are completely given, no extra

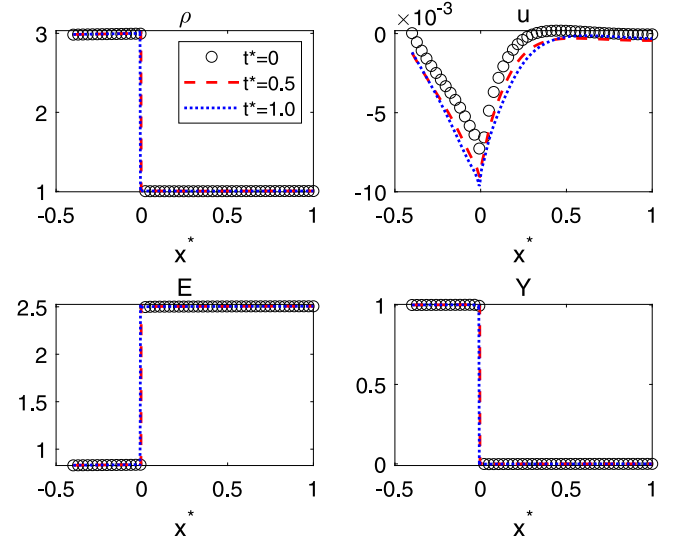


Fig. 2. Results of the present PINNs for solving normalized 1D multi-material Euler equations.

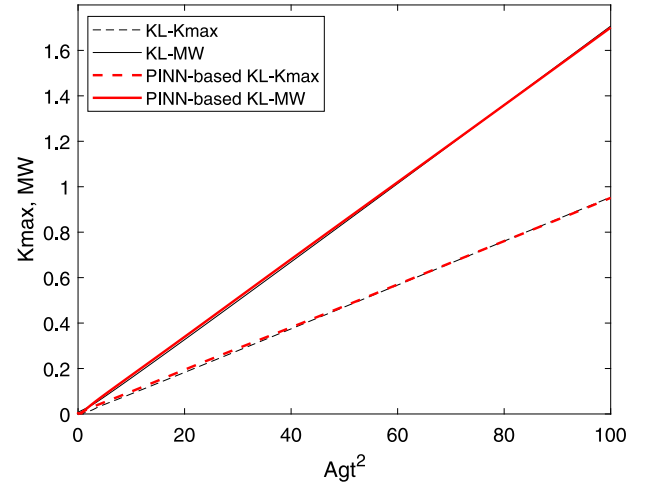


Fig. 3. Using K-L data for PINNs training: temporal evolutions of the integral MW and the K_{max} for RT turbulent mixing with a density ratio of 3:1.

data are required for training, thus solving a forward problem. The inputs of the PINNs model are the temporal coordinate t^* and spatial coordinate x^* , and the outputs are the density $\bar{\rho}^*$, velocity \bar{u}^* , energy \bar{E}^* , and mass fraction \bar{Y}_a^* . For simplicity, the normalized outputs are denoted as ρ , u , E , Y and K , L for the turbulent mixing case.

The density ratio of the test case here is 3:1. The results of the present PINNs are shown in Fig. 2. This indicates that the predicted profiles of ρ , u , E , and Y are frozen as time progresses, which is consistent with the initial profiles. Note that the profiles of the velocity u slightly shift away from zero near the interface (about 10^{-3}), which can be attributed to the limited accuracy of the final loss, and non-zero velocity also exists when using traditional methods.

3.2. Simulation of RT mixing using RANS data for training

Next, we employ PINNs to solve the 1D RANS equations for the RT turbulent mixing problem. The data from the traditional K-L model are used here for the training process, as it can be easily obtained, and the data of the entire field can be used as reference data to examine the predictive ability of PINNs. The used self-similar growth rate parameters (gradient over time) of the integral MW $W \equiv \int Y(1-Y)dx^*$ and

