

Residual-connected physics-informed neural network for anti-noise wind field reconstruction

Runze Tian, Peng Kou*, Yuanhang Zhang, Mingyang Mei, Zhihao Zhang, Deliang Liang

State Key Laboratory of Electrical Insulation and Power Equipment, Shaanxi Key Laboratory of Smart Grid, School of Electrical Engineering, Xi'an Jiaotong University, Xi'an 710049, China

HIGHLIGHTS

- A residual-connected physics-informed neural network (RC-PINN) is proposed.
- The advantage of the RC-PINN is proven mathematically.
- An anti-noise wind field reconstruction method is established based on the RC-PINN.
- The anti-noise mechanism behind the proposed method is demonstrated.

ARTICLE INFO

Keywords:
 Physics-informed neural network
 Residual connection
 Wind field reconstruction
 LIDAR measurements
 Measurement noise

ABSTRACT

Physics-informed neural network (PINN)-based methods have recently been applied to reconstruct the spatiotemporal wind field based on LIDAR measurements. However, the accuracy of this reconstruction considerably degrades when LIDAR measurement noise exists. Unfortunately, LIDAR measurement noise is inevitable in the real world. To address this limitation, this paper proposes a novel anti-noise method to reconstruct the wind field based on real-world noisy LIDAR measurements. To this end, we first address the gradient vanishing problem in PINN's automatic differentiation process, which prevents the Navier-Stokes (NS) equations from being fully leveraged in wind field reconstruction. This problem is addressed by proposing a residual-connected PINN (RC-PINN) framework. Its advantage in solving the gradient vanishing problem is mathematically proven. Subsequently, by incorporating the NS equations into the RC-PINN, an anti-noise wind field reconstruction method is established. The key feature of this method is that, by employing the RC-PINN, the NS equations are more effectively utilized in the training process, thus improving its anti-noise capability. The numerical results show that, at a wide range of noise levels, the proposed method can achieve a significant improvement in reconstruction performance. Especially, at the noise level of commercially available LIDAR, the reconstruction results are quite satisfactory. Moreover, the proposed method can converge faster.

1. Introduction

In recent years, fossil fuel shortage and environmental pollution have become increasingly prominent. The use of renewable energy is vital for addressing these problems. Since wind energy is one of the largest renewable energy resources, wind generation is growing rapidly worldwide. However, the random and intermittent characteristics of wind on temporal and spatial scales pose great challenges for the adequate use of wind energy, such as increasing the fatigue load of wind turbines, reducing the efficiency of wind energy utilization, and undermining electric quality [1–3]. Acquiring accurate and detailed

wind information, especially spatiotemporal wind speed and direction, is crucial for tackling these challenges. The detailed spatiotemporal wind speed and direction of the simulated wind field can be directly obtained through computational fluid dynamics (CFD) methods [4–6], which essentially compute the numerical solution of the Navier-Stokes (NS) equations. However, for the real-world wind field, obtaining its detailed spatiotemporal wind speed and direction is challenging.

Light detection and ranging (LIDAR) technology is effective for measuring the wind speed in real-world wind fields [7,8] and is used in many areas of the wind industry [9–13]. LIDAR can provide partial real-world wind information, i.e., the radial velocity component of wind speed

* Corresponding author.

E-mail address: koupeng@mail.xjtu.edu.cn (P. Kou).

at a limited number of sparse spatial locations in measurement directions [14,15]. Based on this fact, the LIDAR measurements can be regarded as low-dimensional mappings of the real-world spatiotemporal wind speed and direction. That is, the low-dimensional representation of the wind field. In this condition, if LIDAR measurements are combined with flow physics, which reflects the physical law of wind dynamics, the detailed spatiotemporal wind speed and direction in the real-world wind field are expected to be available. Considering the above fact, establishing a spatiotemporal wind field reconstruction method based on the combination of LIDAR measurements and flow physics is of great interest.

Currently, only several wind field reconstruction methods based on the combination of LIDAR measurements and flow physics have been proposed. According to the algorithm and physical model adopted, these methods can be classified into two categories. The first category of methods adopted traditional algorithms (i.e., non-deep learning algorithms) and simplified physical models derived from the NS equations. For example, in [16], the authors first embedded a low-order dynamic model derived from the incompressible NS equations into the unscented Kalman filter (UKF). Then, based on the wind speed measured by LIDAR, the above UKF was used to estimate the wind speed within the unmeasured areas. In [17], a reduced-order dynamic model was first built. Then, with the UKF using this model and LIDAR measurements, a spatiotemporal wind field reconstruction framework was proposed. In [18], the authors derived a simplified wind model by neglecting time dependency in NS equations. With this simplified model, some characteristics of the wind field were reconstructed by a model-fitting technique based on LIDAR measurements. In [19], based on the wind velocity dataset obtained from CFD simulation, proper orthogonal decomposition (POD) was used to extract low-dimensional basis vectors. These basis vectors, combined with sensor measurements, enabled wind field reconstruction. Although a different type of sensor was used in this study, the research findings are transferable to wind field reconstruction based on LIDAR measurements.

Since the non-deep learning algorithms are incapable of handling the complex NS equations, methods in the first category have to use the physical model simplified from the NS equations. This compromises the accuracy of wind field reconstruction. The physics-informed neural network (PINN) framework [20], an emerging deep learning technology, provides a solution to this issue.

Leveraging automatic differentiation (AD) [21], PINN has the capability to directly incorporate the physical model, which is represented in the form of partial differential equations (PDEs). This avoids the error introduced by model simplification. With this advantage, PINN has been widely employed across various fields and has demonstrated satisfactory performance [22–34]. In the field of fluid mechanics, as demonstrated in [28], PINN was used to approximate the Euler equations, which describe the physical laws governing high-speed aerodynamic flows. Additionally, in [29], two formulations of incompressible NS equations were incorporated into PINN for solving the forward and inverse problems of the incompressible NS equations. Within the realm of electromagnetics, as illustrated in [30], the Laplace equation and other electromagnetic equations in PDE form were incorporated into PINN for carrying out electromagnetic field simulations. In [31], PINN was used to solve inverse electromagnetic problems. Furthermore, PINN has also found extensive application in the field of energy. For instance, in the domain of electric energy, as demonstrated in [32], the swing equation, which describes the dynamic of power system, was incorporated into PINN for the simulation of power system frequency. In the field of geothermal energy, illustrated in [33], the PINN incorporated the NS equations coupled with heat equations and Darcy's law coupled with heat equations was used to simulate the porous medium fluid flow. In the field of tidal energy, as exemplified in [34], the authors incorporated the tidal equation, which describes the dynamic of tidal regimes, into PINN to simulate the velocity and water level fields. Moreover, PINN was also employed in the field of natural gas to analyze the transient flow in natural gas pipelines [35].

Considering the aforementioned advantages of PINN and its ideal performance in various fields, the second category of methods adopted

the physics-informed neural network (PINN) framework. In these methods, the NS equations were directly adopted as the physical model of the wind field by taking advantage of PINN. In [14], the PINN framework was composed of a deep multilayer perceptron (MLP) and 2D NS equations. The wind field reconstruction was achieved by minimizing the difference between the reconstructed values and LIDAR measurements, as well as the residues of the NS equations. The authors of [14] further encoded the 3D NS equations into the PINN framework. By doing so, LIDAR measurement-based 3D wind field reconstruction was achieved [36].

To summarize, the first category of methods is based on non-deep learning algorithms. Due to the complexity of the NS equations, this category of methods has to use the physical models simplified from the NS equations. In comparison, thanks to the PINN framework, the second category can directly adopt the NS equations as the physical model. This avoids the model reduction error. Therefore, the second category of methods, i.e., PINN-based methods, can achieve better reconstruction accuracy.

However, the existing PINN-based wind field reconstruction methods share a common limitation, i.e., they are only able to achieve promising reconstruction results in the absence of LIDAR measurement noise. Unfortunately, measurement noise is inevitable for real-world LIDAR [14,16,37], so the performance of the existing methods cannot be guaranteed in real-world applications.

This limitation is mainly caused by the so-called gradient vanishing problem, which arises during the PINN's forward propagation. Fig. 1 illustrates the gradient vanishing problem in the context of wind field reconstruction. It is noteworthy that the training set of the method within Fig. 1 is nondimensionalized by the characteristic length D , the characteristic velocity V_∞ , and the characteristic time D/V_∞ . Here, D and V_∞ represent, respectively, turbine rotor diameter and average freestream wind speed. As shown in this figure, in the PINN-based method, the forward propagation of PINN consists of two steps: (i) forward propagation of a deep MLP, and (ii) taking the derivatives of the MLP output w.r.t. its input using automatic differentiation. In fact, the second step (ii) is essentially the backpropagation from the deepest layer to the shallowest layer of the deep MLP. Therefore, according to the chain rule, the gradients of each fully connected and activation layer accumulate in this step, thus causing the gradient vanishing problem. The gradient vanishing further prevents PINN-based methods from leveraging the NS equations adequately. However, in the presence of LIDAR measurement noise, the wind dynamics described by the NS equations are vital for guaranteeing the reconstruction accuracy. Due to this fact, in the presence of inevitable real-world LIDAR measurement noise, the existing PINN-based methods are unable to achieve the ideal reconstruction accuracy.

ResNet has the potential to address this limitation. ResNet is one of the key technologies developed in the deep learning field in the past few years [38]. Its residual connection structure can effectively address the gradient vanishing problem [39,40]. Due to this advantage, ResNet has widespread applications in the computer vision field [41,42]. It also plays a significant role as a significant role in Transformer [43], a successful and well-known deep learning framework. Additionally, ResNet has proven to be effective in the field of wind power, with applications including wind power forecasting [44], wind speed forecasting [45], and fault diagnosis of wind turbine [46].

Inspired by the above analysis, we propose to incorporate the residual connection into the PINN to address the aforementioned gradient vanishing problem. Based on this, a residual connected PINN framework is proposed, which is referred to as RC-PINN. Through mathematical analysis, its advantage in solving the aforementioned gradient vanishing problem is proven. By encoding the NS equations into the RC-PINN, an anti-noise wind field reconstruction method is established. Through leveraging LIDAR measurements and the NS equations in a unified manner during training, the proposed method reconstructs the spatiotemporal wind speed and direction. The key feature of the proposed method is its anti-noise ability, which is derived from the fact that, by employing RC-PINN, it can more effectively utilize the NS equations in

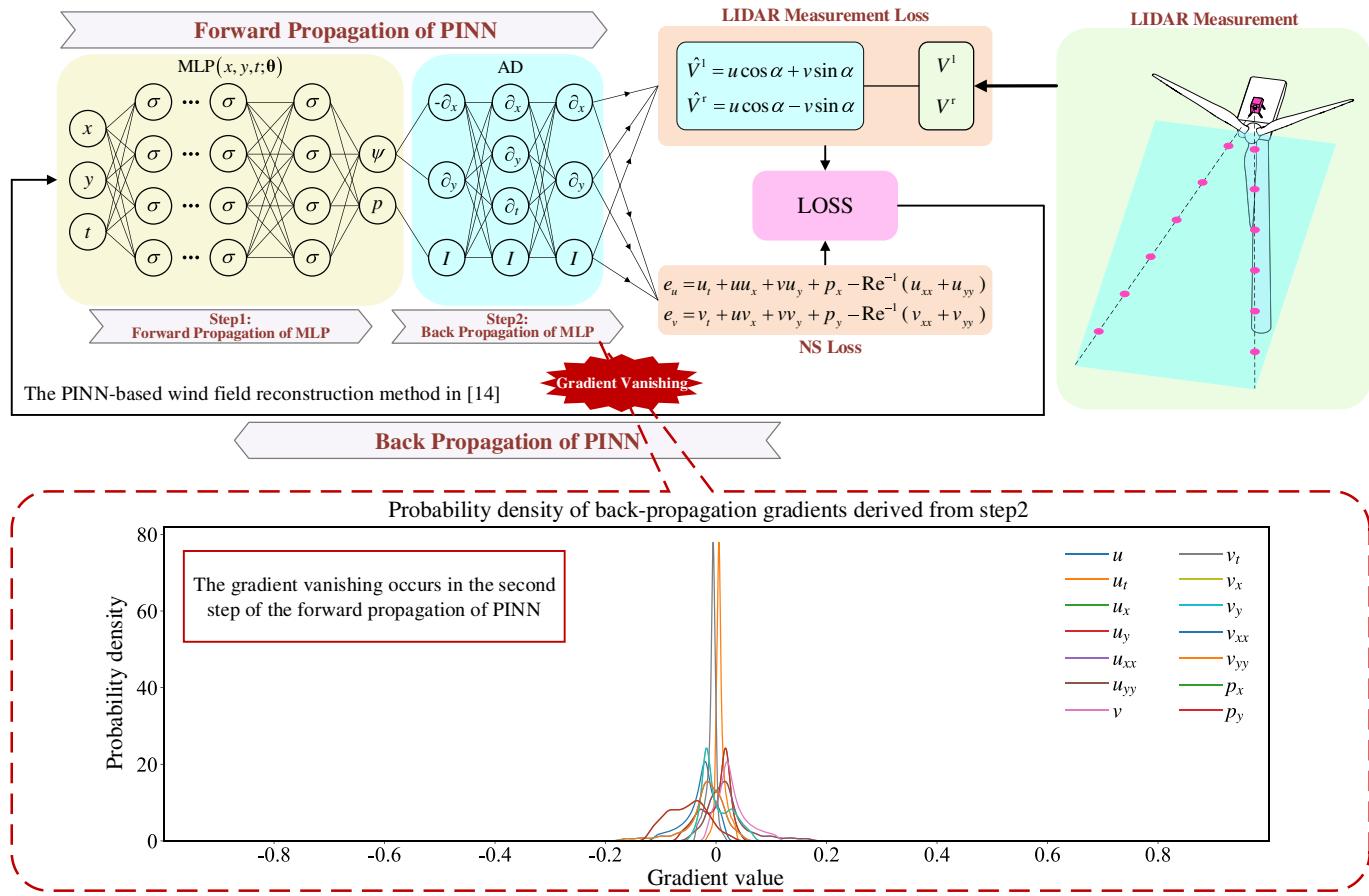


Fig. 1. Illustration of the gradient vanishing problem in the PINN-based wind field reconstruction method.

the training process. Consequently, the proposed method achieves high reconstruction performance at a wide range of noise levels. Moreover, the more effective utilization of the NS equations also leads to faster convergence in the training process.

