



WaveNets: physics-informed neural networks for full-field recovery of rotational flow beneath large-amplitude periodic water waves

Lin Chen^{1,4} · Ben Li¹ · Chenyi Luo² · Xiaoming Lei³

Received: 10 August 2023 / Accepted: 5 January 2024 / Published online: 23 February 2024
© The Author(s) 2024

Abstract

We formulate physics-informed neural networks (PINNs) for full-field reconstruction of rotational flow beneath nonlinear periodic water waves using a small amount of measurement data, coined WaveNets. The WaveNets have two NNs to, respectively, predict the water surface, and velocity/pressure fields. The Euler equation and other prior knowledge of the wave problem are included in WaveNets loss function. We also propose a novel method to dynamically update the sampling points in residual evaluation as the free surface is gradually formed during model training. High-fidelity data sets are obtained using the numerical continuation method which is able to solve nonlinear waves close to the largest height. Model training and validation results in cases of both one-layer and two-layer rotational flows show that WaveNets can reconstruct wave surface and flow field with few data either on the surface or in the flow. Accuracy in vorticity estimate can be improved by adding a redundant physical constraint according to the prior information on the vorticity distribution.

Keywords PINN · Nonlinear water waves · Free surface flow · Rotational flow · Flow field recovery

1 Introduction

Surface water waves dynamics is affected by the underlying current [1, 2]. The wave–current interaction is a well-known topic of interest in different disciplines, e.g., applied mathematics [3, 4] and ocean engineering [5–7]. The flow velocity under the influence of such an interaction is often described by the Euler equation. However, because it is a

free surface problem and of strong nonlinearity in the case of large-amplitude waves, the interaction effect is not yet fully understood. Relevant studies mostly focused on periodic water waves propagating on irrotational (zero vorticity) or rotational flows. For rotational flows with constant vorticity, the steady problem can be formulated using a conformal mapping and then solved numerically, revealing rich characteristics of the stokes waves under the influence of vorticity [8, 9]. For periodic water waves on rotational flow with an arbitrary vorticity distribution, the free surface problem is transformed to a boundary-value problem using the semi-hodograph transform and then solved by numerical continuation to address the strong nonlinearity [10, 11], which nevertheless is unable to deal with flows with stagnation points. Asymptotic and numerical methods have been explored for both steady flows [12–14] and unsteady flows [15]. There exist many field measurements and experiments in water flume which confirmed the wave–current interactions and also validated the presented models [2, 16].

Beside the forward problem solving velocity and pressure fields for given wave period/length, water depth and wave height, the inverse problem related to water waves and wave current interactions is also of significant importance. A typical example is the reconstruction of wave height or wave profile using pressure measurements on the seabed

✉ Chenyi Luo
cheluo@ethz.ch

Lin Chen
linchen@tongji.edu.cn

Ben Li
2232241@tongji.edu.cn

Xiaoming Lei
xiaoming.lei@polyu.edu.hk

¹ Department of Bridge Engineering, Tongji University, 1239 Siping Road, Shanghai 200092, China

² Department of Mechanical and Process Engineering, ETH Zurich, Tannenstrasse 3, Zurich 8092, Switzerland

³ Department of Civil and Environmental Engineering, Hong Kong Polytechnic University, Kowloon, Hong Kong 999077, China

⁴ Shanghai Qi Zhi Institute, Shanghai 200232, China

or at intermediate positions. For examples, Oliveras et al. [17] derived the relation between the pressure at the sea bed and the surface elevation from the Euler equation for irrotational flow and also presented asymptotic approximations. Constantin [18] provided bounds on the wave heights on irrotational flow from sea bed pressure measurements which is applicable to large-amplitude waves. Basu [19] further derived bounds for such waves according to pressure and velocity measurements at an arbitrary intermediate location. It is noted that in most cases, only wave elevation is recovered. The recovery of other flow characteristics has not gained too much attention. Notably, in field measurements and water flume tests, multiple types of data can be measured. In field measurements, pressure transducers/sensors are widely used for pressure measurements [20, 21]. In water flume tests, wave gauges are used to measure water surface and particle image velocimetry can be used to record the movement of seeding particles for velocity measurements [22]. Acoustic doppler velocimetry can also be used for velocity measurements at specific points [23]. The development of remote sensing and aerial imaging technology allows field measurement of the free surface of waves [24, 25].

Solving the inverse coupled computational fluid dynamics (CFD) problem is often computationally challenging and typically involves solving an ill-posed problem. The rapid development of machine learning technology provides alternative methods for investigating forward and inverse problems in various fields [26–28]. In ocean engineering, neural networks (NNs) were applied for wave height and period predictions [29] and also for the construction of wave equation from hydraulic data [30] decades ago. In terms of flow field reconstruction, Convolutional Neural Networks (CNNs) were used to reconstruct three-dimensional turbulent flow beneath the free surface using surface measurement data [31]. In these studies, purely data-driven methods were used and the machine learning models face challenges of poor generalization and interpretability, which are common issues in machine learning [32] and linear water waves were concerned.

In recent years, Physics-Informed Neural Networks (PINN) have been proposed and applied in various fields, including fluid mechanics [33, 34]. Generally, PINNs use a typical deep neural network (DNN) but with relevant physics embedded, e.g., in the loss function, leading to a model driven by physics and data whose respective contribution is adjustable, in contrast to purely data-driven models. In other words, by integrating the physics information/knowledge, PINNs achieve better generalization and interpretability. Typical applications of PINNs for fluid mechanics include solving Navier–Stokes equations [35], Reynolds-Averaged Navier–Stokes equations [36], fluid–structure

interaction problems [37], Euler equation governing high-speed air flow [38], wave propagation and scattering [39], to name but a few, see [40] for a review. The success of PINNs for fluid mechanics is partly attributed to the fact that the associated governing equations are well established but the field data or experimentally measurement data is relatively limited as compared to the whole domain of concern. Generally speaking, PINNs are found probably not as efficient as compared to the traditional method but it is more effective in dealing with the inverse problem. The reasons why PINNs may fail have been discussed in depth in Refs. [41, 42] and several approaches have been presented to improve the performance, e.g., a balance of the gradients of the boundary loss and the residual loss [41], clustering training points at positions where strong non-linearity is present [38], learning rate refined according to residual (residual based adaptive refinement method) [43], dynamic weights [35], and adaptive activation function [44], among others [45].

PINNs have been applied to water wave problems. For rogue periodic wave and periodic wave solutions for the Chen–Lee–Liu equation [46], the accuracy of PINN was shown in the construction of the wave profile using a few data. PINN was applied to solve inverse water waves problem using the Serre–Green–Naghgi equation for estimating the wave surface profile and depth-averaged horizontal velocity [47]. Huang et al. [48] proposed a novel PINN model for free-surface water by including the boundary and initial conditions in the governing equation and the model was tested on linear and small-amplitude water waves. In the present study, a PINN model is proposed to full-field reconstruction of rotational flows beneath large-amplitude steady periodic water waves using only a small amount of measurement data. To the best knowledge of the authors, this has not been considered in the literature. For addressing the free water surface, a separate DNN is used for modeling the free boundary [49, 50] and an algorithm is developed to update the residual points according to the surface profile. Furthermore, the PINN model is formulated to take into account multiple type of data, including pressure, velocity and surface elevation. As waves of small to extremely large heights are concerned, the numerical continuation method is used for data preparation [11]. The method is demonstrated for waves of small to extreme height considering both the cases of rotational flow with constant vorticity and nonconstant vorticity.

The rest of this paper is organized as follows. Section 2 describes the problem under consideration. Section 3 presents the PINN model, numerical implementation, and generation of the data set. Section 4 applies the developed model and discusses the results for typical cases. A conclusion of the study is provided in Sect. 5.

2 Problem statement

2.1 Full-field recovery of flow field

We are interested in two-dimensional (2D) rotational flows under steady periodic water waves. As illustrated in Fig. 1, the Cartesian coordinate system (X, Y) is used to describe the problem, with X the horizontal axis and Y = vertical axis pointing upwards. The X axis is placed on the mean water level, corresponding to $Y = 0$ and flat bed is assumed, corresponding to $Y = -d$ where d denotes the water depth. A periodic wave of wavelength $2L$ is propagating in X direction with wave speed c . The profile of the underlying current can be arbitrary. Inviscid fluid is assumed. Let η denote the wave elevation, (u, v) be the velocity field, and P denote the pressure field. As periodic waves are considered, the time-variable t can be eliminated by the change of coordinates $x = X - ct$ and $y = Y$. In the following, the problem is dealt with in the moving frame (x, y) , i.e., $\eta = \eta(x)$, $u = u(x, y)$, $v = v(x, y)$ and $P = P(x, y)$.

As already mentioned, pressure transducers/sensors are often used to measure pressure in the fluid, mostly at the sea bed in field measurements. The wave surface profile could be measured using wave gauges and velocities at specific points/region can be monitored using particle image velocimetry. Remote sensing and aerial imaging techniques allow us to measure velocities at the free surface. Because different types of data are measured using varied types of sensors, they are thus commonly measured at different locations. Those data can then be used to train a NN to recover the velocity field, pressure field in the whole domain along with the determined top surface. On the other hand, in practice only limited amount of data is available. Hence, well-established physics knowledge about water waves needs to be integrated in the NNs.

2.2 Governing equation

2.2.1 Euler equation

For integrating physics into the PINNs and also generating data sets, the governing equation is briefly introduced herein. Under the incompressibility assumption of the fluid, the equation of mass conservation in the moving frame reads

$$u_x + v_y = 0, \quad (1)$$

where variables with a subscript denote the partial derivatives. The motion of the inviscid fluid under gravity is governed by Euler equation that

$$\begin{cases} (u - c)u_x + vu_y = -P_x, \\ (u - c)v_x + vv_y = -P_y - g, \end{cases} \quad (2)$$

holds in the domain $D_\eta = \{(x, y) \in \mathbb{R}^2 : -d \leq y \leq \eta(x)\}$. In the preceding equations, g is the gravitational constant, the mass density of water is assumed $\rho = 1$ and hence the actual pressure needs to be scaled accordingly. On the free surface of the fluid domain at the top, the kinematic and dynamic conditions read

$$v = (u - c)\eta_x, P = P_{\text{atm}} \text{ on } y = \eta(x), \quad (3)$$

where P_{atm} is atmospheric pressure. Impenetrable flat bottom is considered and the corresponding boundary condition is given as

$$v = 0 \text{ on } y = -d. \quad (4)$$

The vorticity of the rotational flow is defined as

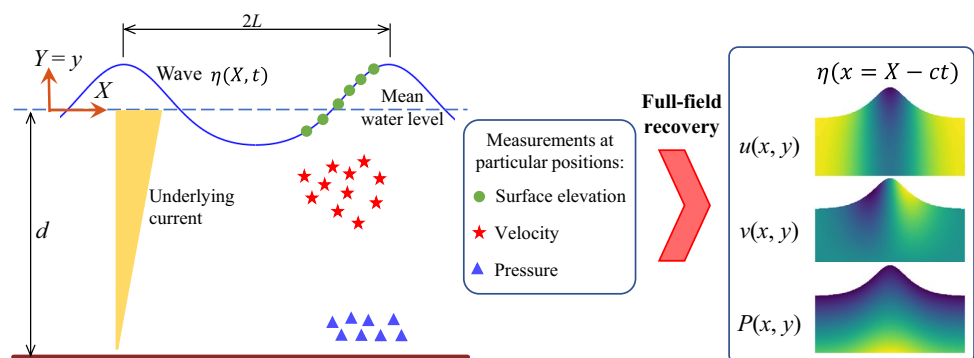
$$\Omega = u_y - v_x. \quad (5)$$

Note it is often required that $u < c$ [51] to allow the computation using the semi-hodograph transform.

2.2.2 The stream function formulation

Define a stream function $\psi(x, y)$ which fulfills

Fig. 1 Problem illustration



$$\psi_y = u - c \quad \text{and} \quad \psi_x = -v. \quad (6)$$

Furthermore, let $\psi = 0$ on $y = \eta(x)$ and define $\psi = -p_0$ on $y = -d$, we have the following relation:

$$p_0 = \int_{-d}^{\eta(x)} \psi_y dy = \int_{-d}^{\eta(x)} [u(x, y) - c] dy, \quad (7)$$

and p_0 is known as the relative mass flux. In the domain, ψ can be expressed as

$$\psi(x, y) = -p_0 + \int_{-d}^y [u(x, s) - c] ds, \quad (8)$$

with s = an integration variable. As aforementioned, $u < c$ is assumed throughout the domain and thus $p_0 < 0$ [3, 51]. According to Eq. (5), the vorticity is a function of the stream function as

$$\Delta\psi := u_y - v_x = \gamma(\psi). \quad (9)$$

Here, $\gamma(\cdot)$ denotes the vorticity as a function of the streamline ψ [3]. According to the Bernoulli's law, throughout the fluid domain, the hydraulic head

$$E = \frac{(u - c)^2 + v^2}{2} + gy + P - \int_0^\psi [\gamma(s)] ds, \quad (10)$$

remains a constant. To summarize, the governing equation can be expressed in terms of the stream function as

$$\Delta\psi = \gamma(\psi) \quad \text{in} \quad -d < y < \eta(x), \quad (11)$$

$$|\nabla\psi|^2 + 2g(y + d) = Q \quad \text{on} \quad y = \eta(x), \quad (12)$$

$$\psi = 0 \quad \text{on} \quad y = \eta(x), \quad (13)$$

$$\psi = -p_0 \quad \text{on} \quad y = -d, \quad (14)$$

with the Bernoulli constant $Q = 2(E - P_{\text{atm}} + gd)$.