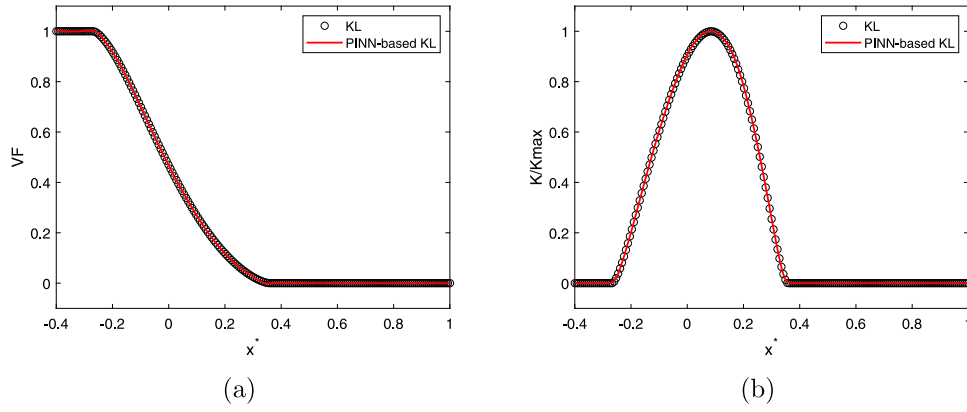


Fig. 4. Using K-L data for PINNs training: volume fraction (a) and turbulent kinetic energy (b) profiles with a density ratio of 3:1.

Table 2

Self-similar growth rate parameters (gradient over time) of the integral MW W and K_{max} for PINNs training. The results of the K-L model are obtained from [45], and the results of ILES are obtained from [24].

Case	W	K_{max}
K-L(3:1)	0.017	0.0095
ILES(3:1)	0.017984	0.010
ILES(20:1)	0.019254	0.0170

K_{max} are reported in Table 2. The data of the profiles of the TKE and VF at the end time $t^* = 1.0$ are also used for PINNs training.

The results of the present PINNs model (denoted as PINN-based KL) are depicted in Figs. 3 and 4, and they are in perfect agreement with the reference K-L data. The predicted spatiotemporal contours of the VF and TKE normalized by the maximum value K_{max} are given in Fig. 5. This indicates that the initially discontinuous profile of the volume fraction is gradually diffused. This is because the mixing of the two materials is getting enhanced as turbulence develops, which is marked by the increasing evolution of the turbulent kinetic energy. This spatiotemporal evolution qualitatively satisfies the physical process. Furthermore, Fig. 6 depicts the prediction of all six outputs at two medium times, i.e., $t^* = 0.35$ and $t^* = 0.7$. These results are quantitatively in good agreement with the calculation of the traditional K-L model, particularly for the predictions of the density ρ and the total energy E , considering there are no training data for these two quantities. It is noticed that the results of the velocity u and turbulent length scale L are not as good as those of the other four outputs. While the former can be attributed to the limited accuracy of the PINNs loss, the latter is because there is neither an equation(physical) information nor data information of the turbulent length scale. L may be better captured if the transport equation of the turbulent length scale is introduced or if extra data information is provided.

This part of work validates the predictive ability of the present PINNs to solve the RT turbulent mixing problem when using limited data for training. This lays the confidence and foundation for our next step-training with limited large eddy simulation data. It is expected that the PINNs model using more accurate LES data for training can be more reasonable and applicable than the PINNs model using K-L data for training or the traditional K-L model.

3.3. Simulation of RT mixing using LES data for training

This section presents the results of the PINNs using the ILES data [24] for training. Two cases, with density ratios of 3:1 and 20:1, are tested. The self-similar growth rate parameters (gradient over time) of the ILES are listed in Table 2.

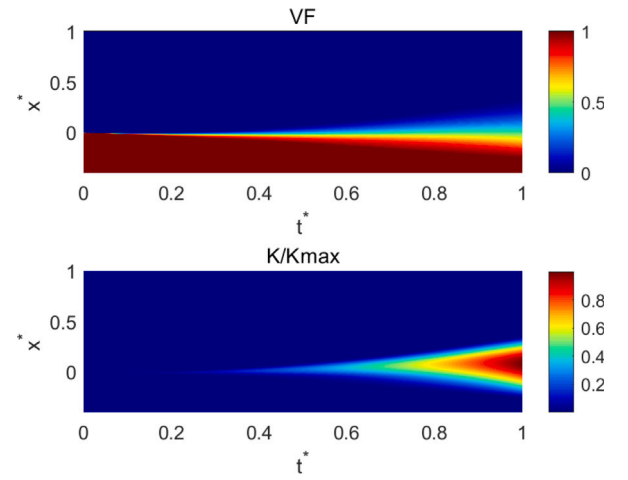


Fig. 5. Using K-L data for PINNs training: predicted contours of the VF and K/K_{max} with a density ratio of 3:1.