To demonstrate the workflow of this research, a step-by-step diagram is presented in Fig. 2.

The novelty and contribution of this paper lie specifically in the following four aspects:

- By embedding the residual connection structure into the PINN framework, a residual-connected PINN framework (RC-PINN) is proposed. In RC-PINN, the gradient vanishing problem arising in the automatic differentiation process is alleviated. By this means, the NS equations can be leveraged more adequately in the wind field reconstruction.
- For the first time, the effectiveness of the residual connection structure in alleviating the gradient vanishing problem in PINN's automatic differentiation process is mathematically proven.
- An anti-noise wind field reconstruction method based on the proposed RC-PINN is presented. This method achieves a significant improvement in wind field reconstruction performance at a wide range of noise levels, together with faster convergence. Especially, at the noise level of commercially available LIDAR, the reconstruction results are very satisfactory.
- Based on a hybrid approach involving both theoretical analysis and numerical simulation, the anti-noise mechanism of the proposed method is demonstrated.

The remainder of this paper is organized as follows. Section 2 describes the wind field reconstruction problem considered in this paper. Section 3 establishes the residual-connected PINN framework and

proves its effectiveness in alleviating the gradient vanishing problem. Section 4 presents the proposed anti-noise wind field reconstruction method and demonstrates its anti-noise mechanism. Numerical results and corresponding analysis are given in Section 5. Finally, conclusions and future research directions are given in Section 6.

2. Wind field reconstruction considering LIDAR measurement noise

2.1. Wind field reconstruction based on LIDAR measurements

Compared to other wind measurement devices, LIDAR can provide wind information with higher spatial and temporal resolution [47]. Considering this fact, LIDAR is the most suitable device for wind farms to obtain detailed spatiotemporal wind information, which is crucial for wind farm control and wind energy assessment. Due to this fact, an increasing number of wind farms choose to install LIDAR on their wind turbines.

Fig. 3 illustrates the wind field measurement in front of the wind turbine conducted by nacelle LIDAR. The nacelle LIDAR on the top of the wind turbine emits two laser beams (which are represented by black dashed lines), and each laser beam contains multiple measurement locations (which are represented by pink dots). The LIDAR measurement (which is represented by the red arrow) captures the velocity component of wind speed (which is represented by the green arrow) in the direction of the laser beam at the measurement location. Therefore, LIDAR measurements can be regarded as low-dimensional mappings of the hub-height 2D wind field in front of the wind turbine (which is represented by the light blue area). Wind field reconstruction is to, based on the wind dynamic model, map LIDAR measurements back to the corresponding spatiotemporal wind speed and direction of the whole wind field.

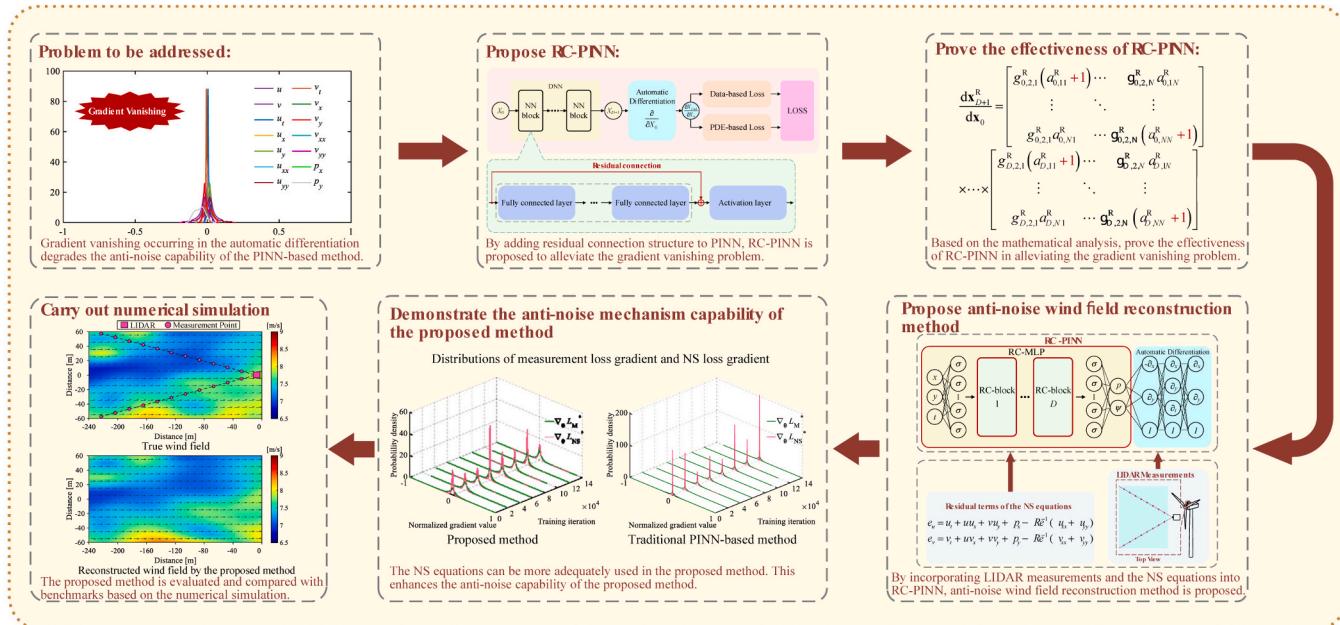


Fig. 2. Step-by-step diagram of the proposed research.

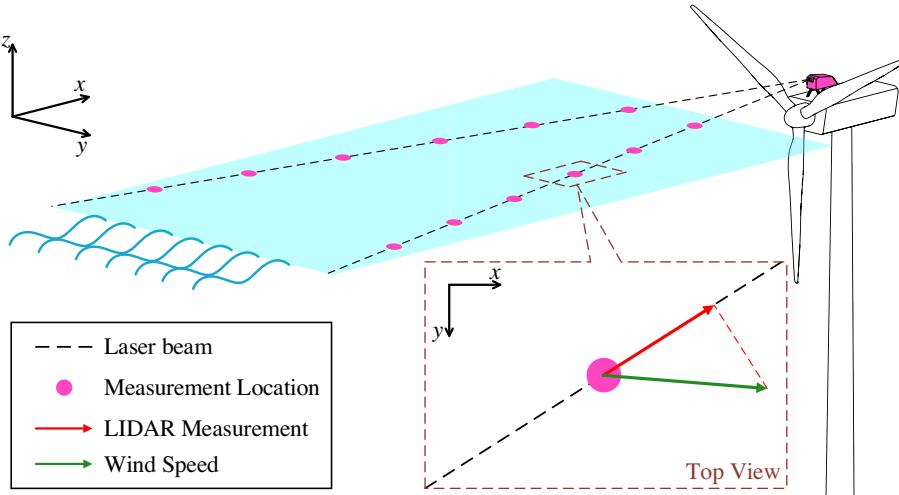


Fig. 3. The LIDAR measurements of the wind field ahead of a wind turbine.

Since LIDAR measurements capture the low-dimensional representation of the wind field, they only provide fundamental information for wind field reconstruction. In this sense, an accompanying method is necessary for the wind field reconstruction. Among many potential methods, the PINN-based method not only adopts the NS equations as the wind dynamic model, but also incorporates the NS equations and LIDAR measurements in a unified manner. Therefore, it is a good choice for wind field reconstruction. In the PINN-based method, by using the nacelle LIDAR, a set of LIDAR measurements during a time period of T is collected firstly. Subsequently, based on the collected LIDAR measurements and physical information (i.e., the NS equations), PINN training is conducted. Finally, using the trained PINN, the wind speed and direction of the whole wind field during the period T are reconstructed.

2.2. LIDAR measurement noise

Theoretically, the PINN-based method is ideal for reconstructing the spatiotemporal wind speed and direction in a real-world wind field. However, in practical applications, the reconstruction accuracy of the PINN-based method is affected by the LIDAR measurement noise. Specifically, during the training process of the PINN-based method, a significant emphasis is placed on LIDAR measurements, while prior physical knowledge (i.e., the NS equations), is comparatively given less attention. In this condition, the presence of measurement noise will mislead the training process, thereby making the training results deviate from the NS equations. Consequently, the PINN reconstruction performance significantly degrades in the presence of measurement noise.

In fact, measurement noise is inevitable in real applications [16]. This has been reported in the product guides of many LIDAR manufacturers, such as VAISALA [48], ZXLDars [49], and MOVELASER [50].

Therefore, an anti-noise wind field reconstruction is imperative for real-world applications.

3. Residual-connected PINN framework

The deficiency of the traditional PINN-based method in terms of anti-noise capability stems from the gradient vanishing problem. Specifically, as stated in [Section 1](#), the second step of PINN forward propagation is the automatic differentiation process, which is essentially a process of taking the partial derivative of the MLP output w.r.t. the MLP input. According to the chain rule, the gradient vanishing will arise in this process if the MLP is deep [[51,52](#)]. Unfortunately, in the wind field reconstruction applications, the MLP in PINN needs to be deep to ensure its approximation capability [[14](#)]. As a result, in wind field reconstruction, there is severe gradient vanishing in the second step of PINN's forward propagation, i.e., the automatic differentiation process. This makes the traditional PINN-based method unable to fully leverage the NS equations in training, thus making the reconstruction accuracy dramatically degrade in the presence of LIDAR measurement noise. The specific mechanism by which the gradient vanishing problem reduces the anti-noise capability of the traditional PINN-based method is detailed in [subsection 4.3](#). To overcome this issue, in this section, a new PINN framework that embeds the residual connection is proposed. The effectiveness of residual connection on alleviating the gradient vanishing has been proven in many applications [[39](#)]. We refer to the proposed residual-connected PINN framework as RC-PINN.

3.1. The structure of RC-PINN framework

[Fig. 4](#) illustrates the relation and difference between the conventional PINN and the proposed RC-PINN. From this figure, one can see that both PINN and RC-PINN have the same generic overall structure. The major difference between them is the deep neural network (DNN) they use. Specifically, the DNN used in the conventional PINN consists of NN blocks shown in the part (A) of [Fig. 4](#). Therefore, the DNN in the conventional PINN is essentially an MLP, so we refer to this NN block as

MLP-block. As mentioned in [Section 1](#), as for the wind field reconstruction, the deep MLP in PINN may lead to gradient vanishing problem. In comparison, the DNN used in the proposed RC-PINN consists of the NN blocks embedded with residual connection, as in the part (C) of [Fig. 4](#). We refer to this NN block as RC-block. Correspondingly, the DNN composed of RC-blocks is referred to as RC-MLP. As demonstrated in [[39](#)], the residual connection can work as a “direct” path for back-propagation. Taking advantage of this fact, by incorporating the residual connection, the fully connected layer, which increases the risk of gradient vanishing, is no longer the only path for the gradient back-propagation during the automatic differentiation process. By this means, RC-PINN alleviates the gradient vanishing problem. This point will be further proven in next subsection.

3.2. Alleviate the gradient vanishing problem by incorporating the residual connection into the RC-PINN

As described in [Section 3.1](#), due to the incorporation of the residual connection, RC-PINN alleviates the gradient vanishing problem that arises in the automatic differentiation process. In this subsection, we present the following corollary to support this claim.