2.2.3 The modified height function formulation

For numerical computation, a fixed computation domain is favored to simplify the problem. For this purpose, Henry [52] proposed a semi-hodograph transform, as expressed as

$$q = x \quad \text{and} \quad p = \psi(x, y)/p_0, \quad (15)$$

where (p, q) are the coordinate axes in the transformed domain and $q \in [-L, L]$ and $p \in [-1, 0]$. The unknown then becomes the modified height function defined as

$$h(q, p) = \frac{y}{d} - p. \quad (16)$$

As the mean water level is assumed at $y = 0$ and $\psi = -p_0$ at $y = -d$, we have

$$\int_{-L}^L h(q, 0) dq = 0 \quad \text{and} \quad h(q, -1) = 0. \quad (17)$$

The vorticity is now a function of p , i.e., $\gamma(\psi) = \gamma(p)$. The governing equations become [52]

$$\begin{aligned} & \left(\frac{1}{d^2} + h_q^2 \right) h_{pp} - 2h_q h_{qp} (h_p + 1) + h_{qq} (h_p + 1)^2 \\ & + \frac{\gamma(p)}{p_0} (h_p + 1)^3 = 0 \quad \text{in} \quad -1 < p < 0, \end{aligned} \quad (18)$$

$$\frac{1}{d^2} + h_q^2 + \frac{(h_p + 1)^2}{p_0^2} [2gd(h + 1) - Q] = 0 \quad \text{on} \quad p = 0, \quad (19)$$

$$h = 0 \quad \text{on} \quad p = -1. \quad (20)$$

By solving the above boundary-value problem for given parameter Q, d, p_0 , the height function h is obtained and then $y = d[h(q, p) + p]$. The wave height can then be recovered. Note that waves of one crest and one trough per period are concerned. According to the semi-hodograph transform, the velocity field is computed afterwards as [53]

$$c - u = -\frac{p_0}{d(h_p + 1)}, \quad v = \frac{p_0 h_q}{h_p + 1}. \quad (21)$$

Correspondingly, the pressure field is obtained by

$$\begin{aligned} P = & P_{\text{atm}} + \tilde{F}(p) - gd(h + 1) - gdp \\ & - \frac{p_0^2(1/d^2 + h_q^2)}{2(h_p + 1)^2} + \frac{Q}{2}, \end{aligned} \quad (22)$$

where $\tilde{F}(p) = \int_0^p p_0 \gamma(s) ds$. The wave speed is determined [3] by

$$c = k - \frac{1}{L} \int_0^L [u(x, -d) - c] dx, \quad (23)$$

where k is the average horizontal current strength on the bed and in this study we always set $k = 0$.

In the following, Eqs. (18–20) are solved using the finite difference method together with the numerical continuation [11] for generating data sets. The governing equations, i.e., Eqs. (1–5), can be effectively integrated into the PINN model, which will be elaborated upon in detail in the next section.

3 PINN model for large-amplitude periodic waves on rotational flow

3.1 PINN architecture

There are various types of neural networks, including fully connected neural networks (FCNNs), convolutional neural networks (CNNs), recurrent neural networks (RNNs), and long short-term memory networks (LSTMs). They have been applied in developing PINNs for different purposes. The fully connected framework is easily applicable to measurement data scattered in disparate temporal and spatial locations, while other frameworks require continuous sequential data. Therefore, existing studies mostly incorporate prior physical knowledge into deep learning models using FCNNs. We also adopt the FCNN herein, which can be simply represented as a nonlinear mapping

$$NN_{\theta} : R^{\text{in}} \rightarrow R^{\text{out}},$$

where R^{in} is the input and R^{out} is the output of the neural network. We follow the original PINN framework [33] to use the DNN. Furthermore, to take into measurement data, we construct the velocity and pressure fields in (x, y) coordinate system instead of (p, q) coordinate system after transform. Note that the wave elevation η is a function of only x , while the velocity and pressure fields are functions of both x and y , we thus utilize two FCNNs, namely NN_1 and NN_2 . NN_1 predicts the wave elevation η for specified x coordinate. NN_2 predicts the velocity field (u, v) and the pressure P for given point coordinates (x, y) . They can be simply expressed as

$$\hat{\eta} = NN_1(x; \mathbf{w}_1, \mathbf{b}_1), \quad (24)$$

$$(\hat{u}, \hat{v}, \hat{P}) = NN_2(x, y; \mathbf{w}_2, \mathbf{b}_2), \quad (25)$$

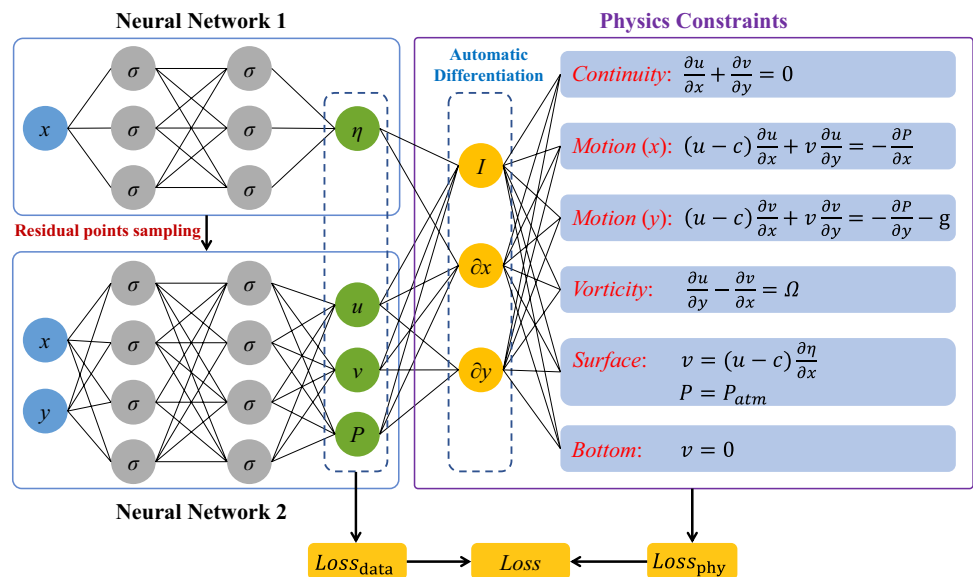
where \mathbf{w}_1 and \mathbf{b}_1 , respectively, denote the weights and biases in NN_1 , and \mathbf{w}_2 and \mathbf{b}_2 , respectively, denote the weights and biases in NN_2 . The two networks are connected through the loss function and in the residual point selection, as will be explained in the subsequent sections. Because in PINN models differentiation operations are often required for imposing physical constraints, the activation function must be differentiable. However, the derivative of the activation function *relu* is discontinuous, making it unsuitable for PINNs. The *tanh* function, with its center point at 0, is often more effective than the *sigmoid* function (with a center point at 0.5) [26]. Therefore, we choose *tanh* function as the activation function.

Figure 2 shows the structure of the NNs, coined WaveNets, where the physical constraints are also illustrated. More constraints or redundant constraints, such as the definition of vorticity, Bernoulli constant and mass flux, can be added using a similar manner, which will be discussed, respectively, in the subsequent section for particular cases.

3.2 Loss function

Following the methods used in many other studies, both the data loss term and the physics loss term in the loss function are represented in the form of mean square error (MSE). The loss term is denoted by \mathcal{L} , and $\mathcal{L}_{\text{data}}$ represents the part related to measurement data, defined as

Fig. 2 WaveNets structure



$$\mathcal{L}_{\text{data}} = \sum_j \frac{1}{N_j} \sum_{i=1}^{N_j} \{[(\hat{u}^i - u^i)^2 + (\hat{v}^i - v^i)^2] + (\hat{P}^i - P^i)^2 + (\hat{\eta}^i - \eta^i)^2\}, \quad (26)$$

where i is the measurement point index in the j th interior region or bound of the fluid domain, and N_j represents the total number of sampling points in each region/bound. The known or measured velocity, pressure and surface profile are denoted by (u^i, v^i) , P^i and η^i at the i th sampling point. Correspondingly, \hat{u}^i , \hat{v}^i , \hat{P}^i and $\hat{\eta}^i$ represent the predicted values by WaveNets at the i th sampling point. It needs to emphasize that for a specific sampling point there probably only one type of measurements available and correspondingly only the term in Eq. (26) related to the measurement is evaluated while others are omitted. For example, the data loss term corresponding to η can only be computed through measurements obtained on the top surface of the flow.

For the definition of physical constraints, we recall the governing equations and flow characteristics described in Sect. 2.2. They can be naturally embedded as regularization terms into WaveNets. First, within the domain, the fluid flow needs to satisfy the continuity condition and momentum conservation, leading to three constraints. The corresponding residuals are denoted by e_1 , e_2 , and e_3 . As WaveNets are built to use (x, y) as input, the three constraints are directly evaluated in the x, y coordinate system, from Eqs. (1, 2) as

$$\begin{cases} e_1 = \hat{u}_x + \hat{v}_y, \\ e_2 = (\hat{u} - \hat{c})\hat{u}_x + \hat{v}\hat{u}_y + \hat{P}_x, \\ e_3 = (\hat{u} - \hat{c})\hat{v}_x + \hat{v}\hat{v}_y + \hat{P}_y + g, \end{cases} \quad (27)$$

Meanwhile, at the upper boundary, i.e., the free surface, the following two constraints are defined

$$\begin{cases} e_4 = \hat{v} - (\hat{u} - \hat{c})\hat{\eta}_x, \\ e_5 = \hat{P} - P_{\text{atm}}, \end{cases} \quad (28)$$

For the bottom boundary, i.e., the flat bed, the following constraint is enforced

$$e_6 = \hat{v}. \quad (29)$$

Recalling that we are concerned with steady periodic waves, under the trough $\hat{v} = 0$ can be enforced, which can be included in the calculation of e_6 . Therefore, the loss related to the physical constraints is generally represented as

$$\mathcal{L}_{\text{phy}} = \frac{1}{N_{\text{in}}} \sum_{i=1}^{N_{\text{in}}} (e_1^2 + e_2^2 + e_3^2) + \frac{1}{N_{\text{up}}} \sum_{i=1}^{N_{\text{up}}} (e_4^2 + e_5^2) + \frac{1}{N_{\text{low}}} \sum_{i=1}^{N_{\text{low}}} e_6^2, \quad (30)$$

where N_{in} , N_{up} , and N_{low} , respectively, denote the number of points in the domain, and on the surface and bottom boundary for physical constraints, and i is the index of the point. In addition, because the waves considered in the case are periodic, the values of u , v , and P on the left boundary of the flow field should be the same as the corresponding points on the right boundary, which can also serve as a physical constraint theoretically. The total loss in WaveNets, as shown in Fig. 2, can be represented as

$$\mathcal{L} = \mathcal{L}_{\text{data}} + \lambda \mathcal{L}_{\text{phy}}, \quad (31)$$

where the hyper-parameter λ is used to balance the data loss term and the physical loss term. It has been found of significant impact on the predictive ability of PINNs. When $\lambda = 0$, the loss function only depends on the data, and PINNs degenerate to fully data-driven models; when λ takes a large value, the predicted values of known points may have a significant discrepancy from the actual values. In training the model, the parameter can be adjusted to find the optimal value that allows the model to achieve the best prediction performance [54], but this requires a significant computational cost. To address the issue of imbalanced back-propagation gradients during the training process, Wang et al. [41] proposed a learning rate annealing algorithm, which utilizes gradient statistics to balance the contribution between the data loss and the physical loss. For the present problem, we also implemented this method, but no significant improvement on the prediction performance was found while the computation time was increased considerably. Therefore, in the following examples, unless otherwise specified, we set λ to 1.