The results of the present PINNs are shown in Fig. 7 for the low density ratio case and in Fig. 8 for the high density ratio case. The predictions of the ILES and traditional K-L model are also provided for comparison. This shows that the profiles obtained from the PINNs predictions are in excellent agreement with the ILES data, which are better than those of the traditional K-L model. The predicted contours of the entire spatiotemporal fields are also shown in Fig. 9.

3.4. Discussion about the extrapolation case

Note that the predicted results above are all interpolated based on training data, and we further examine the predicted performance of the present PINNs under the extrapolation situation. Specifically, the integral training data of the test case is within the temporal range of $[0, 0.7]$, and the profile training data is at $t^* = 0.7$. Thus, the results from $t^* = 0.7$ to $t^* = 1.0$ require to be predicted by extrapolation. The final results of the profiles at $t^* = 0.7$ (training) and $t^* = 1.0$ (predicted) are given in Fig. 10, which shows that the extrapolated profiles of PINNs (denoted as PINN-test) at $t^* = 1.0$ do not match the traditional K-L results as good as in the interpolation case. Fig. 11 also depicts the temporal evolutions of MW and K_{max} . Evidently, the prediction deviates from the results of the traditional K-L model after $Ag\tau^2 = 60 (t^* = 0.77)$, further indicating that the predicted performance of the present PINNs under the extrapolation situation is relatively limited.

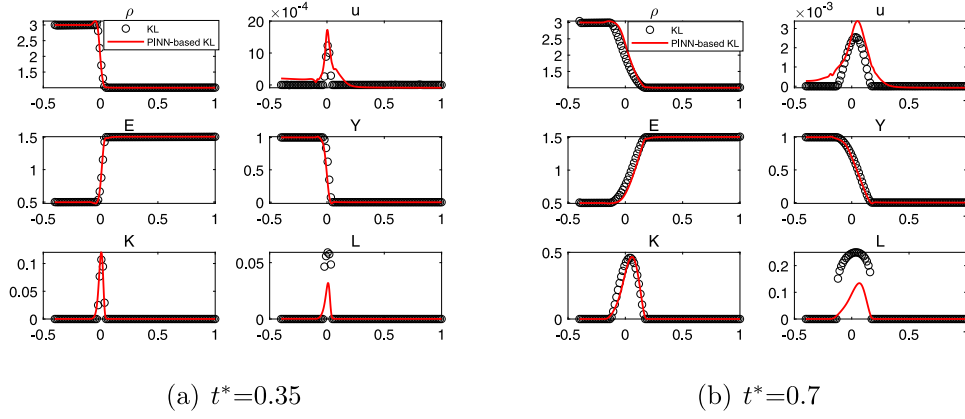


Fig. 6. Using K–L data for PINNs training: predicted six outputs at different times with a density ratio of 3:1.

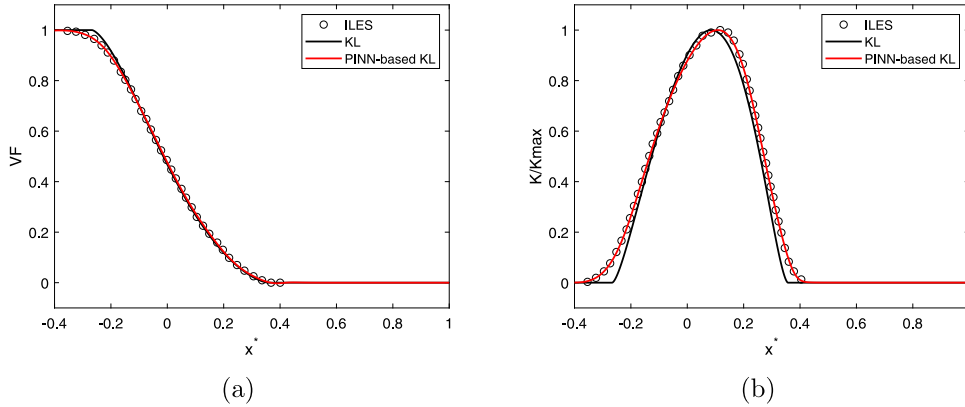


Fig. 7. Using ILES data for PINNs training: VF (a) and K/K_{max} (b) profiles with a density ratio of 3:1.