Corollary 1. If weight matrices are initialized by the commonly used Xavier initialization [[53](#)] and the bias vectors are initialized to zero, $D^R = D^M = D$, $N^R = N^M = N$, then we have $|\partial x_{D+1,i}^R / \partial x_{0,j}^R| > |\partial x_{D+1,i}^M / \partial x_{0,j}^M|$, where $i, j \in [1, N]$. Here, D^R is the number of RC-blocks in the RC-PINN. D^M is the number of MLP-blocks in the PINN. N^R is the dimension of the input and output vectors of RC-blocks. N^M is the dimension of the input and output vectors of MLP-blocks. $x_{0,j}^R$ and x_{D+1}^R are, respectively, the elements of RC-PINN's input vector x_0^R and output vector x_{D+1}^R . $x_{0,j}^M$ and x_{D+1}^M are, respectively, the elements of PINN's input vector x_0^M and output vector x_{D+1}^M . $\partial x_{D+1,i}^R / \partial x_{0,j}^R$ and $\partial x_{D+1,i}^M / \partial x_{0,j}^M$ are, respectively, the outputs of RC-PINN's automatic differentiation process and PINN's automatic differentiation process.

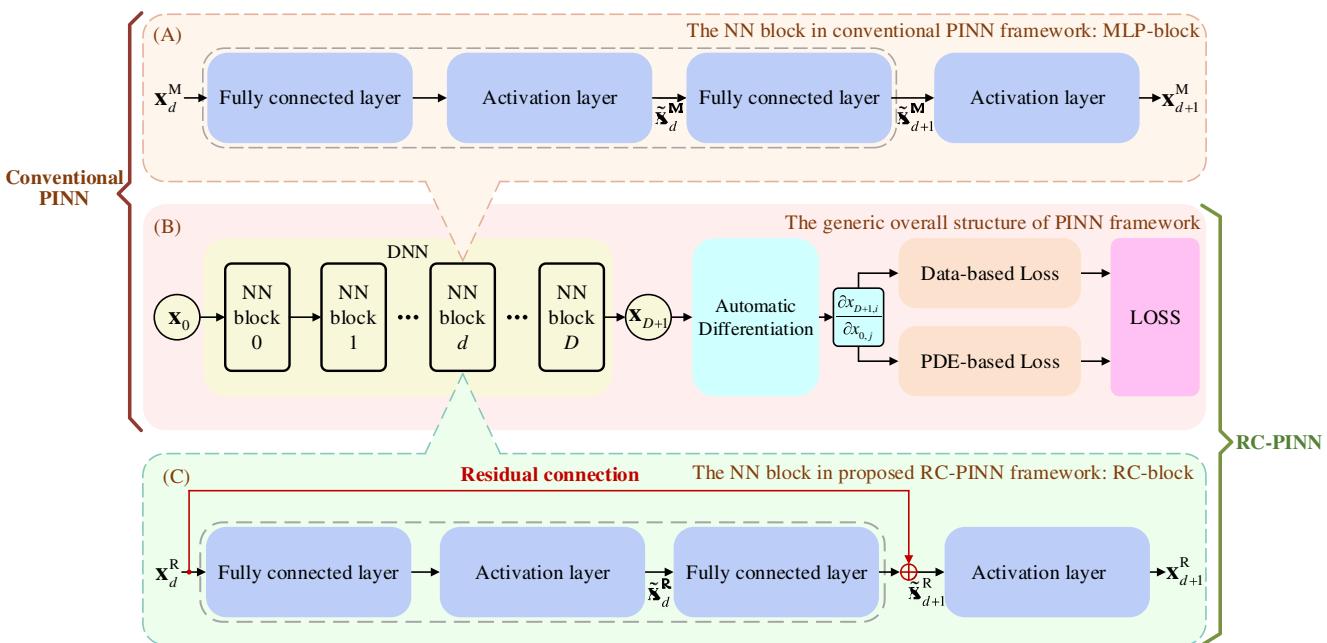


Fig. 4. Block diagram of the conventional PINN framework and RC-PINN framework. (A) The NN block in conventional PINN framework. (B) The generic overall structure of PINN framework. (C) The NN block in proposed RC-PINN framework.

Proof. The d -th MLP-block can be expressed as

$$\mathbf{x}_{d+1}^M = \tanh \left[\mathbf{W}_{d,2}^M \tanh \left(\mathbf{W}_{d,1}^M \mathbf{x}_d^M + \mathbf{b}_{d,1}^M \right) + \mathbf{b}_{d,2}^M \right]. \quad (1)$$

Here, $\mathbf{W}_{d,1}^M$ and $\mathbf{W}_{d,2}^M$ are, respectively, $N \times N$ weight matrixes of the first and second fully connected layers in the d -th MLP-block. $\mathbf{b}_{d,1}^M$ and $\mathbf{b}_{d,2}^M$ are, respectively, $N \times 1$ bias vectors of the first and second fully connected layers in the d -th MLP-block. $\tanh(\cdot)$ is hyperbolic tangent function. Similarly, the d -th RC-block can be expressed as

$$\mathbf{x}_{d+1}^R = \tanh \left[\mathbf{W}_{d,2}^R \tanh \left(\mathbf{W}_{d,1}^R \mathbf{x}_d^R + \mathbf{b}_{d,1}^R \right) + \mathbf{b}_{d,2}^R + \mathbf{x}_d^R \right]. \quad (2)$$

Here, $\mathbf{W}_{d,1}^R$ and $\mathbf{W}_{d,2}^R$ are, respectively, $N \times N$ weight matrixes of the first and second fully connected layers in the d -th RC-block. $\mathbf{b}_{d,1}^R$ and $\mathbf{b}_{d,2}^R$ are, respectively, the $N \times 1$ bias vectors of the first and second fully connected layers in the d -th RC-block.

According to (1), based on the chain rule, $d\mathbf{x}_{d+1}^M/d\mathbf{x}_d^M$ can be expressed as

$$\frac{d\mathbf{x}_{d+1}^M}{d\mathbf{x}_d^M} = \left[\left(\mathbf{W}_{d,1}^M \right)^T \mathbf{G}_{d,1}^M \left(\mathbf{W}_{d,2}^M \right)^T \right] \mathbf{G}_{d,2}^M. \quad (3)$$

Here, diagonal matrixes $\mathbf{G}_{d,1}^M = \text{diag}[g_{d,1,1}^M, g_{d,1,2}^M, \dots, g_{d,1,N}^M]$ and $\mathbf{G}_{d,2}^M = \text{diag}[g_{d,2,1}^M, g_{d,2,2}^M, \dots, g_{d,2,N}^M]$ are, respectively, the gradient matrixes of the first and second activation layer in the d -th MLP-block.

Similarly, according to (2), $d\mathbf{x}_{d+1}^R/d\mathbf{x}_d^R$ can be expressed as

$$\frac{d\mathbf{x}_{d+1}^R}{d\mathbf{x}_d^R} = \left[\left(\mathbf{W}_{d,1}^R \right)^T \mathbf{G}_{d,1}^R \left(\mathbf{W}_{d,2}^R \right)^T + \mathbf{I} \right] \mathbf{G}_{d,2}^R. \quad (4)$$

Here, diagonal matrixes $\mathbf{G}_{d,1}^R = \text{diag}[g_{d,1,1}^R, g_{d,1,2}^R, \dots, g_{d,1,N}^R]$ and $\mathbf{G}_{d,2}^R = \text{diag}[g_{d,2,1}^R, g_{d,2,2}^R, \dots, g_{d,2,N}^R]$ are, respectively, the gradient matrixes of the first and second activation layer in the d -th RC-block.

Since $\tanh'(y) \in [0, 1]$, we have $g_{d,i,j}^M \in [0, 1]$ and $g_{d,i,j}^R \in [0, 1]$, where $i = 1, 2$ and $j = 1, 2, \dots, N$.

As for weight matrices, Xavier initialization is commonly used in many applications to make the training process more efficient. In Xavier initialization, the element of the weight matrix \mathbf{W} , i.e., w_{ij} , is generated from the normal distribution $N(0, \sigma^2)$. The standard deviation is

$$\sigma = \sqrt{\frac{2}{N_{in} + N_{out}}}. \quad (5)$$

Here, N_{in} and N_{out} are, respectively, the numbers of input features and output features of a fully connected layer. In context of this proof, $N_{in} = N_{out} = N$. Since N is generally large to guarantee approximation capability, the value of σ is significantly smaller than 1, thereby making the initial magnitudes of $\{w_{ij} | i \in [1, N], j \in [1, N]\}$ smaller than 1.

As for bias vectors, generally they are initialized to zero.

Meanwhile in the training process, the update on the training parameters can be expressed as

$$\theta' = \theta - \eta \nabla \theta. \quad (6)$$

Here, θ' represents the updated training parameter and η represents learning rate. Generally, in the training process, the absolute value of $\nabla \theta$ is smaller than 1 and it decrease during training. Meanwhile, the magnitude of η is generally as small as 10^{-3} to make the training process stable. As a result, the magnitudes of training parameters will not change dramatically during training.

From the analysis of the initialization and update of training parameters, one can see that, during training process, the magnitudes of training parameters are generally significantly smaller than 1. In context of this proof, the magnitudes of all elements in weight matrices, i.e. $\mathbf{W}_{d,1}^M$, $\mathbf{W}_{d,2}^M$, $\mathbf{W}_{d,1}^R$, and $\mathbf{W}_{d,2}^R$, and bias vectors, i.e. $\mathbf{b}_{d,1}^M$, $\mathbf{b}_{d,2}^M$, $\mathbf{b}_{d,1}^R$, and $\mathbf{b}_{d,2}^R$, are significantly smaller than 1.

Let $\mathbf{A}_d^M = \left(\mathbf{W}_{d,1}^M \right)^T \mathbf{G}_{d,1}^M \left(\mathbf{W}_{d,2}^M \right)^T$. From analysis of the magnitudes of the elements in $\mathbf{W}_{d,1}^M$, $\mathbf{W}_{d,2}^M$, and $\mathbf{G}_{d,1}^M$, we have $|a_{d,ij}^M| \ll 1$, where $a_{d,ij}^M$ is the element of \mathbf{A}_d^M .

Similarly, let $\mathbf{A}_d^R = \left(\mathbf{W}_{d,1}^R \right)^T \mathbf{G}_{d,1}^R \left(\mathbf{W}_{d,2}^R \right)^T$. From analysis of the magnitudes of the elements in $\mathbf{W}_{d,1}^R$, $\mathbf{W}_{d,2}^R$, and $\mathbf{G}_{d,1}^R$, we have $|a_{d,ij}^R| \ll 1$, where $a_{d,ij}^R$ is the element of \mathbf{A}_d^R .

Then, (3) and (4) are reformulated, that is

$$\frac{d\mathbf{x}_{d+1}^M}{d\mathbf{x}_d^M} = \begin{bmatrix} a_{d,11}^M & a_{d,12}^M & \cdots & a_{d,1N}^M \\ a_{d,21}^M & a_{d,22}^M & \cdots & a_{d,2N}^M \\ \vdots & \vdots & \ddots & \vdots \\ a_{d,N1}^M & a_{d,N2}^M & \cdots & a_{d,NN}^M \end{bmatrix} \mathbf{G}_{d,2}^M, \quad (7)$$

$$\frac{d\mathbf{x}_{d+1}^R}{d\mathbf{x}_d^R} = \begin{bmatrix} a_{d,11}^R + 1 & a_{d,12}^R & \cdots & a_{d,1N}^R \\ a_{d,21}^R & a_{d,22}^R + 1 & \cdots & a_{d,2N}^R \\ \vdots & \vdots & \ddots & \vdots \\ a_{d,N1}^R & a_{d,N2}^R & \cdots & a_{d,NN}^R + 1 \end{bmatrix} \mathbf{G}_{d,2}^R. \quad (8)$$

Then, analysis of the magnitudes of the elements in $\mathbf{G}_{d,2}^M$ and $\mathbf{G}_{d,2}^R$ is carried out. The inputs of the second activation layer in the d -th MLP-block (i.e., $\tilde{\mathbf{x}}_{d+1}^M$) and RC-block (i.e., $\tilde{\mathbf{x}}_{d+1}^R$) can be expressed as

$$\begin{aligned} \tilde{\mathbf{x}}_{d+1}^M &= \mathbf{W}_{d,2}^M \tilde{\mathbf{x}}_d^M + \mathbf{b}_{d,2}^M, \\ \tilde{\mathbf{x}}_{d+1}^R &= \mathbf{W}_{d,2}^R \tilde{\mathbf{x}}_d^R + \mathbf{b}_{d,2}^R + \mathbf{x}_d^R. \end{aligned} \quad (9)$$

Here, $\tilde{\mathbf{x}}_d^M$ and $\tilde{\mathbf{x}}_d^R$ are, respectively, the output vectors of the first activation layers in the d -th MLP-block and RC-block. All elements in $\tilde{\mathbf{x}}_d^M$ and $\tilde{\mathbf{x}}_d^R$ belong to $[-1, 1]$. From (9), the magnitude of $\tilde{x}_{d+1,i}^M$ approaches 0 and the maximum magnitude of $\tilde{x}_{d+1,i}^R$ approaches 1. Thus, the difference between $\tanh'(\tilde{x}_{d+1,i}^M)$ and $\tanh'(\tilde{x}_{d+1,i}^R)$ is within one order of magnitude. As a result, it is reasonable to ignore the effects of $\mathbf{G}_{d,2}^M$ and $\mathbf{G}_{d,2}^R$.