3.3 Selection of residual points beneath free surface

In WaveNets, residual points refer to the points located beneath the water surface and subject to the constraints as described in Eq. (27). Existing studies mostly used PINNs for solving partial differential equations on a fixed domain. In that case, the points for evaluating residuals are normally randomly selected within the domain. Wang et al. [49] applied PINNs to solve the Stefan problem of heat transfer where the interface on which the temperature phase changes is unknown, similar to a free boundary problem. Nevertheless, residual points were evaluated still on randomly chosen points in the whole domain, i.e., on both sides of the interface. For the water wave problems, only the domain beneath the free surface is filled with water flow which is subjected to the physical constraints. Imposing the constraints on points outside the upper boundary is lacking physical meaning and could be misleading in model training.

Therefore, we hereby propose a novel scheme to address this issue in PINN models for free-surface problems. We determine the free boundary based on the predictions from trained NN_1 , and then take into account only residuals of those points under the free surface. Figure 3 demonstrates the scheme. At the beginning of the training, when NN_1 lacks predictive capability, we have two options for distributing the residual points. One option is to select the residual points based on information from known measurement points to ensure they are located below the free surface. Alternatively, we can directly distribute the residual points within the fixed domain $[x_{\min}^*, x_{\max}^*] \times [0, y_{\max}^*]$ determined by the minimum and maximum coordinates of the measurement points (denoted with superscript *). In subsequent training steps, the trained NN_1 is used to compute the corresponding wave elevation and thus random points can be selected accordingly. Specifically, a residual point is chosen with its y -coordinate within 0 and the corresponding wave elevation. Repeating the sampling process N_{res} times yields the full set of residual points. For computational efficiency, the process described above is used to update the residual points after a certain number of iterations in modal training. As will be shown in the next section, by using this algorithm, during the model training the residual points gradually distribute below the free surface.

3.4 Data generation and preprocessing

In reality, water waves can have a length of over 100 m, leading to a very large computation domain. To reduce the computational domain and also normalize the variables, the following scaling scheme is applied [3]:

$$\begin{aligned} \eta &= \kappa \bar{\eta}, & c &= \bar{c}, & u &= \bar{u}, & v &= \bar{v}, & P &= \bar{P}, & x &= \kappa \bar{x}, \\ y &= \kappa \bar{y}, & t &= \kappa \bar{t}, & g &= \kappa^{-1} \bar{g}, & \gamma &= \kappa^{-1} \bar{\gamma}, \end{aligned} \quad (32)$$

where the variables with a bar indicate that they correspond to the physical system before scaling and κ is the wavenumber of the system. The normalization leads to a 2π -system governed by Eqs. (18–20), but the parameters \bar{d} , \bar{g} and $\bar{\gamma}$ need to be replaced by $\kappa \bar{d}$, \bar{g}/κ and $\bar{\gamma}/\kappa$, respectively. When using

the numerical continuation for generating the data sets, the computational domain is fixed as $[-\pi, \pi] \times [-1, 0]$.

Using the above expressions, the input x of WaveNets is already normalized, i.e., $x \in [-\pi, \pi]$. For the input y and the output variables, it is rather difficult to implement the widely used min–max normalization or z -score normalization, because only few measurements are available for WaveNets. Therefore, the variables are not further normalized after the scaling using the wavenumber, as in Eq. (32). Note that in water flume experiments or field measurements the wavelength is normally easy to obtain for the above scaling.

For convenience, in model training and validation, we put x -axis on the flat bed instead of on the mean water level as in the numerical analysis. Also, the horizontal velocity after translation, i.e., $c - u$ instead of u , is considered as the output. Note the wave speed can also be recovered using Eq. (23) from the estimated $c - u$.

3.5 Implementation

The physical constraints are implemented based on the automatic differentiation functionality of TensorFlow [55]. For model training, we first use the Adam optimizer which is one of the most commonly used optimizers in deep learning, featured by fast convergence in finding the weight matrix ($\mathbf{w}_1, \mathbf{w}_2$) and bias coefficients ($\mathbf{b}_1, \mathbf{b}_2$) of WaveNets. Following the recommended parameter settings in Ref. [56], we set the initial learning rate l_{r0} of Adam optimizer to $l_{r0} = 0.001$, the momentum parameter β_1 to 0.9, and the second-order momentum parameter β_2 to 0.999. Furthermore, to improve the stability of model training, we adopt the exponential decay method to gradually reduce the learning rate (l_r) during the training process, that is

$$l_r = l_{r0} \xi^{\frac{\text{global step}}{\text{decay step}}}, \quad (33)$$

where ξ is the decay coefficient and $\xi = 0.9$ is adopted herein. The decay step is set to 1000. This helps to slow down the convergence speed when the model approaches the optimal solution and avoids issues such as oscillation and over fitting.

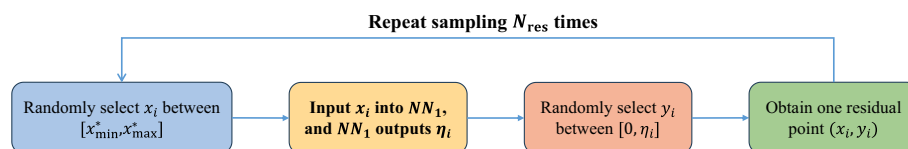


Fig. 3 Proposed algorithm for updating residual points during model training. In the chart, x_{\min}^* and x_{\max}^* represent the maximum and minimum value of the x coordinate of measurement points; trained NN_1 is

used to predict the elevation of the free surface, denoted as $\eta(x)$, as shown in Fig. 2; i is the index of the residual points, ranging from 0 to N_{res} , where N_{res} represents the total number of residual points

A two-stage optimization approach has also been frequently used for PINN models. In other words, after training PINNs for a number of iterations using the Adam optimizer, the optimization is continued using the L-BFGS optimizer [57]. Jagtap et al. [44] suggested that the L-BFGS, being a second-order method, performs better for smooth and regular solutions compared to algorithms for first-order optimization like Adam. Therefore, we also adopt the two-stage optimization approach. Unless otherwise specified, we perform L-BFGS optimization after 20,000 iterations of Adam optimization. For the initialization of WaveNets parameters, we use the Glorot method [58]. As mentioned in Sect. 3.3, we dynamically select residual points based on the predicted top surface. Specifically, we update the residual points every 200 iterations. Mini-batch training is not applied for our problem for computational efficiency. We use early stopping strategy to prevent over-fitting. When the loss does not decrease further after 300 iterations of optimization, we consider the optimization point is reached and the training process is stopped. All model training and validation were performed on a computer equipped with a 12th generation Intel(R) Core(TM) i7-12700F@2.10 GHz CPU and 16 GB of memory. The computation was solely performed on CPU.

4 Results and discussion

In this section, we demonstrate the capability of WaveNets in reconstructing flow fields through several examples. For evaluating the performance, the maximum relative error between the predicted values and the ground truth of all points is a commonly used measure. However, for the concerned wave problem, this measure is not suitable, because there are regions in the flow with the ground truth velocity or pressure close to zero leading to a large relative error therein. Therefore, we chose the normalized MSE (NMSE) as the evaluation metric, that is

$$\epsilon_z = \frac{\|\hat{\mathbf{z}} - \mathbf{z}\|_2}{\|\mathbf{z}\|_2}, \quad (34)$$

where \mathbf{z} denotes a vector of the ground truth of one of the output variables (i.e., u , v , P , or η) evaluated at the validation points, and $\hat{\mathbf{z}}$ denotes the vectorized prediction of the corresponding variable at the validation points. The operator $\|\cdot\|_2$ computes the vector norm (l^2 norm).

4.1 Case I: construction of waves on irrotational/rotational flow with measurements on the surface

In water flume tests, wave gauges are commonly employed to measure wave elevation. Meanwhile, the advancements

in remote sensing, computer vision, and related technologies have made it more convenient to gather information about the free surface in both water flume tests [22] and field measurements [25]. Hence, we first consider that only some measurements of the surface profile are available to reconstruct characteristics of the flow in the whole domain with the aid of physical knowledge.

We consider nonlinear waves on both irrotational flow ($\Omega = 0$) and rotational flow with constant vorticity ($\Omega = -2.7$). In preparing the data sets, we set the scaled water depth $d = 1$ and gravity $g = 9.8$. Correspondingly, the wavenumber is $\kappa = 1$. The numerical continuation method in Ref. [11] was used to solve Eqs. (18–20). For the whole domain, a total of $613 \times 863 = 529,019$ grid points, respectively, in the horizontal and vertical direction were used in computation. The waves are solved using the bifurcation theory with the amplitude increasing from zero (laminar flow) to the largest one that can be obtained. Figure 4 shows the bifurcation curves in the cases of $\Omega = 0$ and $\Omega = -2.7$, i.e., the relation between the wave height and Q . The wave profiles corresponding to specific points on the bifurcation curves are also shown in the figure.

As the wave amplitude increases, the nonlinearity of the wave problem becomes stronger. We thus began with the obtained wave with the largest height (height = 0.63) and $\Omega = 0$. The WaveNets model was built where NN_1 has 2 hidden layers, and each layer has 50 neurons, and NN_2 has 3 hidden layers and each layer also has 50 neurons. We studied the impact of network width and depth on computational cost and accuracy, as described in Appendix 1. It was found that further increasing the depth and width of the NNs can not enhance their predictive performance. Hence, the same WaveNets architecture was used for the subsequent examples if not otherwise specified.

For model training, we selected 1000 points in the flow field as residual points, and 50 points from the lower bound and 50 points from the free surface as the boundary condition points. Figure 5a illustrates the initial distribution of sampling points during WaveNets training. The red points represent measurement points, and their positions, velocity, and pressure values are known. The pressure values at these points equal to atmospheric pressure ($P_{\text{atm}} = 0$). The blue points serve as residual points for the purpose of imposing physical constraints related to the partial differential equations. The black dots denote points located at the bottom, on which zero vertical velocity is enforced. It is important to note that during the training process, both the positions of the residual points and the points on the lower bound undergo changes. The WaveNets model was trained for 20,000 iterations using the Adam optimizer with the setting described in the previous section. Then, the L-BFGS optimizer was used to further train the NNs. The L-BFGS training process automatically stops

Fig. 4 Data sets for Case I: **a** bifurcation curve when $\Omega = 0$; **b** bifurcation curve when $\Omega = -2.7$; **c, e, g** streamlines of waves with heights = 0.21, 0.41 and 0.63 when $\Omega = 0$; **d, f, h** streamlines of waves with heights = 0.50, 1.00 and 1.32 when $\Omega = -2.7$