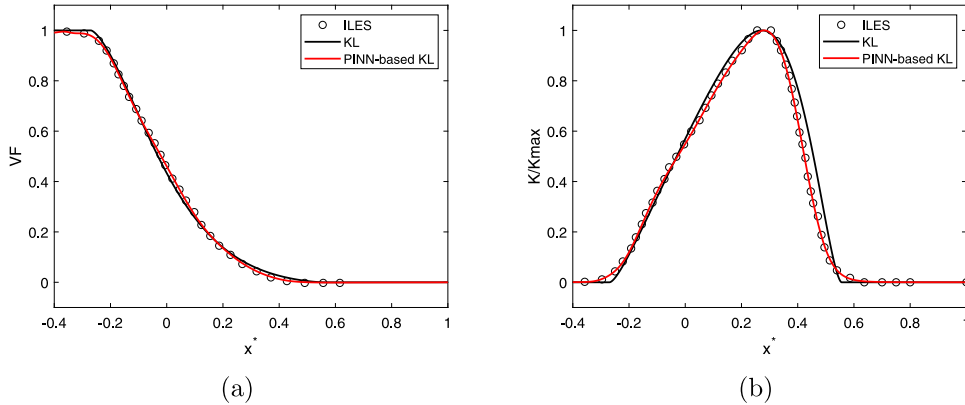


Fig. 8. Using ILES data for PINNs training: VF (a) and K/K_{max} (b) profiles with density ratio 20:1.

4. Conclusions

An improved PINNs framework is proposed for solving the RANS equations for the RT turbulent mixing problem. It is first validated for the steady multi-material problem. Frozen results consistent with the initial conditions have been obtained, satisfying the physical process. Subsequently, the proposed PINNs is applied to solve the RT turbulent mixing problem using training data from the traditional K–L model. The results show that the PINNs yields an excellent description of the training data and provides a reasonably good prediction of the entire spatiotemporal field. Next, we further employ PINNs to solve the RT mixing problem with two density ratios, a low density ratio of 3:1 and high density ratio of 20:1, using the training data from

the ILES. The results of the PINNs simulation are in perfect agreement with the corresponding ILES data, which are better than those of the traditional K–L model. These results demonstrate the applicability of PINNs for solving RT mixing problems under limited training data. This sheds light on further applications of the PINNs approach to complex mixing problems, particularly those with limited measurement data and unknown physical models.

In future studies, two issues may require sustained attention. The first is the generalization problem. It should be noted that the PINNs method yields results in the case of PDEs (with missing physical information constrained by training data for the inverse case) and initial/boundary conditions being determined. Therefore, compared with traditional methods, this method is more like a “solver” than a

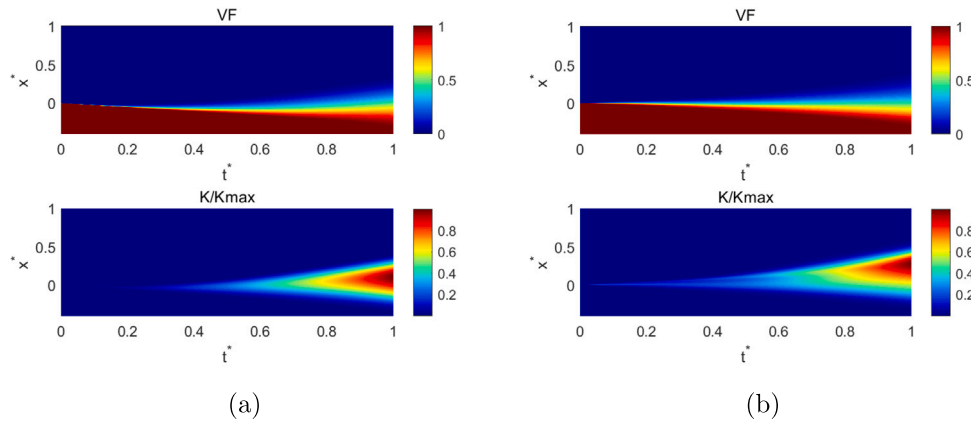


Fig. 9. Using ILES data for PINNs training: predicted contours of the VF and K/K_{max} with the density ratios 3:1 (a) and 20:1 (b).