Comparing (7) and (8), one can see that the magnitudes of diagonal elements in the gradient matrix of the d -th RC-block are larger than those of the d -th MLP-block.

Based on the chain rule, in RC-PINN, all outputs of the automatic differentiation process, which constitute a gradient matrix, can be expressed as

$$\begin{aligned}
& \frac{dx_{D+1}^R}{dx_0} = \left[\begin{array}{ccc} \frac{\partial x_{D+1,1}^R}{\partial x_{0,1}} & \frac{\partial x_{D+1,2}^R}{\partial x_{0,1}} & \dots & \frac{\partial x_{D+1,N}^R}{\partial x_{0,1}} \\ \frac{\partial x_{D+1,1}^R}{\partial x_{0,2}} & \frac{\partial x_{D+1,2}^R}{\partial x_{0,2}} & \dots & \frac{\partial x_{D+1,N}^R}{\partial x_{0,2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial x_{D+1,1}^R}{\partial x_{0,N}} & \frac{\partial x_{D+1,2}^R}{\partial x_{0,N}} & \dots & \frac{\partial x_{D+1,N}^R}{\partial x_{0,N}} \end{array} \right] = \prod_{k=0}^D \frac{dx_{d+1}^R}{dx_d} \\
& = \left[\begin{array}{cccc} g_{0,2,1}^R(a_{0,11}^R + 1) & g_{0,2,2}^R a_{0,12}^R & \dots & g_{0,2,N}^R a_{0,1N}^R \\ g_{0,2,1}^R a_{0,21}^R & g_{0,2,2}^R(a_{0,22}^R + 1) & \dots & g_{0,2,N}^R a_{0,2N}^R \\ \vdots & \vdots & \ddots & \vdots \\ g_{0,2,1}^R a_{0,N1}^R & g_{0,2,2}^R a_{0,N2}^R & \dots & g_{0,2,N}^R(a_{0,NN}^R + 1) \end{array} \right] \quad (10) \\
& \times \left[\begin{array}{cccc} g_{1,2,1}^R(a_{1,11}^R + 1) & g_{1,2,2}^R a_{1,12}^R & \dots & g_{1,2,N}^R a_{1,1N}^R \\ g_{1,2,1}^R a_{1,21}^R & g_{1,2,2}^R(a_{1,22}^R + 1) & \dots & g_{1,2,N}^R a_{1,2N}^R \\ \vdots & \vdots & \ddots & \vdots \\ g_{1,2,1}^R a_{1,N1}^R & g_{1,2,2}^R a_{1,N2}^R & \dots & g_{1,2,N}^R(a_{1,NN}^R + 1) \end{array} \right] \\
& \times \dots \\
& \times \left[\begin{array}{cccc} g_{D,2,1}^R(a_{D,11}^R + 1) & g_{D,2,2}^R a_{D,12}^R & \dots & g_{D,2,N}^R a_{D,1N}^R \\ g_{D,2,1}^R a_{D,21}^R & g_{D,2,2}^R(a_{D,22}^R + 1) & \dots & g_{D,2,N}^R a_{D,2N}^R \\ \vdots & \vdots & \ddots & \vdots \\ g_{D,2,1}^R a_{D,N1}^R & g_{D,2,2}^R a_{D,N2}^R & \dots & g_{D,2,N}^R(a_{D,NN}^R + 1) \end{array} \right].
\end{aligned}$$

From (10), we finally have $|\partial x_{D+1,i}^R / \partial x_{0,j}^R| > |\partial x_{D+1,i}^M / \partial x_{0,j}^M|$, where $i, j \in [1, N]$.

Notice that, when replacing hyperbolic tangent function with sigmoid function or Xavier initialization with Kaiming initialization [54], Corollary 1 still holds true. The proof methodology is similar. In addition, this corollary only considers the first-order partial derivative outputs of automatic differentiation process, so it only supports that the RC-PINN alleviates the gradient vanishing problem arising in the first-order partial derivative outputs of the automatic differentiation process. Nonetheless, the simulation results in Section 4 indicate that RC-PINN is effective in alleviating the gradient vanishing problem arising in all of the first, second, and third-order partial derivative outputs of the automatic differentiation process.

4. Anti-noise wind field reconstruction using RC-PINN

Based on the proposed RC-PINN, we propose an anti-noise wind field reconstruction method, which incorporates the real-world noisy LIDAR measurements and the NS equations in a unified manner. The proposed method is depicted in Fig. 5. As shown in this figure, the proposed method can be divided into two processes: the training process (illustrated in the part (A) of Fig. 5) and the reconstruction process (illustrated in the part (B) of Fig. 5). During the training process, RC-PINN is trained by optimizing to minimize NS loss and measurement loss. Subsequently, the trained RC-PINN is employed in the reconstruction process. Specifically, given a spatiotemporal coordinate, the trained RC-PINN outputs the wind velocity at this coordinate.

By using RC-PINN, the gradient vanishing problem can be significantly alleviated, thereby leveraging the NS equations more adequately.

In the wind field reconstruction, the NS equations are essential for guaranteeing the reconstruction accuracy in the presence of LIDAR measurement noise, so the proposed method has a stronger anti-noise ability. The details of this method are described in the rest of this section.

4.1. Application of RC-PINN in the anti-noise wind field reconstruction

In the context of the proposed method, the structures of the RC-PINN in the training and reconstruction processes are, respectively, illustrated in the two black boxes of Fig. 5. One can see that, there are two core components of RC-PINN, i.e., RC-MLP and the automatic differentiation process.

Firstly, we discuss the RC-MLP, which is the NN part of the RC-PINN and contains all its training parameters. The detailed structure of RC-MLP is represented by the yellow box in the part (A) of Fig. 5. From this figure, one can see that a RC-MLP is mainly composed of multiple RC-blocks, which have been introduced in Section 3. In the proposed method, RC-MLP is used to approximate the mapping from the spatiotemporal coordinate to the corresponding wind field variables. Specifically, the input of RC-MLP is the spatiotemporal coordinate, $\mathbf{x} = [x, y, t]$, where x and y represent the two-dimensional spatial coordinates of the wind field described in section 2, and t represents the time coordinate. The output of RC-MLP is $\mathbf{y} = [\psi, p]$, where ψ is the latent function [20] and p is the pressure. This way, when we input a spatiotemporal coordinate $[x_i, y_i, t_i]$ to RC-MLP, the RC-MLP returns the latent function and pressure at the corresponding coordinate $[\psi(x_i, y_i, t_i), p(x_i, y_i, t_i)]$. It is worth noting that both ψ and p are only the intermediate variables in the RC-PINN, so we do not need their true values during training.

We further present the expression of RC-MLP in the context of the proposed anti-noise wind field reconstruction method. The main component of RC-MLP is RC-block. Each RC-block can be expressed as

$$\mathbf{h}_{d+1} = F_{rc}(\mathbf{x}_d, \mathcal{W}_d) = \sigma(\mathbf{h}_d + \mathbf{W}_{d,2}\sigma(\mathbf{W}_{d,1}\mathbf{h}_d + \mathbf{b}_{d,1}) + \mathbf{b}_{d,2}), \quad (11)$$

$$\mathcal{W}_d = \{\mathbf{W}_{d,1}, \mathbf{b}_{d,1}, \mathbf{W}_{d,2}, \mathbf{b}_{d,2}\}. \quad (12)$$

Here, F_{rc} represents the calculation performed by RC-block. \mathbf{h}_d and \mathbf{h}_{d+1} represent the input and output of the d -th RC-block, respectively. σ represents the activation function, which is a hyperbolic tangent function in this work. \mathcal{W}_d represents a set of training parameters in the d -th RC-block. $\mathbf{W}_{d,1}$ and $\mathbf{b}_{d,1}$ represent, respectively, the weight and bias of the first fully connected layer in the d -th RC-block. $\mathbf{W}_{d,2}$ and $\mathbf{b}_{d,2}$ represent, respectively, the weight and bias of the second fully connected layer in the d -th RC-block.

As in Fig. 5, slightly different from the generic RC-MLP introduced in Section 3, in the context of proposed method, in addition to RC-blocks, RC-MLP also contains an input layer and an output layer. The input layer used for feature expansion is composed of a fully connected layer and an activation layer, while the output layer used for feature reduction is composed of a fully connected layer. So, the RC-MLP in the proposed method can be ultimately expressed as

$$\mathbf{h}_1 = \sigma(\mathbf{W}_1 \mathbf{x} + \mathbf{b}_1), \quad (13)$$

$$\mathbf{h}_{d+1} = F_{rc}(\mathbf{h}_d, \mathcal{W}_d), 1 \leq d \leq D, \quad (14)$$

$$\mathbf{y} = \mathbf{W}_{D+1}\mathbf{h}_{D+1} + \mathbf{b}_{D+1}, \quad (15)$$

where \mathbf{W}_1 and \mathbf{b}_1 represent the weight and bias of the RC-MLP's input layer, respectively. \mathbf{W}_{D+1} and \mathbf{b}_{D+1} represent the weight and bias of the RC-MLP's output layer, respectively. The shapes of \mathbf{W}_1 , $\{\mathbf{W}_{d,1}, \mathbf{W}_{d,2} | 1 \leq d \leq D\}$, and \mathbf{W}_{D+1} are $[3, N_h]$, $[N_h, N_h]$, and $[N_h, 2]$, respectively, where N_h is the number of neurons in the hidden layers. The shapes of bias vector \mathbf{b}_1 , $\{\mathbf{b}_{d,1}, \mathbf{b}_{d,2} | 1 \leq d \leq D\}$, and \mathbf{b}_{D+1} are $[1, N_h]$, $[1, N_h]$, and $[1, 2]$, respectively.

Secondly, we discuss the automatic differentiation process. Once the

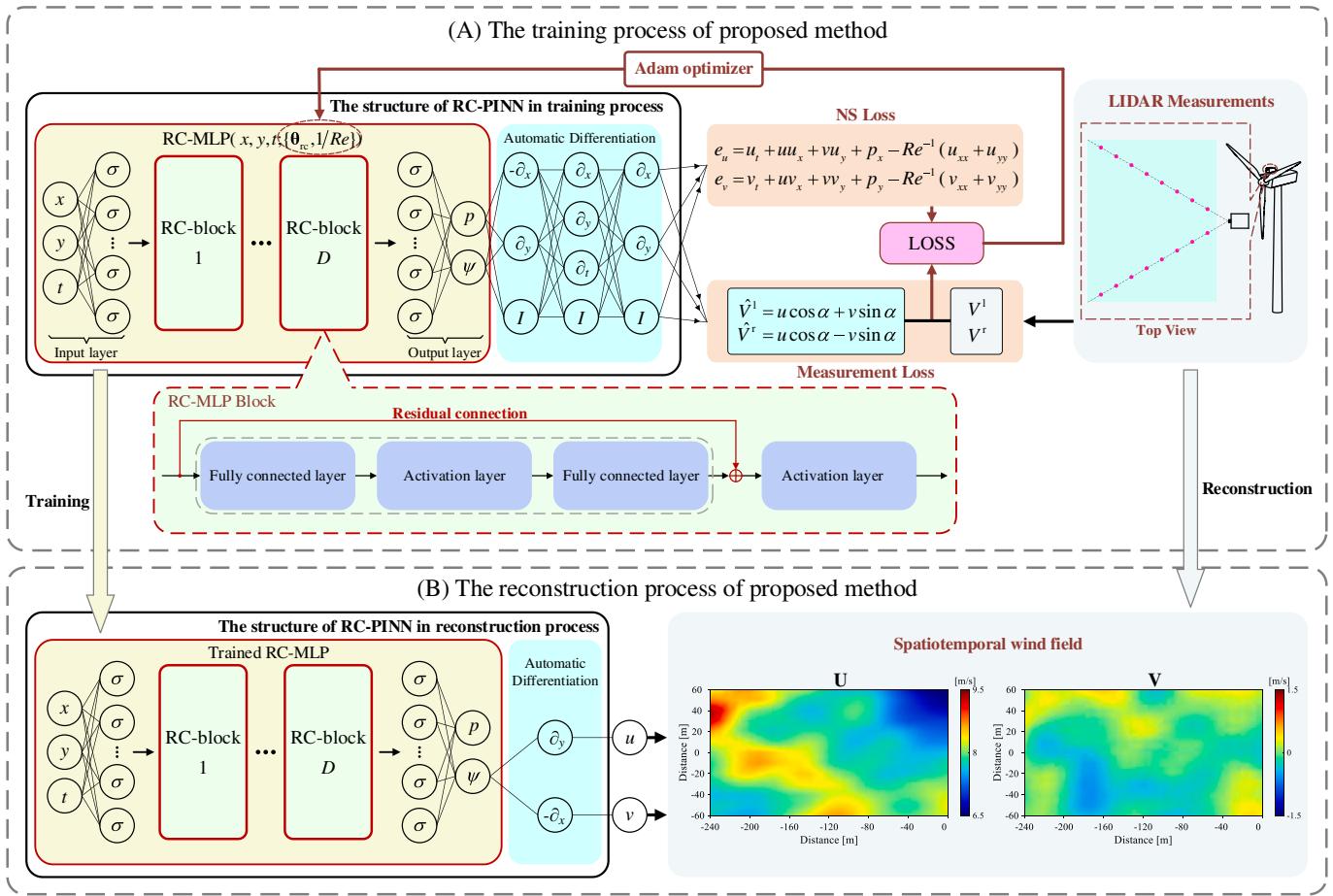


Fig. 5. The proposed anti-noise wind field reconstruction method based on RC-PINN.

formulation of RC-MLP is obtained, we can apply automatic differentiation on the output of RC-MLP. By doing so, the NS equations and LIDAR measurements can be incorporated into the training process. Next, we show the details regarding this incorporation.