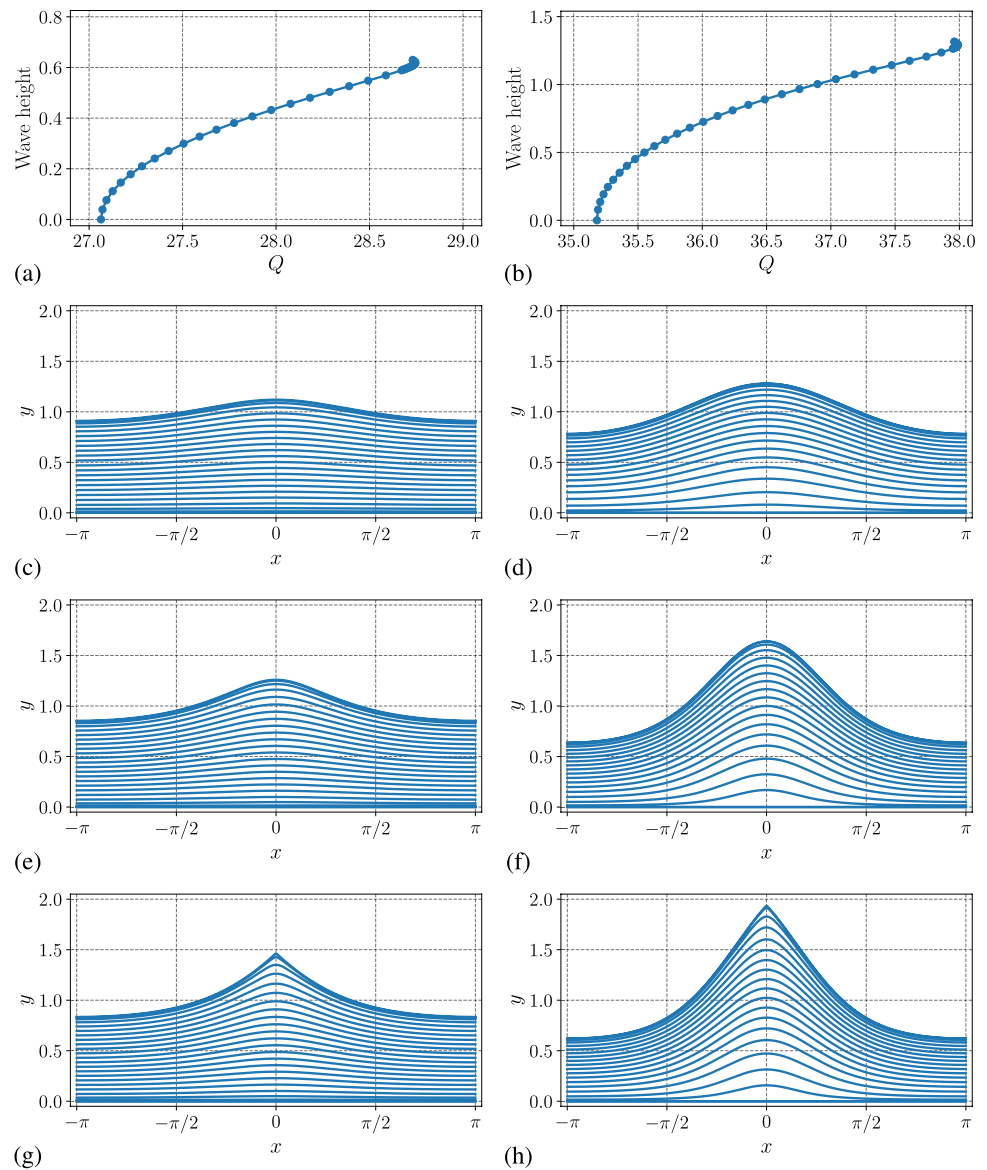
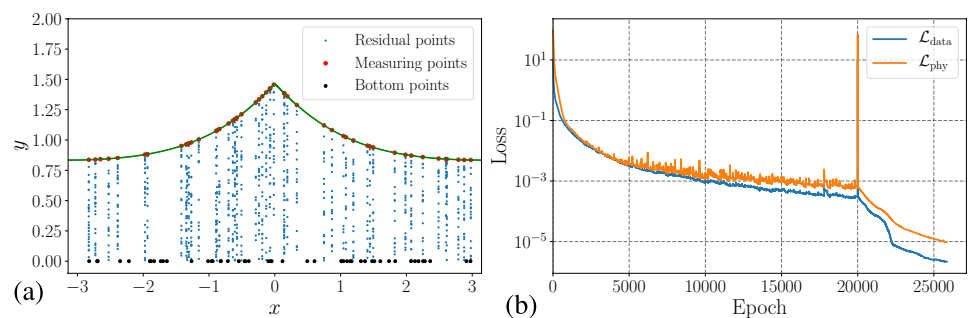


Fig. 5 **a** Initial distribution of training points and **b** the loss evolution in training process. The solid green curve in **(a)** represents the reference solution of the free surface

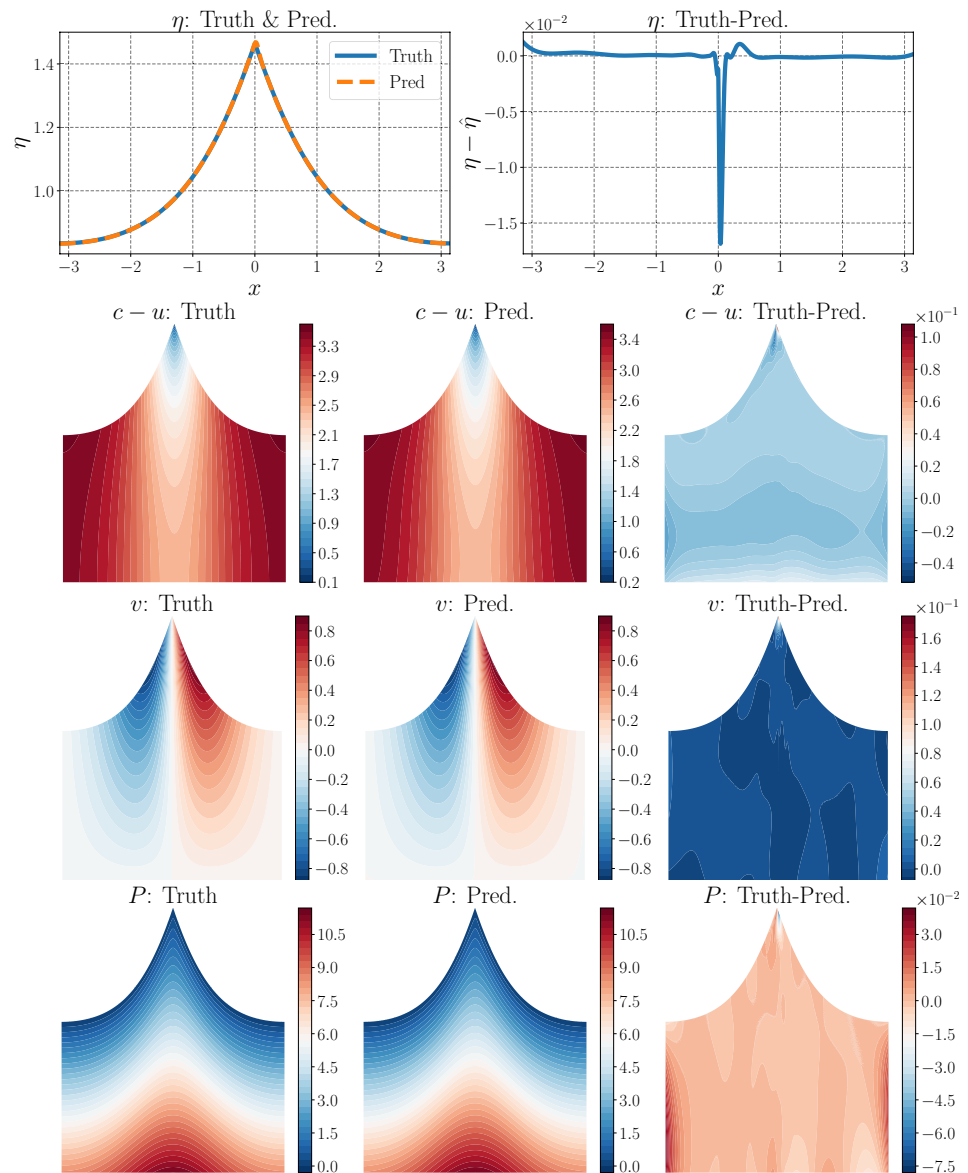


according to the variation of the loss function. Figure 5b shows the evolution of the loss function during training. It is observed that with the increasing epoch (iteration) when using the Adam optimizer, the decreasing rate of the loss function is gradually declined, and after L-BFGS

optimizer is applied the loss function decreases significantly again.

The trained model is then used for evaluating velocity and pressure fields in the whole domain. The predictions are compared with the ground truth values in Fig. 6.

Fig. 6 WaveNets performance evaluation in case I when $\Omega = 0$ and wave height = 0.63

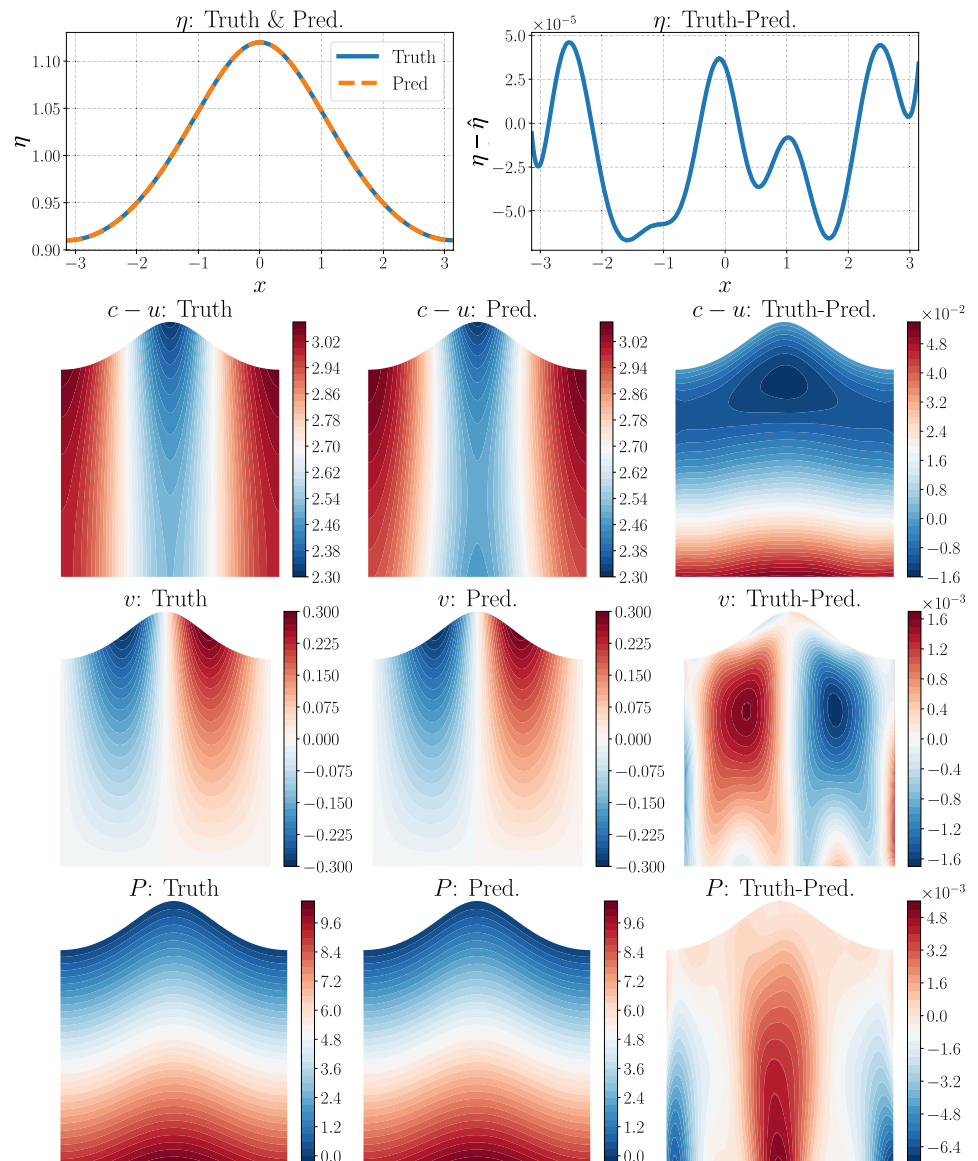


It can be observed that the errors (difference between ground truth and predicted values) are relatively small, and the points with the largest errors are concentrated to the wave crest where nonlinearity is the strongest. Using Eq. (34), NMSE of the predictions are calculated as $\epsilon_u = 2.07 \times 10^{-3}$, $\epsilon_v = 1.42 \times 10^{-2}$, $\epsilon_P = 8.75 \times 10^{-4}$, and $\epsilon_\eta = 2.51 \times 10^{-3}$. In this case, we also tried the dynamic weighting scheme in training, which involves dynamically adjusting the weight (λ) in the total loss as expressed in Eq. (31). However, no significant improvement has been achieved. This can be partially explained by Fig. 5b where both variations of the data loss ($\mathcal{L}_{\text{data}}$) and the physical loss (\mathcal{L}_{phy}) are plotted and it is seen they are comparable and decrease simultaneously. In other words, in this case, the data loss and the physics loss contribute almost equally to the training. Therefore, using dynamic weighting to

balance their respective contributions seems unnecessary and ineffective.

Waves of smaller heights ($\Omega = 0$) have also been used to test WaveNets. The parameter setting and training strategy were the same as in the previous example. The prediction results using the respective trained models are shown in Figs. 7 and 8 when the wave height is 0.21 and 0.42 correspondingly. Compared to the wave with the largest height, the maximum prediction errors in these two cases have decreased by approximately an order of magnitude. Moreover, the regions with larger errors are mainly concentrated along the left, right, and bottom boundaries, rather than at the wave crest. This observation suggests that in cases of smaller wave heights, indicating weaker nonlinearity of the problem, the reconstruction of the flow field becomes easier and more accurate when using WaveNets.