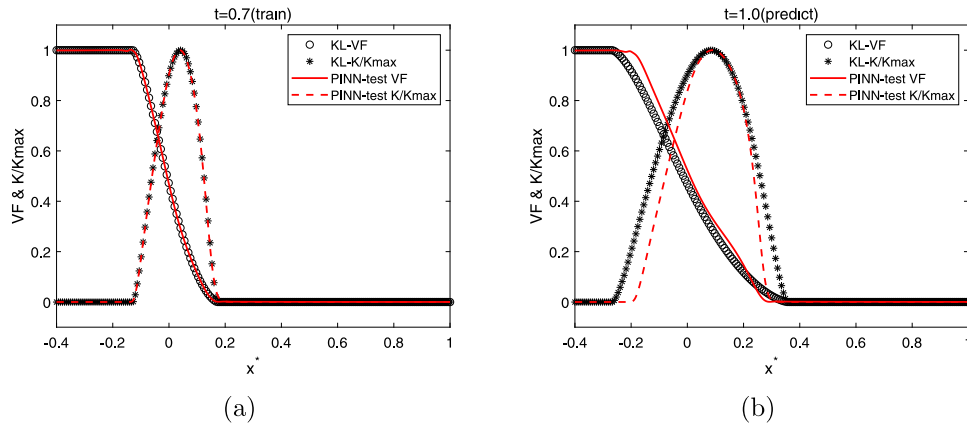


Fig. 10. Using K-L data for PINNs training: training (a) and predicted (b) profiles of the test case under the extrapolation situation with a density ratio of 3:1.

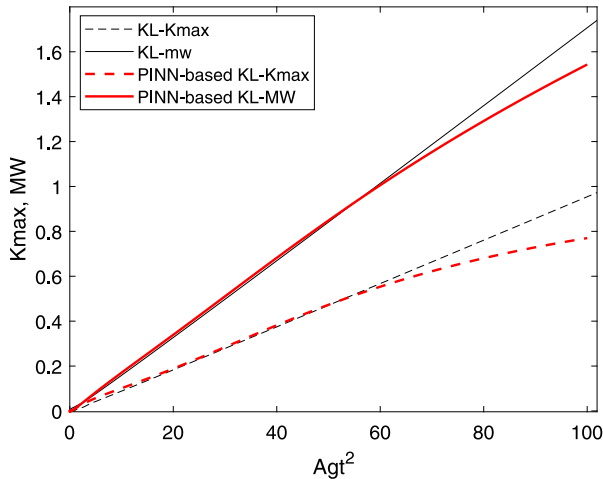


Fig. 11. Using K-L data for PINNs training: the evolutions of MW and K_{max} of the test case under the extrapolation situation with a density ratio of 3:1.

“model”. The latter can be independently generalized to other problems, whereas the former requires recalculation/retraining as the initial or boundary conditions change for different physical problems. For PINNs, the testable generalization problem involves testing their extrapolation performance for the same problem, as discussed above. The second involves solving complex problems using a forward method, that is no training data are included. In particular, when multi-scale variations (turbulence) are involved, it may be necessary to improve the

framework of PINNs training, for example, optimizing the activation function or using multiple sub-networks for training.

CRediT authorship contribution statement

Meng-Juan Xiao: Conceptualization, Methodology, Software, Writing – original draft. **Teng-Chao Yu:** Software, Writing – review & editing. **You-Sheng Zhang:** Investigation, Writing – review & editing. **Heng Yong:** Investigation, Supervision, Writing – review & editing.

Declaration of competing interest

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.

Data availability

Data will be made available on request.

Acknowledgments

We acknowledge the financial support from the National Nature Science Foundation of China (NSFC) under Grant Number 12002059, the NSAF under Grant Number U2230208, and the Key Laboratory of Nuclear Data foundation under Grant Number JCKY2022201C155. We also want to thank Dr. Fan-Sheng Xiong and Dr. Li Liu for helpful discussions.