Incorporation of LIDAR measurements: Based on the output of RC-MLP, we find exact derivatives of the latent function using automatic differentiation, as follows

$$u = \frac{\partial \psi}{\partial y}, v = -\frac{\partial \psi}{\partial x}. \quad (16)$$

Here, u and v represent the wind velocity components in the x direction and the y direction, respectively. As for u and v , the following continuity equation for incompressible fluids is satisfied automatically

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \quad (17)$$

Since the LIDAR measurement is only the radial component of wind speed at the measurement point, the following mappings exist between the wind velocity vector and the LIDAR measurement

$$V^l = u \cos \alpha + v \sin \alpha, \quad (18)$$

$$V^r = u \cos \alpha - v \sin \alpha,$$

where V^l and V^r represent, respectively, the LIDAR measurements in the left and right laser beams. α represents the half-angle between the LIDAR beams. Based on these mappings, RC-PINN predictions for the measurement in the left and right beams, \hat{V}^l and \hat{V}^r , can be obtained. For brevity, the LIDAR measurement prediction process can be shortly denoted as

$$\begin{aligned} \hat{V}^l &= F_M^l(\mathbf{x}^l; \theta_{rc}), \\ \hat{V}^r &= F_M^r(\mathbf{x}^r; \theta_{rc}), \end{aligned} \quad (19)$$

where function F_M^l and function F_M^r denote, respectively, the prediction processes for the left and right beams. θ_{rc} represents all weights and biases in the RC-MLP. \mathbf{x}^l and \mathbf{x}^r represent the spatiotemporal coordinates of the measurement points in the left and right LIDAR measurement beams, respectively.

Using (19), the LIDAR measurements can be incorporated into the loss function of the proposed method, which will be detailed in the next subsection.

Incorporation of NS equations: As stated in Section 2, the wind field information provided by real-world LIDAR measurements is limited and noisy. In this case, incorporating the NS equations is essential for wind field reconstruction. The essentiality comes from two aspects: (i) the NS equations provide the wind dynamics of the whole wind field; (ii) During the training process, the noisy LIDAR measurements may not conform to the fluid physics law and mislead the training process, while the NS equations can be regarded as constraints that correct the training process, thus guaranteeing the reconstruction accuracy.

Similar to the approach of incorporating LIDAR measurements, we incorporate the NS equations through the loss function. Specifically, the wind physics in the wind field can be expressed by the following 2D NS equations

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{1}{Re} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \quad (20)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = - \frac{\partial p}{\partial y} + \frac{1}{Re} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right). \quad (21)$$

To incorporate (20)–(21) into the loss function, further formula transformations are performed as follows:

$$e_u = \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + \frac{\partial p}{\partial x} - \frac{1}{Re} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \quad (22)$$

$$e_v = \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + \frac{\partial p}{\partial y} - \frac{1}{Re} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right). \quad (23)$$

Here, e_u and e_v are residue terms of the NS equations, which can be directly incorporated into the loss function. Re is Reynolds number. As for the real-world wind field within a wind farm, the Reynolds number is unknown, so we treat $1/Re$ as a training parameter (i.e., the set of training parameters in RC-PINN is $\{\theta_{rc}, 1/Re\}$) and infer it through the training process. To compute the above two residue terms, automatic differentiation is applied to derive the first, second, and third order partial derivatives of RC-MLP's output w.r.t. its input.

For brevity, the computation process of NS residue terms can be shortly denoted as

$$e_u = F_{NS}^u(\mathbf{x}^{ns}; \{\theta_{rc}, 1/Re\}), \quad (24)$$

$$e_v = F_{NS}^v(\mathbf{x}^{ns}; \{\theta_{rc}, 1/Re\}), \quad (25)$$

where function F_{NS}^u and function F_{NS}^v are, respectively, the prediction processes for NS residue terms. \mathbf{x}^{ns} represents the spatiotemporal coordinates of any location in the wind field.

Compared to training, the automatic differentiation process during reconstruction is simpler. It can be directly expressed as (16). The output of this process is the wind velocity vector $[u, v]$.

4.2. The training and reconstruction process

The training process of the proposed anti-noise wind field reconstruction method is shown in the part (A) of Fig. 5. One can see that this training is essentially to infer the parameters of RC-PINN by minimizing the loss function. Specifically, the loss function consists of two parts. The first part is referred to as *measurement loss*, which is expressed as

$$L_M = \frac{1}{N_l} \sum_{i=1}^{N_l} (F_M^l(\mathbf{x}_i^l; \theta_{rc}) - V_i^l)^2 + \frac{1}{N_r} \sum_{i=1}^{N_r} (F_M^r(\mathbf{x}_i^r; \theta_{rc}) - V_i^r)^2, \quad (26)$$

where $\{\mathbf{x}_i^l, V_i^l\} | 1 \leq i \leq N_l$ is a batch of randomly-sampled LIDAR measurement data in the left beam, and $\{\mathbf{x}_i^r, V_i^r\} | 1 \leq i \leq N_r$ is a batch of randomly-sampled LIDAR measurement data in the right beam. The measurement loss reflects the deviation between the true LIDAR measurements and RC-PINN predictions.

The second part of loss function represents the physical loss. It is referred to as *NS loss*, and it can be expressed as

$$L_{NS} = \frac{1}{N_{ns}} \sum_{i=1}^{N_{ns}} F_{NS}^u(\mathbf{x}_i^{ns}; \{\theta_{rc}, 1/Re\})^2 + \frac{1}{N_{ns}} \sum_{i=1}^{N_{ns}} F_{NS}^v(\mathbf{x}_i^{ns}; \{\theta_{rc}, 1/Re\})^2, \quad (27)$$

where $\{\mathbf{x}_i^{ns} = [x_i^{ns}, y_i^{ns}, t_i^{ns}] | 1 \leq i \leq N_{ns}\}$ is a batch of randomly-sampled spatiotemporal coordinates of the spatial domain in front of a wind turbine. The NS loss reflects the conformity between the RC-PINN predictions and the NS equations.

It is noteworthy that the computation of the measurement loss needs measurement data, i.e., LIDAR measurements, while the computation of the NS loss does not require any measurement data.

Combining the measurement loss (26) and NS loss (27), the loss function of the proposed anti-noise wind field reconstruction method can be expressed as

$$L = L_M + L_{NS}. \quad (28)$$

During the training process, the parameters θ_{rc} and $1/Re$ are updated by the Adam algorithm, with the aim of minimizing the loss function (28).

After the training is finished, the RC-PINN can be used to reconstruct the spatiotemporal wind field. The reconstruction process is illustrated in the part (B) of Fig. 5. One may see that, the spatiotemporal coordinate of interest ($[x_i, y_i, t_i]$) is firstly fed into the RC-MLP, and then automatic differentiation is applied on the output of RC-MLP. After automatic differentiation has been applied, the corresponding two-dimensional wind velocity vector ($[u(x_i, y_i, t_i), v(x_i, y_i, t_i)]$) is obtained, instantly. Further, we can input a batch of spatiotemporal coordinates $\{[x_i, y_i, t_{Mol}] | 1 \leq i \leq N\}$ to obtain the wind velocity vectors of the whole wind field in front of a wind turbine at the moment of interest (as shown in the light grey box in the part (B) of Fig. 5). N is the number of spatial sampling locations in the wind field, and t_{Mol} is the time coordinate corresponding to the moment of interest.

Based on the wind velocity vector, the wind speed and direction can be directly computed, which are commonly used in engineering practice.

4.3. The mechanism behind the anti-noise capability of the proposed method

In this subsection, through both theoretical analysis and numerical simulation, the mechanism behind the anti-noise capability of the proposed method is demonstrated.

As mentioned in Section 4.1, the essentiality of incorporating the NS equations into the wind field reconstruction is two-fold. On the one hand, NS equations provide the wind dynamics of the whole wind field. On the other hand, during the training process, the noisy LIDAR measurements may not conform to the fluid physics law and mislead the training process, while NS equations can be regarded as constraints that correct the training process, thus guaranteeing the reconstruction accuracy.

In view of this, as for a deep learning-based wind field reconstruction method, the key to the anti-noise capability lies in the efficient utilization of NS equations, especially their constraining effect during the training process. Unfortunately, in traditional PINN-based methods, due to the gradient vanishing problem, the NS equations are difficult to fully leverage. In contrast, in the proposed method, the utilization of RC-PINN overcomes the gradient vanishing problem, and hence allowing for the NS equations to be fully leveraged during training. This is the mechanism behind the anti-noise capability of the proposed method. In the following part of this subsection, we will demonstrate this mechanism via both numerical simulation and theoretical analysis.

Firstly, we demonstrate this mechanism via numerical simulation. Specifically, the training process of the proposed method is investigated, thus showing its capability of mitigating gradient vanishing. The training process of the traditional PINN-based method is also simulated, which serves as a benchmark for comparison. A training set consisting of noisy LIDAR measurements and wind field spatiotemporal coordinates is adopted. The noise level of the noisy LIDAR measurements is 0.1 m/s, which represents the typical noise level of commercially available LIDAR [16]. This training set is used for both the traditional PINN-based method and the proposed method.

The comparison results are shown in Fig. 6 and Fig. 7. Fig. 6 shows the training behavior of the traditional PINN-based method and the proposed method. Specifically, the probability density of the automatic differentiation outputs at the beginning of training is plotted in this figure. These outputs consist of first-order derivatives (u, v, p_x, p_y), second-order derivatives (u_x, u_y, v_x, v_y), and third-order derivatives ($u_{xx}, u_{yy}, v_{xx}, v_{yy}$). As shown in Fig. 6(a), obviously, as for the traditional PINN-based method, all of the first, second, and third-order

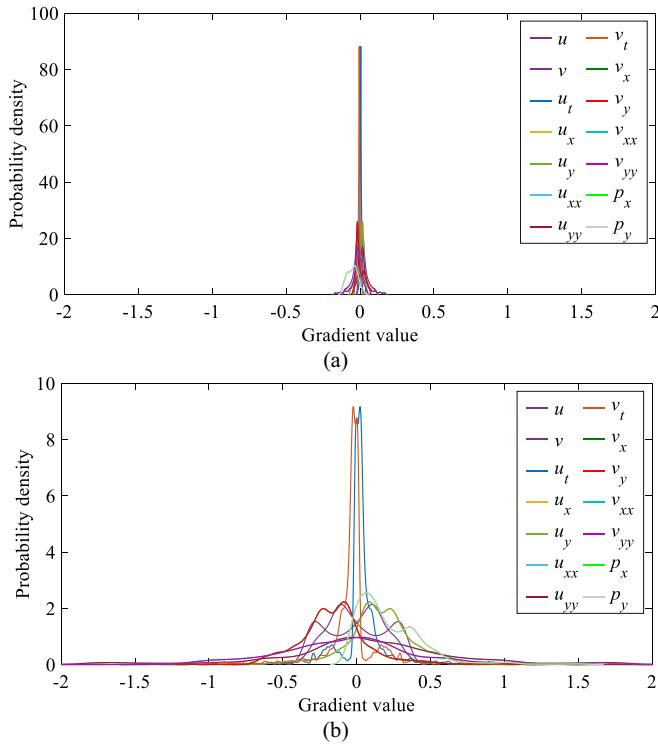


Fig. 6. The probability density of the first-order (u, v, p_x, p_y), second-order ($u_t, u_x, u_y, v_t, v_x, v_y$), and third-order ($u_{xx}, u_{yy}, v_{xx}, v_{yy}$) derivative outputs of automatic differentiation at the beginning of training, in (a) the traditional PINN-based method and (b) the proposed method. Note that, since the training set is nondimensionalized by characteristic length, characteristic velocity, and characteristic time, the gradient values are dimensionless.