Fig. 7 WaveNets performance evaluation in case I when $\Omega = 0$ and wave height = 0.21



We further consider the wave of the largest height shown in Fig. 4b when $\Omega = -2.7$. Similarly, the data set includes velocity and pressure fields on 529,019 grid points for the whole domain. We performed the same sampling process, setting 1000 points as residual points in the flow and 50 points on each of the upper and lower bounds. Figure 9 presents the true values, predicted values, and errors. In this case, it can be observed that the maximum errors of the predicted solutions are still within an acceptable range. Specifically, the maximum absolute error is approximately 5.81×10^{-3} for the free surface elevation, 1.26×10^{-1} for the horizontal velocity, 1.56×10^{-1} for the vertical velocity, and 4.06×10^{-2} for the pressure. It should be noted that the maximum errors are primarily concentrated in two regions: close to wave crest and the left and right bounds. The relatively large

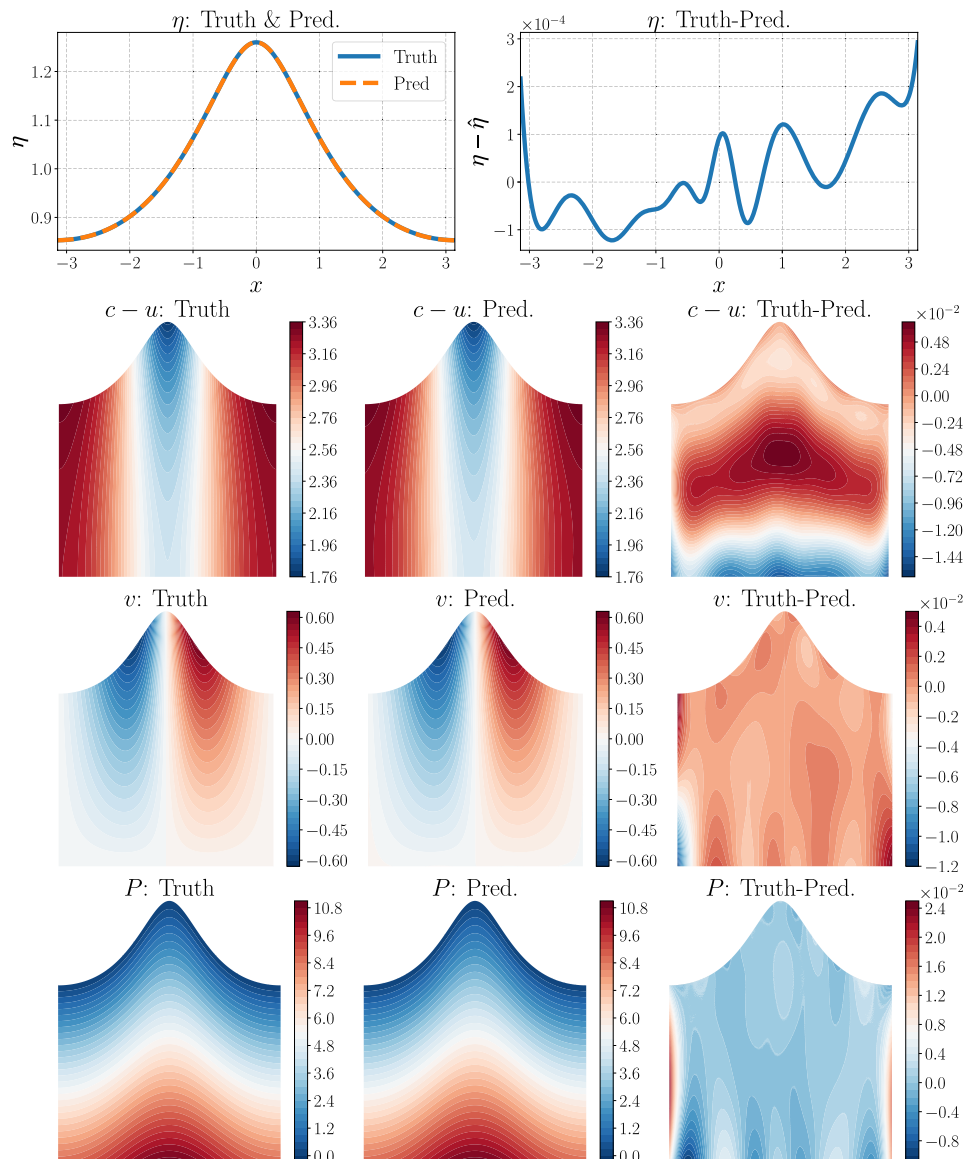
errors occur near the crest because of the strong nonlinear behavior in this region, which is challenging to accurately capture. The relatively large errors at the left and right bounds occur because no measurements there were used in model training. The NMSEs of the predictions are $\epsilon_\eta = 8.95 \times 10^{-4}$, $\epsilon_u = 7.78 \times 10^{-4}$, $\epsilon_v = 3.03 \times 10^{-3}$, and $\epsilon_P = 8.33 \times 10^{-4}$.

We compute the vorticity from the predicted velocity field using automatic differentiation, following Eq. (5), i.e.,

$$\hat{\Omega} = \frac{\partial \hat{v}}{\partial x} - \frac{\partial \hat{u}}{\partial y}.$$

Figure 10 plots the computed vorticity distribution along with its histogram. It can be observed from the figure that

Fig. 8 WaveNets performance evaluation in case I when $\Omega = 0$ and wave height = 0.41



most of the vorticity estimates are between -2.75 and -2.65 , which indicates that the calculated vorticity has high accuracy for most points. A closeup view of the wave crest shows that there is a significant error in the calculated vorticity at the wave crest. Recall that at the crest of the wave with the largest height the velocity field is constructed with relatively larger errors as shown in Fig. 9. Hence, the error in vorticity computed from the velocity field using the automatic differentiation of the velocity is exaggerated. For waves with a smaller height, the accuracy of vorticity estimates at the wave crest is higher as the velocity field is more accurate, see Figs. 7 and 8.

In the previous examples, we have successfully reconstructed full-field flow field under large-amplitude waves using few measurements of elevation and velocity at the top surface. Noticeably, it only requires utilizing elevation data

from the free surface measurement points, supplemented by values from one or two velocity measurement points (which can be located either on the free surface or within the flow field), to reconstruct the distribution of the flow field, as demonstrated in Fig. 11). For the steady problems, if $(x, y, u - c, v, p)$ is a solution, then $(x, y, c - u, -v, p)$ is also a solution [59]. The inclusion of one or two velocity measurement points here is intended to determine the orientation of the flow field, steering the predicted values of WaveNets towards one particular solution. When comparing Figs. 9 and 11, the predictive performance is comparable, but notably, the results are more accurate when additional velocity measurement points are included. This suggests that incorporating data loss terms computed from velocity measurements contributes to enhancing the training accuracy of the neural network.

Fig. 9 WaveNets performance evaluation in case I when $\Omega = -2.7$ and wave height = 1.32

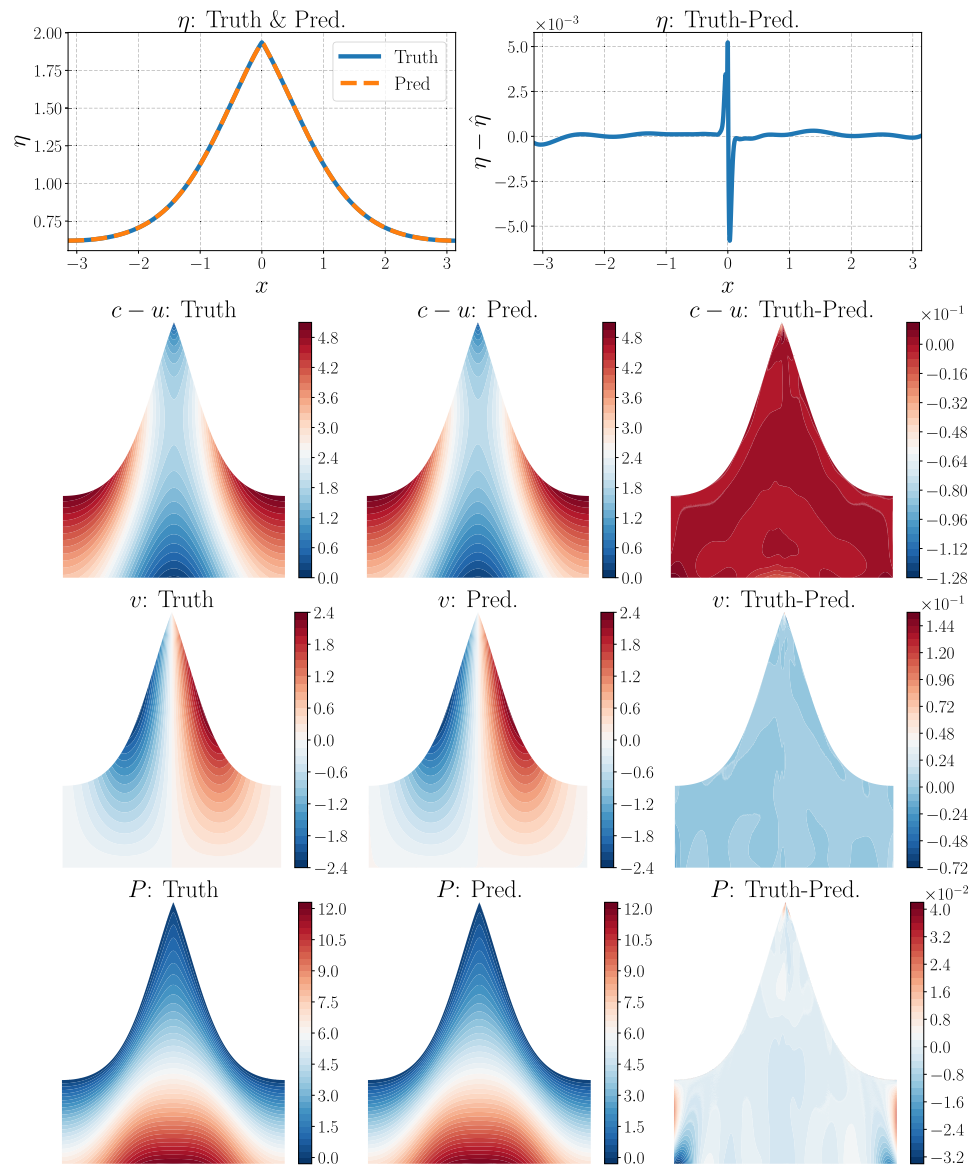
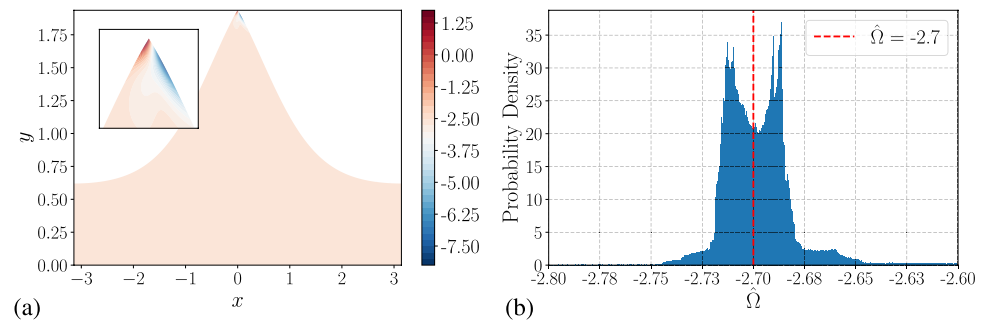


Fig. 10 Vorticity $\hat{\Omega}$ computed from the predicted velocity field in Case I with $\Omega = -2.7$ and wave height=1.32: **a** $\hat{\Omega}$ distribution; **b** histogram of $\hat{\Omega}$