References

- [1] Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J Comput Phys* 2019;378:686–707.
- [2] Raissi M, Wang Z, Triantafyllou MS, Karniadakis GE. Deep learning of vortex-induced vibrations. *J Fluid Mech* 2019;861:119–37.
- [3] Raissi M, Babaee H, Givi P. Deep learning of turbulent scalar mixing. *Phys Rev Fluids* 2019;4(12):124501.
- [4] Fathi N, Rezaei N. Lymphopenia in COVID-19: Therapeutic opportunities. *Cell Biol Int* 2020;44(9):1792–7.
- [5] Arzani A, Wang J-X, D'Souza RM. Uncovering near-wall blood flow from sparse data with physics-informed neural networks. *Phys Fluids* 2021;33(7):071905.
- [6] Mao Z, Jagtap AD, Karniadakis GE. Physics-informed neural networks for high-speed flows. *Comput Methods Appl Mech Engrg* 2020;360:112789.
- [7] Lu L, Meng X, Mao Z, Karniadakis GE. DeepXDE: A deep learning library for solving differential equations. *SIAM Rev* 2021;63(1):208–28.
- [8] Xie H, Zhai C, Liu L, Yong H. A weighted first-order formulation for solving anisotropic diffusion equations with deep neural networks. 2022, arXiv preprint arXiv:2205.06658.
- [9] Papados A. Solving hydrodynamic shock-tube problems using weighted physics-informed neural networks with domain extension. 2021.
- [10] Liu L, Liu S, Yong H, Xiong F, Yu T. Discontinuity computing with physics-informed neural network. 2022, arXiv preprint arXiv:2206.03864.
- [11] Xiong F, Liu L, Liu S, Wang H, Yong H. Gradient-weighted physics-informed neural networks for one-dimensional Euler equation. 2022.
- [12] Jin X, Cai S, Li H, Karniadakis GE. Nsfnets (Navier-Stokes flow nets): Physics-informed neural networks for the incompressible Navier-Stokes equations. *J Comput Phys* 2021;426:109951.
- [13] Eivazi H, Tahani M, Schlatter P, Vinuesa R. Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations. *Phys Fluids* 2022;34(7):075117.
- [14] Hillebrandt W, Niemeyer JC. Type Ia supernova explosion models. *Annu Rev Astron Astrophys* 2000;38(1):191–230.
- [15] Lindl JD. Inertial confinement fusion: The quest for ignition and energy gain using indirect drive. American Institute of Physics; 1998.
- [16] Zhou Y. Rayleigh–Taylor and Richtmyer–Meshkov instability induced flow, turbulence, and mixing. II. *Phys Rep* 2017;723:1–160.
- [17] Zhou Y. Rayleigh–Taylor and Richtmyer–Meshkov instability induced flow, turbulence, and mixing. II. *Phys Rep* 2017;723–725:1–160.
- [18] Rayleigh L. On the circulation of air observed in Kundt's tubes, and on some allied acoustical problems. *Philos Trans R Soc Lond* 1884;175:1–21.
- [19] Taylor GI. The instability of liquid surfaces when accelerated in a direction perpendicular to their planes. I. *Proc R Soc London. Ser A. Math Phys Sci* 1950;201(1065):192–6.
- [20] Zhou Y, Williams RJ, Ramaprabhu P, Groom M, Thornber B, Hillier A, Mostert W, Rollin B, Balachandrar S, Powell PD, Mahalov A, Attal N. Rayleigh–Taylor and Richtmyer–Meshkov instabilities: A journey through scales. *Physica D* 2021;423:132838.
- [21] Read K. Experimental investigation of turbulent mixing by Rayleigh–Taylor instability. *Physica D* 1984;12:45–58.
- [22] Cabot WH, Cook AW. Reynolds number effects on Rayleigh–Taylor instability with possible implications for type Ia supernovae. *Nat Phys* 2006;2(8):562.
- [23] Youngs DL. The density ratio dependence of self-similar Rayleigh–Taylor mixing. *Phil Trans R Soc A* 2013;371(2003):20120173.
- [24] Kokkinakis I, Drikakis D, Youngs D, Williams R. Two-equation and multi-fluid turbulence models for Rayleigh–Taylor mixing. *Int J Heat Fluid Flow* 2015;56:233–50.
- [25] Pal N, Kurien S, Clark T, Aslangil D, Livescu D. Two-point spectral model for variable-density homogeneous turbulence. *Phys Rev Fluids* 2018;3:124608.
- [26] Liang H, Hu X, Huang X, Xu J. Direct numerical simulations of multi-mode immiscible Rayleigh–Taylor instability with high Reynolds numbers. *Phys Fluids* 2019;31(11).
- [27] Pal N, Boureima I, Braun N, Kurien S, Ramaprabhu P, Lawrie A. Local wave-number model for inhomogeneous two-fluid mixing. *Phys Rev E* 2021;104:025105.