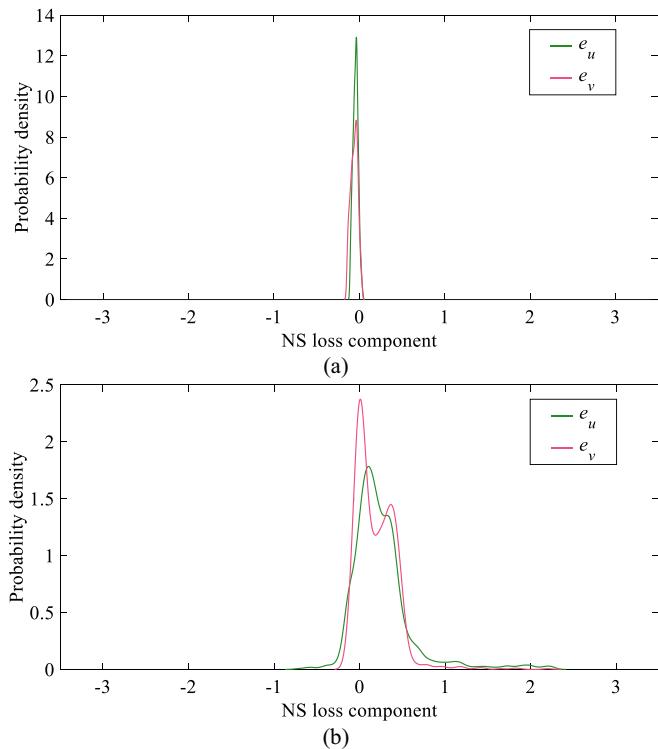


Fig. 7. The probability density of the NS loss components at the beginning of training, (a) traditional PINN-based method, (b) proposed method. Note that, since the training set is nondimensionalized by characteristic length, characteristic velocity, and characteristic time, e_u and e_v are dimensionless.

partial derivative outputs of the automatic differentiation process are highly concentrated around zero. This indicates that, the gradient vanishing exists in the first, second, and third-order derivative outputs of automatic differentiation. This is caused by the deep structure of MLP in PINN. Conversely, from Fig. 6(b), one can see that with the proposed method, the distributions of these outputs are all much more uniform, although the RC-MLP in the RC-PINN is equally deep as the MLP in the PINN. This indicates that with the proposed method, the absolute values of automatic differentiation outputs are much larger, i.e., the gradient vanishing problem of all the first, second, and third-order partial derivative outputs is significantly alleviated. This further supports the findings of Corollary 1.

Fig. 7 illustrates the probability density of the NS loss components at the beginning of training. Revisiting the NS loss, i.e., e_u in (22) and e_v in (23), one can see that both of them contain a subtraction term weighted by $1/Re$. In fact, in a wind farm, the magnitude of Reynolds number Re can be as large as 10^7 [55], so the value of $1/Re$ will be very small. Considering this fact, it is reasonable to ignore the subtraction terms in (22) and (23). By doing so, the magnitudes of both e_u and e_v are positively correlated with the magnitude of the automatic differentiation results. Therefore, in the traditional PINN-based method, even at the beginning of training, the values of e_u and e_v will be close to zero. In contrast, as for the proposed method, this issue is alleviated thanks to RC-PINN. This is confirmed by the results in Fig. 7. As shown in Fig. 7(a), with the traditional PINN-based method, the distributions of e_u and e_v are concentrated around zero. In comparison, as shown in Fig. 7(b), with the proposed method, the values of e_u and e_v are obviously larger. It is noteworthy that, as for the traditional PINN-based method, the underlying cause of the observed skewed distribution in Fig. 7(a) is the deep structure of MLP in PINN. In other words, it is an inherent problem of the traditional PINN-based method due to its structure. Therefore, this issue will not vanish during the training process.

Secondly, we investigate the anti-noise mechanism of the proposed method from a theoretical perspective. Inspired by [56], we investigate this mechanism by quantifying the effect of the NS equations on the training performance. To do so, the backpropagation gradients of the NS loss w.r.t. the NN training parameters are computed, that is

$$\nabla_{\theta} L_{NS} = \frac{\partial L_{NS}}{\partial \theta} = \frac{2}{N_{ns}} \sum_{i=1}^{N_{ns}} \left[e_{u,i} \frac{\partial e_{u,i}}{\partial \theta} \right] + \frac{2}{N_{ns}} \sum_{i=1}^{N_{ns}} \left[e_{v,i} \frac{\partial e_{v,i}}{\partial \theta} \right], \quad (29)$$

where $e_{u,i}$ and $e_{v,i}$ represent, respectively, the e_u and e_v at the i -th selected spatiotemporal coordinate in the wind field and θ is a vector of all training parameters. From the analysis of Fig. 7(a), if gradient vanishing occurs in the automatic differentiation, the values of $e_{u,i}$ and $e_{v,i}$ will be close to zero in the training process. Correspondingly, from (29), the distribution of the values of $\nabla_{\theta} L_{NS}$ will be concentrated around zero. Therefore, as for the traditional PINN-based method, due to the gradient vanishing problem, the values of $\nabla_{\theta} L_{NS}$ concentrate around zero. Conversely, as for the proposed method, due to the adoption of RC-PINN, the gradient vanishing problem is alleviated, and thus the values of $\nabla_{\theta} L_{NS}$ are distributed more uniformly.

Similarly, by computing the backpropagation gradients of the measurement loss w.r.t. the NN training parameters, we quantify the effect of the LIDAR measurements on the training performance, that is

$$\begin{aligned} \nabla_{\theta} L_M &= \frac{\partial L_M}{\partial \theta} = \\ &\frac{2}{N_l} \sum_{i=1}^{N_l} \left\{ \left[(\hat{u}_i^l \cos \alpha + \hat{v}_i^l \sin \alpha) - V_i^l \right] \frac{\partial (\hat{u}_i^l \cos \alpha + \hat{v}_i^l \sin \alpha)}{\partial \theta} \right\} \\ &+ \frac{2}{N_r} \sum_{i=1}^{N_r} \left\{ \left[(\hat{u}_i^r \cos \alpha - \hat{v}_i^r \sin \alpha) - V_i^r \right] \frac{\partial (\hat{u}_i^r \cos \alpha - \hat{v}_i^r \sin \alpha)}{\partial \theta} \right\}. \end{aligned} \quad (30)$$

Here, \hat{u}_i^l and \hat{v}_i^l represent, respectively, the predictions for u and v at the i -th selected spatiotemporal coordinate of the measurement point in the

left beam. \hat{u}_i^r and \hat{v}_i^r represent, respectively, predictions for u and v at the i -th selected spatiotemporal coordinate of the measurement point in the right beam. From (30), one can see that, unlike the case of $\nabla_\theta L_{NS}$, due to the presence of LIDAR measurements V_i^l and V_i^r , the values of $\nabla_\theta L_M$ will not concentrate around zero in the presence of gradient vanishing. In other words, the absolute values of $\nabla_\theta L_M$ are not significantly decreased by the gradient vanishing problem.

Consequently, in each training iteration of the traditional PINN-based method, the values of $\nabla_\theta L_{NS}$ will be significantly smaller than those of $\nabla_\theta L_M$. This indicates that, as for the traditional PINN-based method, the training process is strongly influenced by LIDAR measurements, thereby limiting the full utilization of the NS equations. In contrast, in each iteration of the proposed method, there is a better balance between the values of $\nabla_\theta L_{NS}$ and $\nabla_\theta L_M$. This indicates that, the training process of the proposed method is not completely dominated by LIDAR measurements, i.e., the NS equations also play an important role during training. In this sense, the NS equations are leveraged more sufficiently in the proposed method.

Thirdly, we again go back to the numerical simulation investigation. This time, the probability density distributions of $\nabla_\theta L_{NS}$ and $\nabla_\theta L_M$ are investigated. For ease of presentation, $\nabla_\theta L_{NS}$ and $\nabla_\theta L_M$ are normalized as follows

$$\nabla_\theta L_{NS}^*(i) = \frac{\nabla_\theta L_{NS}(i)}{\max(|\nabla_\theta L_M(i)|)}, \quad (31)$$

$$\nabla_\theta L_M^*(i) = \frac{\nabla_\theta L_M(i)}{\max(|\nabla_\theta L_M(i)|)}. \quad (32)$$

where $\nabla_\theta L_{NS}(i)$ and $\nabla_\theta L_{NS}^*(i)$ represent $\nabla_\theta L_{NS}$ and normalized $\nabla_\theta L_{NS}$ at the i -th training iteration, respectively. $\nabla_\theta L_M(i)$ and $\nabla_\theta L_M^*(i)$ represent $\nabla_\theta L_M$ and normalized $\nabla_\theta L_M$ at the i -th training iteration, respectively.

The resulted probability density distributions of $\nabla_\theta L_{NS}^*$ and $\nabla_\theta L_M^*$ are shown in Fig. 8. As shown in Fig. 8(a), with the traditional PINN-

based method, the values of $\nabla_\theta L_{NS}^*$ are, in general, significantly smaller than those of $\nabla_\theta L_M^*$. This indicates that LIDAR measurements dominate the training process of the traditional PINN-based method, whereas the role played by the NS equations is limited. Conversely, as shown in Fig. 8(b), in the training process of the proposed method, a much better balance between the values of $\nabla_\theta L_{NS}^*$ and those of $\nabla_\theta L_M^*$ is achieved. This indicates that, with the proposed method, both LIDAR measurements and NS equations contribute adequately to the training process. In this sense, compared to the traditional PINN-based method, the proposed method leverages the NS equations more adequately during the training process.

Finally, based on the above theoretical analysis and numerical simulations, we may summarize the underlying mechanism behind the anti-noise ability of the proposed method as follows. 1) The proposed method employs the RC-PINN, which alleviates the gradient vanishing problem arising in the automatic differentiation process. This allows the NS equations to be leveraged more adequately during the training process. 2) Due to the more efficient use of NS equations, the constraining effect of NS equations is improved, thus preventing the LIDAR measurement noise from deteriorating the training process. 3) With the more efficient utilization of NS equations and their corresponding improved constraining effects, the proposed method achieves a much stronger anti-noise capability.

5. Numerical results

In this section, the proposed anti-noise wind field reconstruction method is extensively evaluated and compared with benchmarks.

5.1. Dataset

In this study, the high-fidelity large eddy simulation solver SOWFA is used to generate the spatiotemporal wind velocity vectors in the wind field [57]. The SOWFA has been successfully used to simulate atmospheric boundary layer flows and wind turbine wakes in many works [58–61]. For the settings of SOWFA simulation, we refer to the setting in [14] and reduce the mesh size to further improve the simulation accuracy. The size of the whole simulation domain is 3000 m × 3000 m × 1000 m. In this domain, we use a uniform mesh of size 10 m × 10 m × 10 m. So, there are totally 9×10^6 cells in the simulation domain. To establish the initial state of the wind field, a 20,000 s simulation with a time step of 0.5 s is carried out firstly, and then a 1000 s simulation with a time step of 0.02 s is carried out to generate the more detailed wind velocity data. The instantaneous wind velocity at hub height of the entire SOWFA simulation domain is illustrated in Fig. 9. The colors in this figure represent the magnitude of wind velocity, and their corresponding mapping is illustrated in the colorbar. Subsequently, we collect the data from the last 100 s of simulations. From this 100 s of data, we further extract the turbine hub-height wind velocity vectors within a reconstruction area (demarcated by the black box in Fig. 9). The average freestream wind speed in this area is 7.76 m/s. The size of this area is determined based on the LIDAR measurement range. The wind velocity vectors within this area, together with their corresponding spatiotemporal coordinates, constitute the wind velocity dataset, which is used in the following simulations.