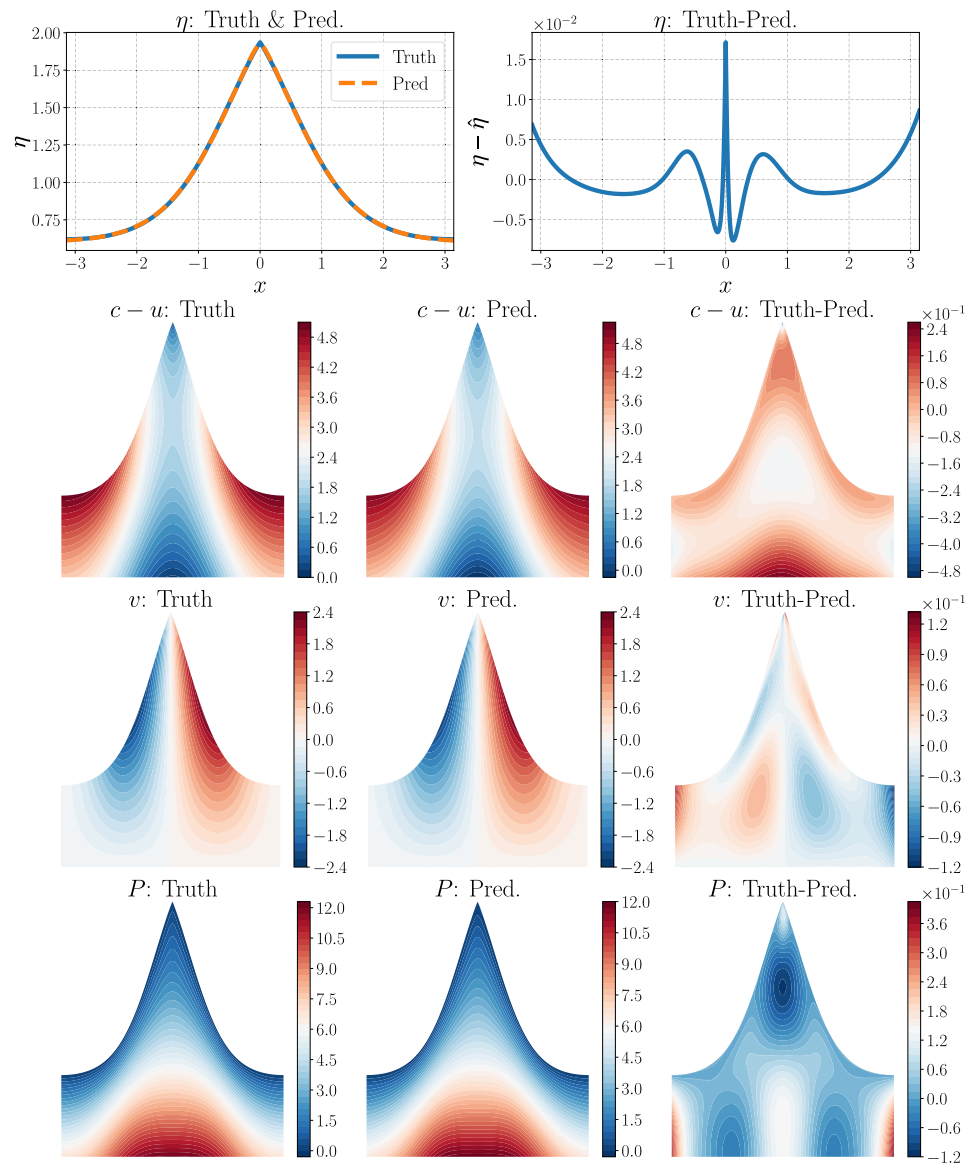


4.2 Case II: construction of waves on rotational flow with measurement data both on and under the surface

In this case, we consider the situation when measurements

under the surface are available, e.g., pressure and velocity in the flow. The width, depth, and other hyperparameters of WaveNets remain the same as in Case I. The total number of measurement points is still 50, but only 10 points are at the top and the rest measurements are at positions in the

Fig. 11 Performance evaluation of WaveNets trained with only elevation measurements at the top surface when $\Omega = -2.7$ and wave height = 1.32



flow. Figure 12a–d illustrates the evolution of the distribution of sampling points. In the initial stages of WaveNets training, residual points are randomly distributed within the region $[x_{\min}^*, x_{\max}^*] \times [0, y_{\max}^*]$, where $*$ denotes measurement points. It can be seen that as the training progresses, the residual points are gradually scattered over the entire area below the free surface, making the physical constraints imposed on the residual points more meaningful. Due to space limitations, we only show the prediction results of $\Omega = -2.7$ and height=1.32 in Fig. 13 and it is seen that flow field can still be reconstructed although the maximum absolute error is slightly larger than that in Case I. The NMSEs of the predictions are calculated as $\epsilon_\eta = 1.11 \times 10^{-2}$, $\epsilon_u = 3.19 \times 10^{-3}$, $\epsilon_v = 7.29 \times 10^{-3}$, and $\epsilon_P = 9.91 \times 10^{-4}$.

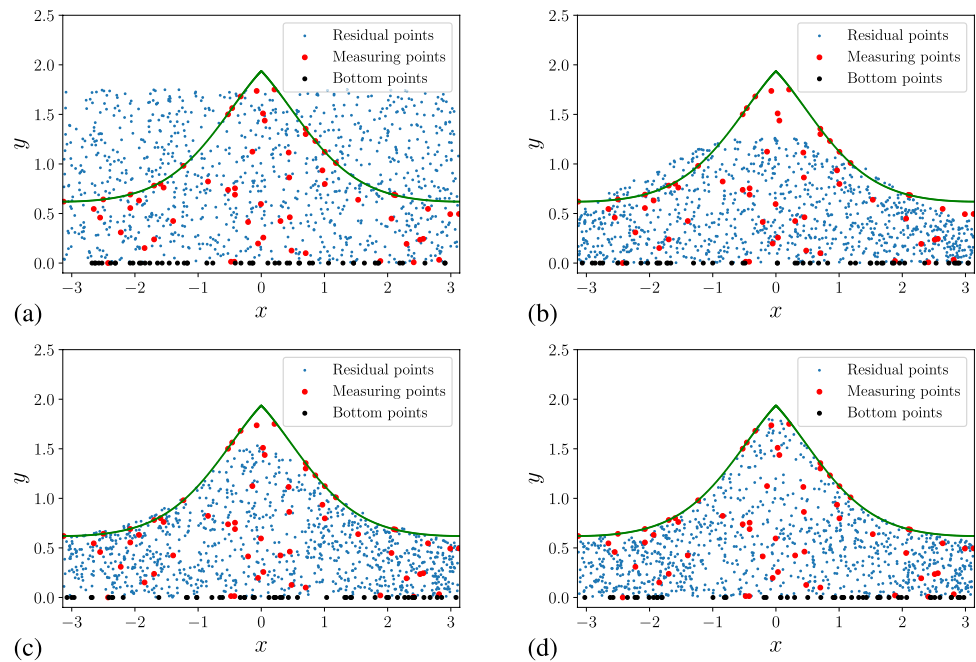
4.3 Case III: construction of waves on rotational flow with two layers

In the third case, we consider waves on rotational flow with two layers, to further validate the generality of WaveNets. The vorticity function is set as

$$\gamma = \begin{cases} \gamma_1, & \text{the upper layer} \\ 0, & \text{the lower layer,} \end{cases} \quad (35)$$

where γ_1 denotes vorticity of the top layer and the lower layer is irrotational. The waves on two-layer rotational flow were also solved using the numerical continuation method [11]. Again, we set the scaled water depth $d = 1$, gravity $g = 9.8$, and $\gamma_1 = -10$. For laminar flow (zero-amplitude wave) the

Fig. 12 Distribution of training points in Case II for $\Omega = -2.7$ and wave height = 1.32: **a** initial distribution; **b–d** updated distribution in training process. The solid green curve represents the reference solution of the free surface



depth of the upper layer is 0.2. The grid size used for computation was $613 \times 975 = 597,675$ with refined meshgrid near the interface of the two layers. The bifurcation curve is shown in Fig. 14a and the profile of the solved wave with the largest height is shown in Fig. 14b. It is noted that the depth of the upper layer varies along with the wave amplitude [11]. For the wave in Fig. 14b, the depth of the upper layer is about 0.27.

We used the same architecture and hyper-parameter setting of WaveNets as in Case I. The prediction results are shown in Fig. 15. In this case, the maximum absolute error of the predicted solution is similar to the previous cases, and the areas with larger absolute errors are mainly distributed near the wave crest. In addition, at the interface of the two layers where the vorticity is discontinuous, the errors in the predicted velocity $c - u$ and v increase significantly, but they remain within an acceptable range. The NMSEs are $\epsilon_\eta = 3.66 \times 10^{-3}$, $\epsilon_u = 6.54 \times 10^{-3}$, $\epsilon_v = 2.11 \times 10^{-2}$, and $\epsilon_p = 2.38 \times 10^{-3}$.

Figure 16 shows the vorticity distribution computed from the predicted velocity field in this case. It is clear in Fig. 16a that two layers of different vorticity are present and the histogram in Fig. 16b shows that the vorticity values are, respectively, -10 and 0 . It is also noted close to the interface of the two layers, there is a transition band where the vorticity value varies continuously from -10 to zero downwards. The discontinuous vorticity distribution is difficult to recover, although the transition band is narrow. It is expected for cases with continuous vorticity distribution, more accurate vorticity recovery can be achieved. In Fig. 16b, the probability density is higher near $\hat{\Omega} = 0$ while is lower near $\hat{\Omega} = -10$, because in

this particular case, the region where $\Omega = 0$ is significantly larger than the region where $\Omega = -10$, as shown in Fig. 14.

4.4 Improved estimate of vorticity distribution

In the previous three cases, the accuracy of the vorticity reconstruction from predicted velocity field is lower than the prediction of velocity of itself. Here, we propose to use the inferring capability of PINN model to improve vorticity estimate accuracy. The parameter identification capability of PINNs has been successfully applied to various problems, such as high-speed fluid flow [38] and heat conduction problems [50].

Considering the constant vorticity case, the physical relationship between the horizontal velocity u , vertical velocity v , and vorticity Ω within the flow field can also be incorporated into WaveNets, as illustrated in Fig. 2. Consequently, this loss term can be expressed as

$$e_7 = \hat{u}_y - \hat{v}_x - \hat{\Omega}. \quad (36)$$

where $\hat{\Omega}$ represents the vorticity of the flow field and is treated as a variable in the neural network. The overall loss function can thus be represented

$$\begin{aligned} \mathcal{L}'_{\text{phy}} = & \frac{1}{N_{\text{in}}} \sum_{i=1}^{N_{\text{in}}} (e_1^2 + e_2^2 + e_3^2 + e_7^2) + \frac{1}{N_{\text{up}}} \sum_{i=1}^{N_{\text{up}}} (e_4^2 + e_5^2) \\ & + \frac{1}{N_{\text{low}}} \sum_{i=1}^{N_{\text{low}}} e_6^2. \end{aligned} \quad (37)$$

Fig. 13 WaveNets performance evaluation for case II when $\Omega = -2.7$ and wave height = 1.32

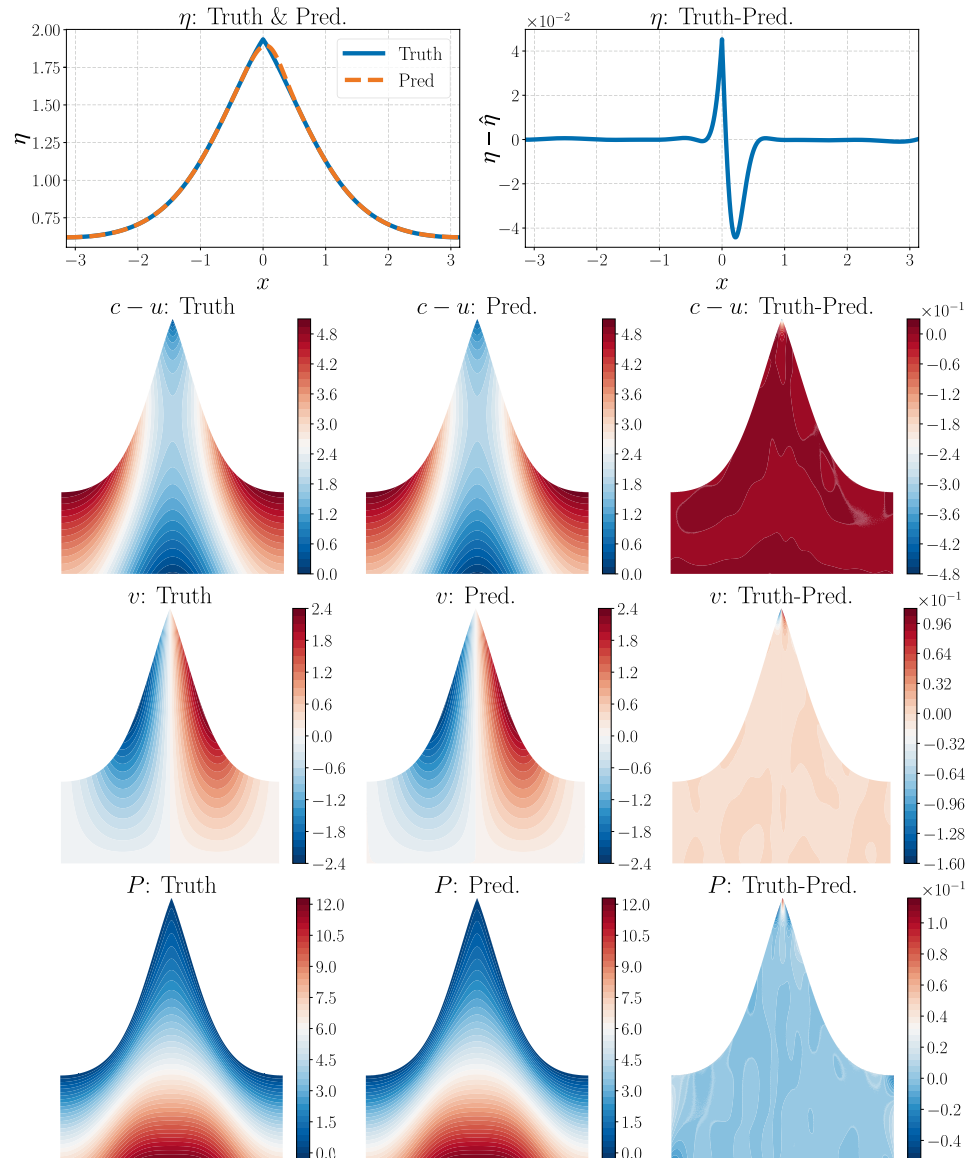
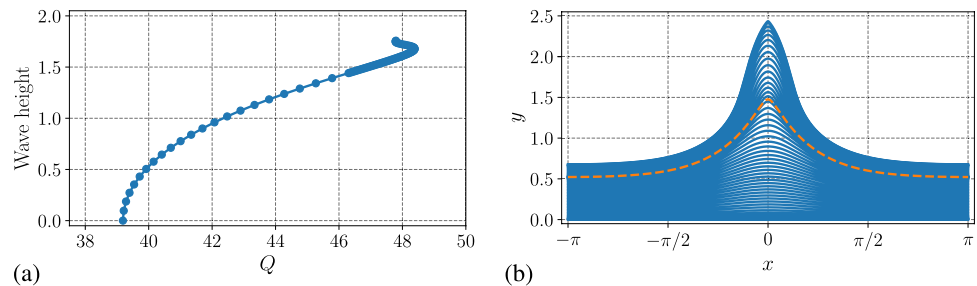


Fig. 14 Data sets for case III: **a** bifurcation curve; **b** streamlines of the wave with the largest height (the dashed line indicates the interface of the two layers)



During the neural network training process, $\hat{\Omega}$ and other WaveNets parameters (**w**, **b**) are automatically adjusted based on the principle of minimizing the loss function [33].