- [28] Su X, Walters R, Aslangil D, Yu R. Forecasting variable-density 3D turbulent flow. In: ICLR 2021 SimDL workshop. 2021, URL <https://simdl.github.io/files/44.pdf>.
- [29] Sengupta A, Sundaram P, Suman VK, Sengupta TK. Three-dimensional direct numerical simulation of Rayleigh–Taylor instability triggered by acoustic excitation. *Phys Fluids* 2022;34(5).
- [30] Boureima I, Gyrya V, Saenz J, Kurien S, Francois M. Dynamic calibration of differential equations using machine learning, with application to turbulence models. *J Comput Phys* 2022;457:110924.
- [31] Roberts MS, Jacobs JW. The effects of forced small-wavelength, finite-bandwidth initial perturbations and miscibility on the turbulent Rayleigh–Taylor instability. *J Fluid Mech* 2016;787:50–83.
- [32] Zhou Y, Cabot WH. Time-dependent study of anisotropy in Rayleigh–Taylor instability induced turbulent flows with a variety of density ratios. *Phys Fluids* 2019;31(8).
- [33] Zhou Y, Clark TT, Clark DS, Gail Glendinning S, Aaron Skinner M, Huntington CM, Hurricane OA, Dimits AM, Remington BA. Turbulent mixing and transition criteria of flows induced by hydrodynamic instabilities. *Phys Plasmas* 2019;26(8). 080901.
- [34] Aslangil D, Farley Z, Lawrie AGW, Banerjee A. Rayleigh–Taylor Instability With Varying Periods of Zero Acceleration. *J Fluids Eng* 2020;142(12).
- [35] Aslangil D, Lawrie AGW, Banerjee A. Effects of variable deceleration periods on Rayleigh–Taylor instability with acceleration reversals. *Phys Rev E* 2022;105:065103.
- [36] Sundaram P, Sengupta A, Sengupta TK. A non-overlapping high accuracy parallel subdomain closure for compact scheme: Onset of Rayleigh–Taylor instability by ultrasonic waves. *J Comput Phys* 2022;470:111593.
- [37] Sundaram P, Sengupta A, Suman VK, Sengupta TK. Non-overlapping high-accuracy parallel closure for compact schemes: Application in multiphysics and complex geometry. *ACM Trans Parallel Comput* 2023;10(1).
- [38] Sengupta A, Verma AK. Role of unstable thermal stratifications on the Rayleigh–Taylor instability. *Comput & Fluids* 2023;252:105773.
- [39] Sengupta A, Joshi B. Effects of stabilizing and destabilizing thermal gradients on reversed shear-stratified flows: Combined Kelvin–Helmholtz Rayleigh–Taylor instability. *Phys Fluids* 2023;35(1).
- [40] Dimonte G, Tipton R. K-L turbulence model for the self-similar growth of the Rayleigh–Taylor and Richtmyer–Meshkov instabilities. *Phys Fluids* 2006;18(8):085101.
- [41] Chiravalle VP. The kl turbulence model for describing buoyancy-driven fluid instabilities. *Laser Part Beams* 2006;24(3):381–94.
- [42] Morgan B, Greenough J. Large-eddy and unsteady RANS simulations of a shock-accelerated heavy gas cylinder. *Shock Waves* 2016;26(4):355–83.
- [43] Kokkinakis IW, Drikakis D, Youngs DL. Modeling of Rayleigh–Taylor mixing using single-fluid models. *Phys Rev E* 2019;99(1):013104.
- [44] Xiao M, Zhang Y, Tian B. Unified prediction of reshocked Richtmyer–Meshkov mixing with KL model. *Phys Fluids* 2020;32(3):032107.
- [45] Xiao M-J, Zhang Y-S, Tian B-L. Modeling of turbulent mixing with an improved K–L model. *Phys Fluids* 2020;32(9):092104.
- [46] Xiao M, Zhang Y, Tian B. AK–L model with improved realizability for turbulent mixing. *Phys Fluids* 2021;33(2):022104.
- [47] Zhang YS, Xie HS, Xiao MJ, He ZW, Tian BL. Methodology for determining the coefficients of turbulent mixing model. *J Fluid Mech* 2020;905(A26):1–40.
- [48] Livescu D. Numerical simulations of two-fluid turbulent mixing at large density ratios and applications to the Rayleigh–Taylor instability. *Phil Trans R Soc A* 2013;371(2003):20120185.
- [49] Baydin AG, Pearlmutter BA, Radul AA, Siskind JM. Automatic differentiation in machine learning: A survey. *J March Learn Res* 2018;18:1–43.
- [50] Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014, arXiv preprint arXiv:1412.6980.
- [51] Zhu C, Byrd R, Lu P, Nocedal J. A limited memory algorithm for bound constrained optimisation. *SIAM J Sci Stat Comput* 1995;16(5):1190–208.