LIDAR measurement dataset is generated using the same method as in [14]. Specifically, the measurement range of the LIDAR is set to 220 m, and the distance between adjacent LIDAR measurement points is set to 20 m. Each LIDAR beam includes 11 spatial measurement points in total. The angle α is 15°. The time resolution of the LIDAR measurements is 1 s. Considering the yaw control of wind turbines, we select the mean wind direction in the wind velocity dataset as the LIDAR look direction. With these settings, the LIDAR measurement dataset can be obtained based on the wind velocity dataset. Specifically, by mapping the wind velocity vectors at the measurement locations to the measurement

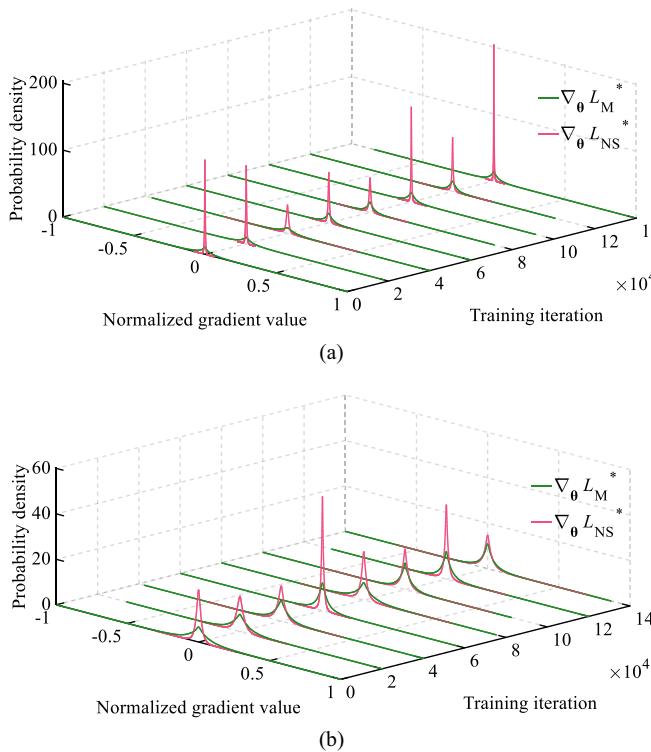


Fig. 8. The probability density distributions of $\nabla_\theta L_{NS}^*$ and $\nabla_\theta L_M^*$ during the training process, with (a) the traditional PINN-based method and (b) the proposed method.

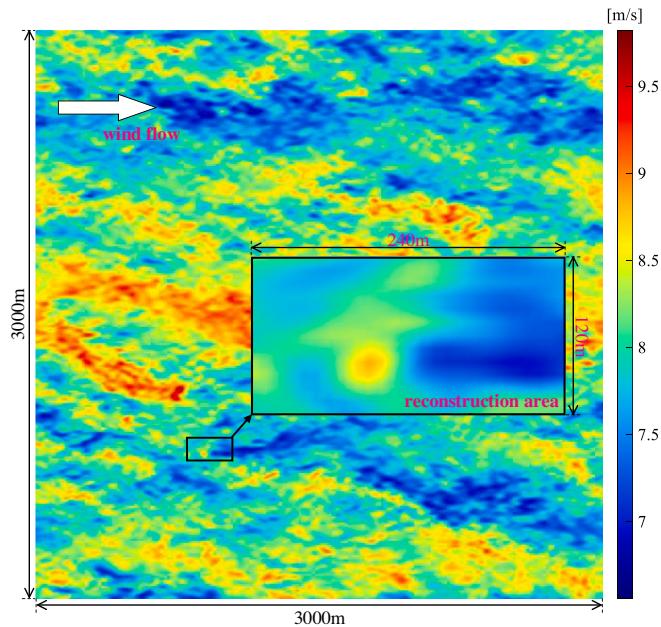


Fig. 9. The simulation domain at turbine hub height, together with the reconstruction area (magnified plot).

directions using (18), we obtain the corresponding LIDAR measurements. Moreover, random noise is added to the collected LIDAR measurements, thus simulating the real-world LIDAR measurement noise. The noise is generated from the uniform distribution $U(-n, n)$ [36], where n is noise level.

Furthermore, as for both the proposed method and benchmarks, a NS dataset is required in the training process. This dataset contains the spatiotemporal coordinates of the reconstruction area. In this study, this dataset includes $100 \times 97 \times 49$ coordinates in the domain of $[1100] \text{ s} \times [-240, 0] \text{ m} \times [-60, 60] \text{ m}$. The temporal resolution of these coordinates is 1 s, and their spatial resolution is 2.5 m.

It is worth noting that in following simulations, only the LIDAR measurement dataset and the NS dataset are used for training. The wind velocity dataset, which includes the wind velocity vectors of the whole reconstruction area, is only used as the ground truth for validation.

In all, in this study, the training dataset consists of two components. The first component is the LIDAR measurement dataset, which is used to calculate the measurement loss. In this dataset, the inputs and outputs are the spatiotemporal coordinates of the LIDAR measurement locations and the corresponding LIDAR measurements, respectively. The second component is the NS dataset. This dataset has only inputs, which are the spatiotemporal coordinates of the reconstruction area. It is using these spatiotemporal coordinates that the NS loss is calculated. The test dataset, which is used for validation, is the aforementioned wind velocity dataset. The inputs of this test dataset are the spatiotemporal coordinates of the uniform meshes in the reconstruction area. The outputs of this test dataset, which is used for validation, are the true spatiotemporal wind velocity vectors of the corresponding coordinates obtained from SOWFA simulation.

5.2. Description of numerical study

In order to demonstrate the superiority in wind field reconstruction and anti-noise ability of the proposed method, we compare it with benchmarks. As for benchmarks, besides the traditional PINN-based method in [14], two additional benchmarks are also employed. These two benchmarks are, respectively, built upon two innovative PINN frameworks, i.e., the adaptive weight PINN (which introduces adaptive weights to the data-fit terms of the loss function) in [56] and the

adaptive activation function PINN (which introduces a scalable hyperparameter in the activation function) in [62]. Using the approach described in subsection 4.1, we incorporate LIDAR measurements and the NS equations into these two innovative PINN frameworks. Then, these two benchmarks, referred to as AW-PINN-based and AA-PINN-based wind field reconstruction methods, are established.

As for hyperparameters, the optimal hyperparameters in [14] are used for both the proposed method and benchmarks. These hyperparameters are summarized in Table 1.

The number of fully connected layers is 12, indicating that there are 5 RC-blocks in the proposed method. No activation layer is applied to the output layer of the proposed method and all benchmarks.

The mean value of the root-mean-squared errors (RMSE) between the reconstructed wind field and the true wind field is used as the criterion of reconstruction accuracy, as follows:

$$E_s = \frac{1}{T} \sum_{t=1}^T RMSE_s^t, \quad (33)$$

$$E_d = \frac{1}{T} \sum_{t=1}^T RMSE_d^t, \quad (34)$$

$$RMSE_s^t = \sqrt{\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (\hat{\gamma}_{x_i, y_i, t} - \gamma_{x_i, y_i, t})^2}, \quad (35)$$

$$RMSE_d^t = \sqrt{\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (\hat{\varphi}_{x_i, y_i, t} - \varphi_{x_i, y_i, t})^2}. \quad (36)$$

Here, E_s is the RMSE between the reconstructed wind speed and true wind speed, and E_d is the RMSE between the reconstructed wind direction and true wind direction. $RMSE_s^t$ is the root-mean-squared errors (RMSE) between the reconstructed wind speed and true wind speed at timestamp t . $RMSE_d^t$ is the RMSE between the reconstructed wind direction and true wind direction at timestamp t . The reconstruction time period T is 100 s, the total number of test points N_{test} is 32,500, and $\{(x_i, y_i), 1 \leq i \leq N_{test}\}$ is the 25×13 uniform spatial coordinates in the reconstruction area. $\hat{\gamma}_{x_i, y_i, t}$ and $\gamma_{x_i, y_i, t}$ represent the corresponding reconstructed wind speed and true wind speed, respectively. $\hat{\varphi}_{x_i, y_i, t}$ and $\varphi_{x_i, y_i, t}$ represent the corresponding reconstructed wind direction and true wind direction, respectively.

Compared to the wind velocity vector, the wind speed and direction provide a more intuitive depiction of the wind velocity field and are more widely employed in engineering practice. Therefore, in this study, the wind velocity vector is further converted into wind speed (m/s) and direction ($^\circ$) for the presentation and evaluation of the reconstruction results.

It is worth noting that the proposed anti-noise wind field reconstruction is mesh-free. In other words, input any spatiotemporal coordinate of the reconstruction area, and the trained NN can output the wind velocity vector at the corresponding location instantly. However, for validation purpose, the input coordinates are chosen to be consistent

Table 1

The hyperparameters used in this case study for both the proposed method and benchmark methods. These hyperparameters were also used in [14].

Item	Value
Number of fully connected layers N_f	12
Number of activation layers N_{act}	11
Number of neurons in hidden layer N_h	128
The batch size for NS loss N_{ns}	1000
The batch size for measurement loss of left beam N_l	1100
The batch size for measurement loss of right beam N_r	1100
Learning rate η	10^{-4}

with the mesh coordinates used in the SOWFA simulation.

The computational environment in this study is a desktop computer with an i5-10400F CPU, 16 GB of memory, and a GeForce GTX 1650 GPU.

5.3. Results and analysis

To fully assess the anti-noise capability of the proposed method, we test the proposed method in four cases. In these four cases, the noise level of LIDAR measurements is different, i.e., the noise level n in these cases is 0.025 m/s, 0.05 m/s, 0.1 m/s, and 0.2 m/s, respectively. Moreover, to assess the effectiveness of the proposed method in the absence of noise, another case that the value of n is 0 m/s is also considered. For these five cases, the reconstruction performances of the proposed method and benchmarks are reported in Table 2.

From this table, one may see that, in all five cases, the proposed method gives the lowest E_s and E_d . This indicates that the proposed method exhibits the highest reconstruction accuracy for both wind speed and direction. The reason for the aforementioned superiorities lies in the fact that the use of RC-PINN allows the proposed method to fully leverage the NS equations. Furthermore, it is noteworthy that, among all four methods, the AW-PINN-based method performs worst. The main reason is that, in the AW-PINN-based method, two adaptive weights are, respectively, applied to the first and second terms in measurement loss (26). The mean values of these two weights are approximately 60 in training. Following the chain rule, these weights further amplify the gradient of measurement loss in training and exacerbate the imbalance in gradients, thereby significantly hindering the effectiveness of the NS equation in training. This further verifies the importance of fully leveraging the NS equations in training.

Based on the wind field reconstruction results in Table 2, we further calculate the reconstruction accuracy improvement of the proposed method over benchmark methods, that is

$$impr = \frac{1}{3} \sum_{i=1}^3 \frac{(\bar{E}_d^i - \tilde{E}_d^i)/\bar{E}_d^i + (\bar{E}_s^i - \tilde{E}_s^i)/\bar{E}_s^i}{2}. \quad (37)$$

Here, $impr$ represents the reconstruction accuracy improvement of the proposed method over all benchmark methods. \bar{E}_d^i and \tilde{E}_d^i represent, respectively, E_d of the i -th selected benchmark method and the proposed method. \bar{E}_s^i and \tilde{E}_s^i represent, respectively, E_s of the i -th selected benchmark method and the proposed method. The calculation results are illustrated in Fig. 10.

From the results in Fig. 10, one can see that, compared to the reconstruction accuracy improvement under the zero noise level case, the proposed method exhibits a greater improvement in the presence of LIDAR measurement noise. This indicates that the proposed method has more obvious advantages in noisy scenarios. The reason for this is that, in the zero noise level case, the only advantage of the proposed method is that it enhances the importance of NS equations, which describe wind dynamics during training. In contrast, in the presence of LIDAR measurement noise, besides the aforementioned advantage, the proposed method also has the advantage of using NS equations as constraints more adequately to correct the training process, thus guaranteeing

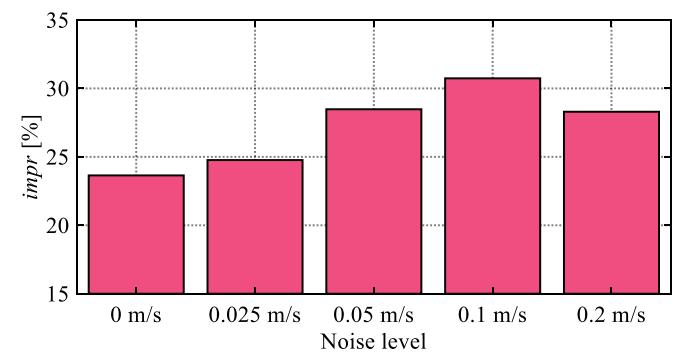


Fig. 10. The reconstruction accuracy improvement of the proposed method over all benchmark methods.

reconstruction accuracy. This also explains the phenomenon that, compared to low noise level (i.e., noise level of 0.025 m/s), the improvement is higher under relatively high noise levels (i.e., noise levels of 0.05 m/s, 0.1 m/s, and 0.2 m/s).

It is noteworthy that at the typical noise level of the commercially available LIDAR (i.e., noise level of 0.1 m/s), the reconstruction accuracy improvement achieved by the proposed method is the highest (up to 30.7%). As a result, at this noise level, the proposed method achieves a reconstruction accuracy that is comparable to the highest reconstruction accuracy obtained by the benchmark methods under the noise-free assumption. This indicates that, as a deep learning-based wind field reconstruction method, the proposed method avoids dependence on perfect and ideal LIDAR measurements. In other words, it makes it possible to achieve high accuracy in the reconstruction of real-world wind field based on commercially available LIDAR.