For demonstration, we use the wave with the largest amplitude in Fig. 4b corresponding to $\Omega = -2.7$.

Measurement data was only available on the top surface. The hyperparameters of the neural network are the same as in the previous case, the number of internal residual points is still 1000, and on the free surface and the bottom of the flow field there are, respectively, 50 sampling points.

Fig. 15 WaveNets performance evaluation for Case III for the wave with the largest height

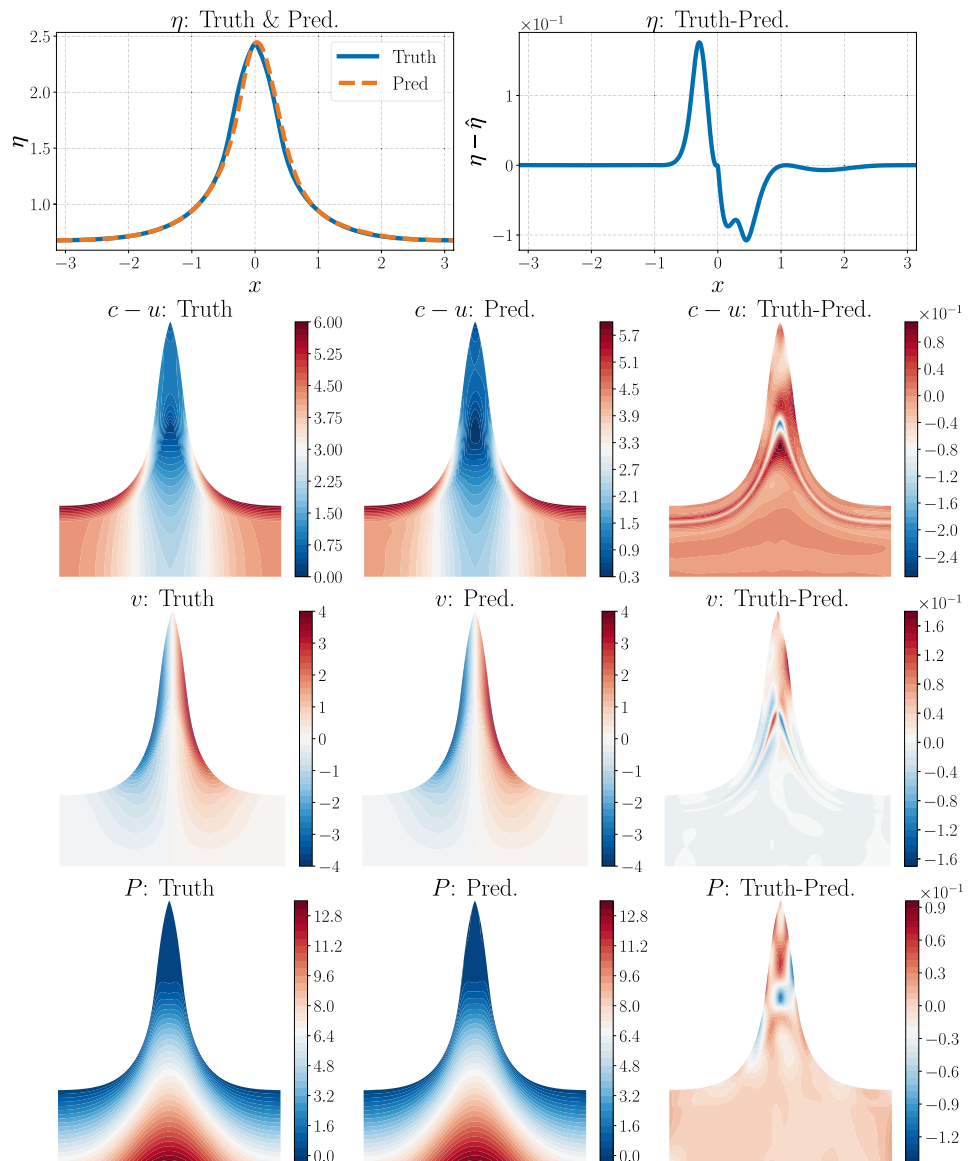


Fig. 16 Vorticity computed from WaveNets predicted velocity field: **a** distribution; **b** histogram

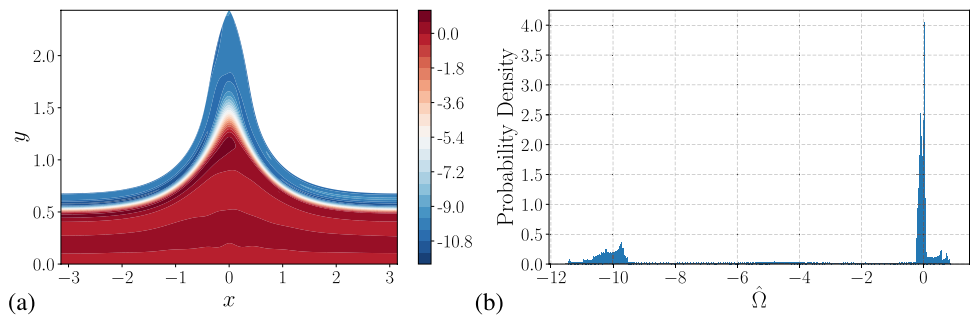
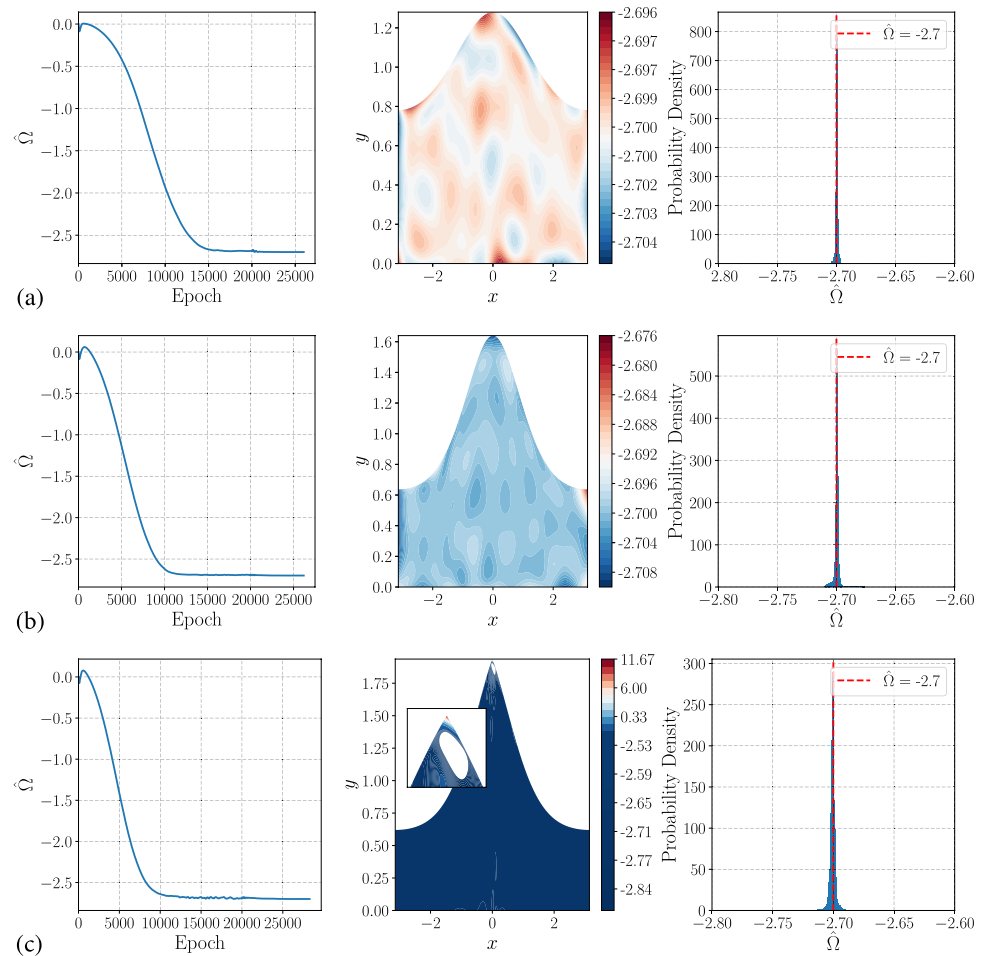


Figure 17 shows the evolution of the vorticity $\hat{\Omega}$ with zero initial value during the training process. It is seen that after nearly 10,000 iterations of training, $\hat{\Omega}$ has approached the actual value of -2.7 . The results for three waves of different heights are shown in Fig. 17. Compared to the results

in Fig. 10, adding the constraint term on vorticity distribution in WaveNets largely increases the accuracy of vorticity distribution construction. Moreover, the velocity field prediction accuracy is also improved to some extent. This case also demonstrates that introducing appropriate redundant

Fig. 17 Vorticity estimate by WaveNets with additional constraints for Case I when $\Omega = -2.7$ for waves with different heights: **a** wave height = 0.50; **b** wave height = 1.00; **c** wave height = 1.32



physical constraints into WaveNets can improve prediction accuracy.

5 Conclusions

This study formulates WaveNets for recovery of large-amplitude periodic waves propagating on rotational flow. For such a free boundary problem, we use two NNs, respectively, to reconstruct the free surface and the flow characteristics (velocity and pressure). The Euler equations and other physics knowledges about the flow mechanics are incorporated as physical constraints and taken into account in the loss function. Furthermore, we propose a residual point sampling method applicable to free surface problems. The method dynamically selects residual points based on the predictions of NNs for free surface, ensuring that the residuals are evaluated at points in the domain. This residual point sampling method can be applied to other free boundary problems with complex geometric shapes.

Waves of small to the largest height on both irrotational flow and rotational flow are solved using the numerical

continuation method and then used for WaveNets training and validation. Results show that by using a small amount of pointwise measurements on the top surface, WaveNets can reconstruct wave surface, velocity and pressure field with good accuracy. The velocity and pressure fields are recovered with higher accuracy as compared to the surface profile. It is also found only using elevation measurements on sampled points at the surface is sufficient for WaveNets to reconstruct the flow field while horizontal velocity direction at specific points need to be predetermined because of the indetermination of the wave problem. When some measurements in the flow field are available, the number of measurements at the surface can be reduced and WaveNets performs comparable. For complicated cases of wave close to the largest height on two layer flow with discontinuous vorticity, WaveNets also performs well in surface and flow field construction from limited amount of measurements. From the predicted velocity field, the computed distribution shows degraded accuracy as compared to the velocity field. Adding physical constraints according to prior information of the vorticity distribution can largely improve the estimate accuracy.

Table 1 NMSE of prediction error using WaveNets with different widths and depths

Depth×width		NMSE				Calculation time (s)
NN_1	NN_2	$c - u$	v	P	η	
2×50	3×50	$5.50\text{e-}03$	$7.92\text{e-}04$	$1.61\text{e-}03$	$9.63\text{e-}04$	$9.21\text{e+}01$
2×100	3×100	$4.34\text{e-}03$	$7.17\text{e-}04$	$1.10\text{e-}03$	$8.67\text{e-}04$	$2.56\text{e+}02$
4×50	6×50	$2.26\text{e-}03$	$5.20\text{e-}04$	$1.39\text{e-}03$	$3.52\text{e-}04$	$2.06\text{e+}02$
4×100	6×100	$1.07\text{e-}02$	$9.88\text{e-}04$	$2.43\text{e-}03$	$1.95\text{e-}04$	$6.94\text{e+}02$
6×50	9×50	$8.66\text{e-}03$	$6.23\text{e-}04$	$1.39\text{e-}03$	$2.52\text{e-}04$	$3.68\text{e+}02$
6×100	9×100	$7.09\text{e-}03$	$7.26\text{e-}04$	$2.05\text{e-}03$	$1.86\text{e-}03$	$1.49\text{e+}03$

Several relevant studies on WaveNets are expected in future. WaveNets need to be validated using experimental data in wave flume or field measurement data. Bayesian method can be incorporated in WaveNets to account for the uncertainties in measurement data to improve the performance, particularly for dealing with real-world data set.

Appendix A The influence of NN width and depth

We have conducted a careful study on the effects of network width and depth of WaveNets in Fig. 2 on computation cost and accuracy. The case used for the study is the wave of height = 0.63 on irrotational flow ($\Omega = 0$) in Case I, where only measurement data on the top surface is available. Six WaveNets of different width and depth are selected for comparison. In all the test cases, except for NN depth and width, the values of other hyper-parameters and setting remain the same. The accuracy and the computation time are listed in Table 1. Generally, deeper and wider DNNs have better capability to approximate continuous functions. However, for this particular problem, increasing the depth and width of the neural network does not always result in improved predictive performance (including velocity, pressure, and free surface elevation). Instead, it significantly increases computational costs. Given that vorticity is computed through automatic differentiation, as per Eq. (5), the quality of vorticity reconstruction is directly linked to the reconstruction quality of the velocity field. Consequently, augmenting the width and depth of the neural network has a relatively limited impact on the reconstruction quality of vorticity as well. Considering both accuracy and computational efficiency, the size of NN_1 and NN_2 are set to 2×50 and 3×50 in the remaining sections of this paper.

Funding Open access funding provided by Swiss Federal Institute of Technology Zurich. This work is partly supported by the National Natural Science Foundation of China (Grant No. 51978506) and the

Science and Technology Cooperation Project of Shanghai Qi Zhi Institute (Grant No. SQZ202310).

Data Availability This is not applicable because there is no experimental data in this paper. This paper does contain mention of computational results produced by the authors' personal codes but none of them is available to the public. Some of them could be provided upon request.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Thomas G (1981) Wave-current interactions: an experimental and numerical study. Part 1. Linear waves. *J Fluid Mech* 110:457–474
2. Thomas G (1990) Wave-current interactions: an experimental and numerical study. Part 2. Nonlinear waves. *J Fluid Mech* 216:505–536
3. Constantin A (2011) Nonlinear water waves with applications to wave-current interactions and tsunamis. In: *CMBS-NSF Reg Conf Ser Appl Math*, vol 81. SIAM, Philadelphia
4. Chen L, Basu B, Martin C-I (2021) On rotational flows with discontinuous vorticity beneath steady water waves near stagnation. *J Fluid Mech* 912:A44
5. Chen L, Basu B (2018) Wave-current interaction effects on structural responses of floating offshore wind turbines. *Wind Energy* 22:327–339. <https://doi.org/10.1002/we.2288>
6. Kang A, Zhu B, Lin P, Ju J, Zhang J, Zhang D (2020) Experimental and numerical study of wave-current interactions with a dumbbell-shaped bridge cofferdam. *Ocean Eng* 210:107433
7. Nguyen H, Chen L, Basu B (2023) On the influence of large amplitude nonlinear regular waves on the structural response of spar-type floating offshore wind turbines. *Ocean Eng* 269:113448

8. Dyachenko SA, Hur VM (2019) Stokes waves with constant vorticity: I. Numerical computation. *Stud Appl Math* 142(2):162–189
9. Dyachenko SA, Hur VM (2019) Stokes waves with constant vorticity: folds, gaps and fluid bubbles. *J Fluid Mech* 878:502–521
10. Amann D, Kalimeris K (2018) A numerical continuation approach for computing water waves of large wave height. *Eur J Mech B/Fluids* 67:314–328
11. Chen L, Basu B (2021) Numerical continuation method for large-amplitude steady water waves on depth-varying currents in flows with fixed mean water depth. *Appl Ocean Res* 111:102631
12. Ali A, Kalisch H (2013) Reconstruction of the pressure in long-wave models with constant vorticity. *Eur J Mech B/Fluids* 37:187–194
13. Amann D, Kalimeris K (2018) Numerical approximation of water waves through a deterministic algorithm. *J Math Fluid Mech* 20:1815–1833
14. Zhang J-S, Zhang Y, Jeng D-S, Liu PL-F, Zhang C (2014) Numerical simulation of wave-current interaction using a RANS solver. *Ocean Eng* 75:157–164
15. Turner MR, Bridges TJ (2016) Time-dependent conformal mapping of doubly-connected regions. *Adv Comput Math* 42:947–972
16. Swan C, Cummins IP, James RL (2001) An experimental study of two-dimensional surface water waves propagating on depth-varying currents. Part 1. Regular waves. *J Fluid Mech* 428:273–304
17. Oliveras KL, Vasan V, Deconinck B, Henderson D (2012) Recovering the water-wave profile from pressure measurements. *SIAM J Appl Math* 72(3):897–918
18. Constantin A (2014) Estimating wave heights from pressure data at the bed. *J Fluid Mech* 743:R2
19. Basu B (2018) Wave height estimates from pressure and velocity data at an intermediate depth in the presence of uniform currents. *Phil Trans R Soc A* 376(2111):20170087
20. Tsai C-H, Huang M-C, Young F-J, Lin Y-C, Li H-W (2005) On the recovery of surface wave by pressure transfer function. *Ocean Eng* 32(10):1247–1259
21. Marino M, Rabionet IC, Musumeci RE (2022) Measuring free surface elevation of shoaling waves with pressure transducers. *Cont Shelf Res* 245:104803
22. Paprota M, Sulisz W, Reda A (2016) Experimental study of wave-induced mass transport. *J Hydraul Res* 54(4):423–434
23. Fernando PC, Guo J, Lin P (2011) Wave-current interaction at an angle I: experiment. *J Hydraul Res* 49(4):424–436
24. Lund B, Graber HC, Hessner K, Williams NJ (2015) On shipboard marine x-band radar near-surface current “calibration”. *J Atmos Ocean Technol* 32(10):1928–1944
25. Metoyer S, Barzegar M, Bogucki D, Haus BK, Shao M (2021) Measurement of small-scale surface velocity and turbulent kinetic energy dissipation rates using infrared imaging. *J Atmos Ocean Technol* 38(2):269–282
26. Goodfellow I, Bengio Y, Courville A (2016) *Deep Learning*. MIT press, Cambridge
27. Lei X, Xia Y, Wang A, Jian X, Zhong H, Sun L (2023) Mutual information based anomaly detection of monitoring data with attention mechanism and residual learning. *Mech Syst Signal Process* 182:109607
28. Lei X, Dong Y, Frangopol DM (2023) Sustainable life-cycle maintenance policymaking for network-level deteriorating bridges with a convolutional autoencoder-structured reinforcement learning agent. *J Bridge Eng* 28(9):04023063
29. Jain P, Deo M (2006) Neural networks in ocean engineering. *Ships Offshore Struct* 1(1):25–35
30. Dibike YB, Minns AW, Abbott MB (1999) Applications of artificial neural networks to the generation of wave equations from hydraulic data. *J Hydraul Res* 37(1):81–97
31. Xuan A, Shen L (2023) Reconstruction of three-dimensional turbulent flow structures using surface measurements for free-surface flows based on a convolutional neural network. *J Fluid Mech* 959:A34
32. Rumsey CL, Coleman GN (2022) NASA symposium on turbulence modeling: roadblocks, and the potential for machine learning. Technical Memorandum NASA/TM-20220015595, Langley Research Center Hampton, Virginia 23681-2199
33. Raissi M, Perdikaris P, Karniadakis GE (2019) Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J Comput Phys* 378:686–707
34. Karniadakis GE, Kevrekidis IG, Lu L, Perdikaris P, Wang S, Yang L (2021) Physics-informed machine learning. *Nat Rev Phys* 3(6):422–440
35. Jin X, Cai S, Li H, Karniadakis GE (2021) NSFnets (Navier-Stokes flow nets): physics-informed neural networks for the incompressible Navier-Stokes equations. *J Comput Phys* 426:109951
36. Eivazi H, Tahani M, Schlatter P, Vinuesa R (2022) Physics-informed neural networks for solving Reynolds-averaged Navier-Stokes equations. *Phys Fluids* 34(7):075117
37. Tang H, Liao Y, Yang H, Xie L (2022) A transfer learning-physics informed neural network (TL-PINN) for vortex-induced vibration. *Ocean Eng* 266:113101
38. Mao Z, Jagtap AD, Karniadakis GE (2020) Physics-informed neural networks for high-speed flows. *Comput Methods Appl Mech Engrg* 360:112789
39. Lee SY, Park C-S, Park K, Lee HJ, Lee S (2022) A physics-informed and data-driven deep learning approach for wave propagation and its scattering characteristics. *Eng Comput* 39(4):2609–2625
40. Cai S, Mao Z, Wang Z, Yin M, Karniadakis GE (2021) Physics-informed neural networks (PINNs) for fluid mechanics: a review. *Acta Mech Sin* 37:1727–1738
41. Wang S, Teng Y, Perdikaris P (2021) Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM J Sci Comput* 43(5):3055–3081
42. Wang S, Yu X, Perdikaris P (2022) When and why Pinns fail to train: a neural tangent kernel perspective. *J Comput Phys* 449:110768
43. Lu L, Meng X, Mao Z, Karniadakis GE (2021) DeepXDE: a deep learning library for solving differential equations. *SIAM Rev* 63(1):208–228
44. Jagtap AD, Kawaguchi K, Karniadakis GE (2020) Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *J Comput Phys* 404:109136
45. Chen X, Chen R, Wan Q, Xu R, Liu J (2021) An improved data-free surrogate model for solving partial differential equations using deep neural networks. *Sci Rep* 11:19507
46. Peng W-Q, Pu J-C, Chen Y (2022) PINN deep learning method for the Chen-Lee-Liu equation: Rogue wave on the periodic background. *Commun Nonlinear Sci Numer Simul* 105:106067
47. Jagtap AD, Mitsotakis D, Karniadakis GE (2022) Deep learning of inverse water waves problems using multi-fidelity data: application to Serre-Green-Naghdi equations. *Ocean Eng* 248:110775
48. Huang YH, Xu Z, Qian C, Liu L (2023) Solving free-surface problems for non-shallow water using boundary and initial conditions-free physics-informed neural network (bif-PINN). *J Comput Phys* 479:112003
49. Wang S, Perdikaris P (2021) Deep learning of free boundary and Stefan problems. *J Comput Phys* 428:109914
50. Cai S, Wang Z, Wang S, Perdikaris P, Karniadakis GE (2021) Physics-informed neural networks for heat transfer problems. *J Heat Transfer* 143(6):060801
51. Constantin A, Strauss WA (2004) Exact steady periodic water waves with vorticity. *Comm Pure Appl Math* 57(4):481–527
52. Henry D (2013) Large amplitude steady periodic waves for fixed-depth rotational flows. *Comm Part Differ Equ* 38(6):1015–1037

53. Henry D (2013) Steady periodic waves bifurcating for fixed-depth rotational flows. *Q Appl Math* 71(3):455–487
54. Ni P, Li Y, Sun L, Wang A (2022) Traffic-induced bridge displacement reconstruction using a physics-informed convolutional neural network. *Comput Struct* 271:106863
55. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M, Levenberg J, Monga R, Moore S, Murray DG, Steiner B, Tucker P, Vasudevan V, Warden P, Wicke M, Yu Y, Zheng X (2016) TensorFlow: A system for Large-Scale machine learning. In: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). USENIX Association, Savannah, GA, pp 265–283. <https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi>
56. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
57. Liu DC, Nocedal J (1989) On the limited memory bfgs method for large scale optimization. *Math Program* 45(1–3):503–528
58. Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp 249–256. JMLR Workshop and Conference Proceedings
59. Chen L, Basu B (2021) Numerical investigations of two-dimensional irrotational water waves over finite depth with uniform current. *Appl Anal* 100(6):1247–1255

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.