Moreover, the reconstruction accuracy improvement brought by the proposed method does not consistently increase with the increase in noise level. At the noise level of 0.2 m/s, which is twice the noise level of the commercially available LIDAR, the improvement brought by the proposed method is lower than that at the noise level of 0.1 m/s. The reason for this is that at this noise level, the LIDAR measurements are too noisy to provide real-time information of the true wind field. Therefore, the advantage of the proposed method in better utilizing the NS equations begins to be limited. Nevertheless, at this noise level, the accuracy improvement brought by the proposed method is still high (up to 28.3%). This indicates that, even using low-accuracy LIDAR instruments, compared to the benchmark methods, the proposed method is able to achieve higher reconstruction accuracy.

Next, to evaluate the stability of the proposed method's reconstruction accuracy in the time dimension, the variance of the root-mean-squared errors (VRMSE) between the reconstructed and the true wind fields is introduced, as follows:

$$V_s = \frac{1}{T} \sum_{t=1}^T (E_s - RMSE_s^t)^2, \quad (38)$$

$$V_d = \frac{1}{T} \sum_{t=1}^T (E_d - RMSE_d^t)^2. \quad (39)$$

Table 2
Wind field reconstruction results.

Noise level	E_s / E_d			
	AW-PINN-based method	AA-PINN-based method	Traditional PINN-based method	Proposed method
0.0000 m/s	0.2192 / 3.1277	0.1542 / 2.0899	0.1574 / 2.0692	0.1292 / 1.8207
0.0250 m/s	0.2313 / 3.0156	0.1451 / 1.9307	0.1865 / 2.3183	0.1327 / 1.8068
0.0500 m/s	0.2533 / 3.6055	0.1664 / 2.1843	0.2039 / 2.4902	0.1435 / 1.8990
0.1000 m/s	0.3010 / 4.1055	0.1854 / 2.5300	0.2129 / 2.8016	0.1560 / 2.0721
0.2000 m/s	0.4008 / 4.6737	0.2809 / 2.8448	0.2515 / 2.8043	0.2204 / 2.2681

Table 3

Test results of reconstruction accuracy stability in time dimension.

Noise level	$[V_s \times 10^{-2}] / V_d$			
	AW-PINN-based method	AA-PINN-based method	Traditional PINN-based	Proposed method
0.0000 m/s	0.6527 / 0.7900	0.1807 / 0.3046	0.2971 / 0.2745	0.0694 / 0.1220
0.0250 m/s	0.6733 / 0.3387	0.0964 / 0.1138	0.4713 / 0.3812	0.0688 / 0.0696
0.0500 m/s	0.2580 / 0.4107	0.4191 / 0.1895	0.5010 / 0.3954	0.1045 / 0.0540
0.1000 m/s	0.5645 / 1.3369	0.4121 / 1.2382	0.9003 / 2.2751	0.1029 / 0.1998
0.2000 m/s	1.3841 / 2.0719	1.0905 / 0.6132	0.7529 / 0.7711	0.3109 / 0.4861

Here, V_s is the VRMSE between the reconstructed wind speed and the true wind speed, and V_d is the VRMSE between the reconstructed wind direction and the true wind direction. The value of VRMSE reflects the differences in wind field reconstruction accuracy at various moments within the reconstruction time period T . Hence, a low value of VRMSE indicates greater stability of the reconstruction accuracy in the time dimension, which is vital for real-world applications. For the selected 5 cases, the VRMSE values of the proposed method and the benchmark methods are all presented in Table 3.

From this table, one can see that, in all five cases, the proposed method presents the lowest value of V_s and V_d . This indicates that, compared with benchmark methods, in the time dimension, the reconstruction accuracy of the proposed method is the most stable. Moreover, at the typical noise level of the commercially available LIDAR (i.e., noise level of 0.1 m/s), the V_s and V_d of the proposed method are significantly lower than the lowest V_s and V_d given by the benchmark methods under the noise-free assumption. This result is able to improve the reliability of the reconstruction results given by the proposed method in real-world applications.

To intuitively demonstrate the improvement of the proposed method, Figs. 11–12 present, respectively, the wind field reconstruction results at 25 s and 75 s with the noise level of 0.05 m/s. Similarly, the results for the noise levels of 0.1 m/s and 0.2 m/s are shown in Figs. 13–16, respectively. The subfigures (a), (c), (e), and (g) of Figs. 11–16 illustrate the reconstructed wind velocity field by benchmark methods and the proposed method, respectively. Meanwhile, the subfigures (i) illustrate the true wind velocity field. The colors in these subfigures represent the wind speed at various locations within the wind field. The colorbars in these subfigures provide a clear mapping between colors and wind speed, with blue colors indicating lower wind speed and red colors indicating higher wind speed. Furthermore, the direction of the arrows in these subfigures represents the wind direction at various locations within the wind field. Moreover, in the subfigures (i), we denote the LIDAR (which is represented by a pink rectangle) and LIDAR measurement points (which are represented by pink dots). In subfigures (b), (d), (f), and (h), we depict the magnitudes of the absolute differences between the true wind speed and the reconstructed wind speed at various locations within the wind field. The magnitude of the absolute differences is represented by the colors in these subfigures. Eventually, in subfigures (j), the $RMSE_s^t$ and $RMSE_d^t$ of benchmark methods and the proposed method are shown.

The reason for choosing the time stamps of 25 s and 75 s is that, the wind fields at these two timestamps represent two typical wind field conditions. As shown in Figs. 11–16, at 25 s, the variations in wind speed and direction are comparatively small throughout the wind field, indicating a relatively stable wind field. In comparison, at 75 s, the variations in wind speed and direction are great throughout the wind field, indicating a relatively unstable wind field.

From these six figures, it is obvious that, compared to the benchmark methods, the reconstructed wind speed and direction given by the proposed method are much closer to the true wind speed and direction. It demonstrates that, under both relatively stable and unstable wind conditions, the proposed method achieves the best reconstruction performance at all three selected noise levels.

Specifically, as for the reconstructed performance at 25 s, under each

selected noise level, wind speed reconstruction error is primarily observed in the high-speed flow region within the coordinate range $x \in [-240, -170], y \in [0, 20]$ and in the region within the coordinate range $x \in [-80, 0], y \in [-60, -45]$. For the reconstructed error within the coordinate range $x \in [-240, -170], y \in [0, 20]$, at the noise level of 0.05 m/s, it becomes noticeably observable in the benchmark methods. Conversely, in the proposed method, it becomes evident until the noise level reaches 0.2 m/s. For the reconstructed error within the coordinate range $x \in [-80, 0], y \in [-60, -45]$, although it is present in the benchmark methods and the proposed method at each selected noise level, it is consistently noticeably lower in the proposed method. Moreover, at the noise level of 0.2 m/s, the benchmark methods exhibit obvious error in the region within the coordinate range $x \in [-80, -15], y \in [40, 60]$, while such error does not occur in the proposed method.

As for the reconstructed performance at 75 s, under each selected noise level, wind speed reconstruction error is primarily present in the high-speed flow region within the coordinate range $x \in [-160, -120], y \in [-40, -5]$. Under the noise level of 0.1 m/s, this error in the proposed method is obviously the smallest. However, under the noise level of 0.05 m/s, except for the AW-PINN-based method, where this error is obviously larger, this error seems to be similar in the AA-PINN-based method, the traditional PINN-based method, and the proposed method. Nevertheless, from the overall view of subfigures (d), (f), and (h) within Fig. 12, it is evident that the proposed method continues to exhibit the smallest reconstruction error. Moreover, at the noise level of 0.2 m/s, compared with the AA-PINN-based method and the traditional PINN-based method, this error is higher in the proposed method. However, compared with benchmark methods, the error within other regions is significantly lower in the proposed method. For instance, at the region within the coordinate range $x \in [-230, -170], y \in [0, 20]$, in the AA-PINN-based method and the traditional PINN-based method, there is higher reconstruction error, while this error does not appear in the proposed method. This makes the proposed method retain the highest reconstruction accuracy under the noise level of 0.2 m/s. Furthermore, at certain noise levels, reconstruction error is noticeably observed in the region in the coordinate range $x \in [-20, 5], y \in [-240, -225]$ in the benchmark methods. Conversely, this phenomenon does not appear in the proposed method.

Moreover, from subfigures (j) of Figs. 11–16, it can be observed that, at the timestamps of 25 s and 75 s and across all three selected noise levels, the $RMSE_s^t$ and $RMSE_d^t$ of the proposed method are always the lowest. This again quantitatively indicates that, compared to benchmark methods, under both relatively stable and unstable wind conditions, the proposed method achieves the best reconstruction accuracy in wind speed and direction at all three noise levels.

Next, recognizing that the stochastic nature of LIDAR measurement noise can introduce uncertainty into the reconstruction accuracy, an analysis is conducted to evaluate this uncertainty. To this end, firstly, four additional noisy LIDAR measurement datasets are collected at each noise level (noise levels of 0.025 m/s, 0.05 m/s, 0.1 m/s, and 0.2 m/s). Including the original noisy LIDAR measurement datasets, there are a total of 5 noisy LIDAR measurement datasets for each noise level. Subsequently, to compare the reconstruction accuracy uncertainty of the proposed method and the benchmark methods at the noise level of commercially available LIDAR, the wind field reconstruction is

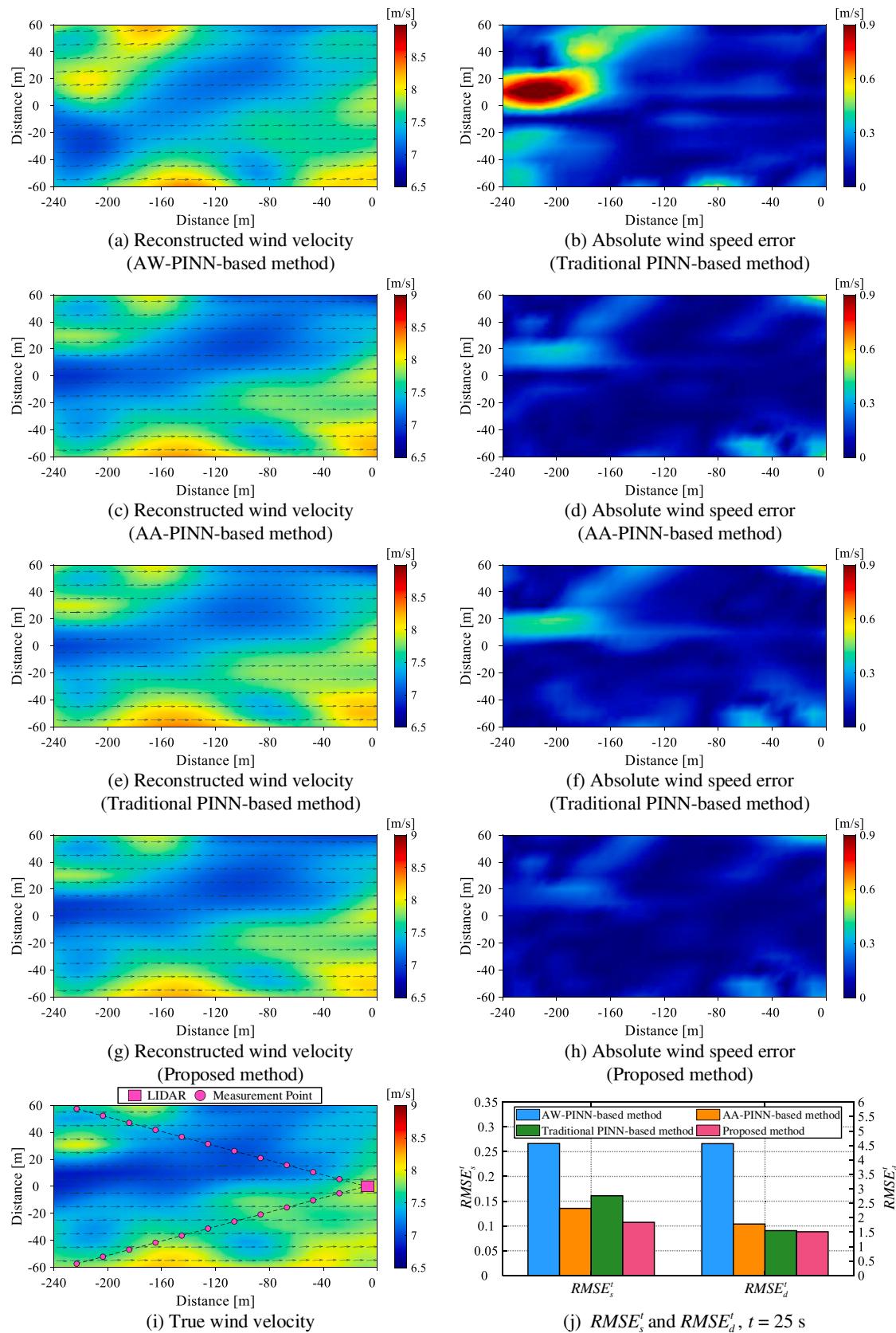
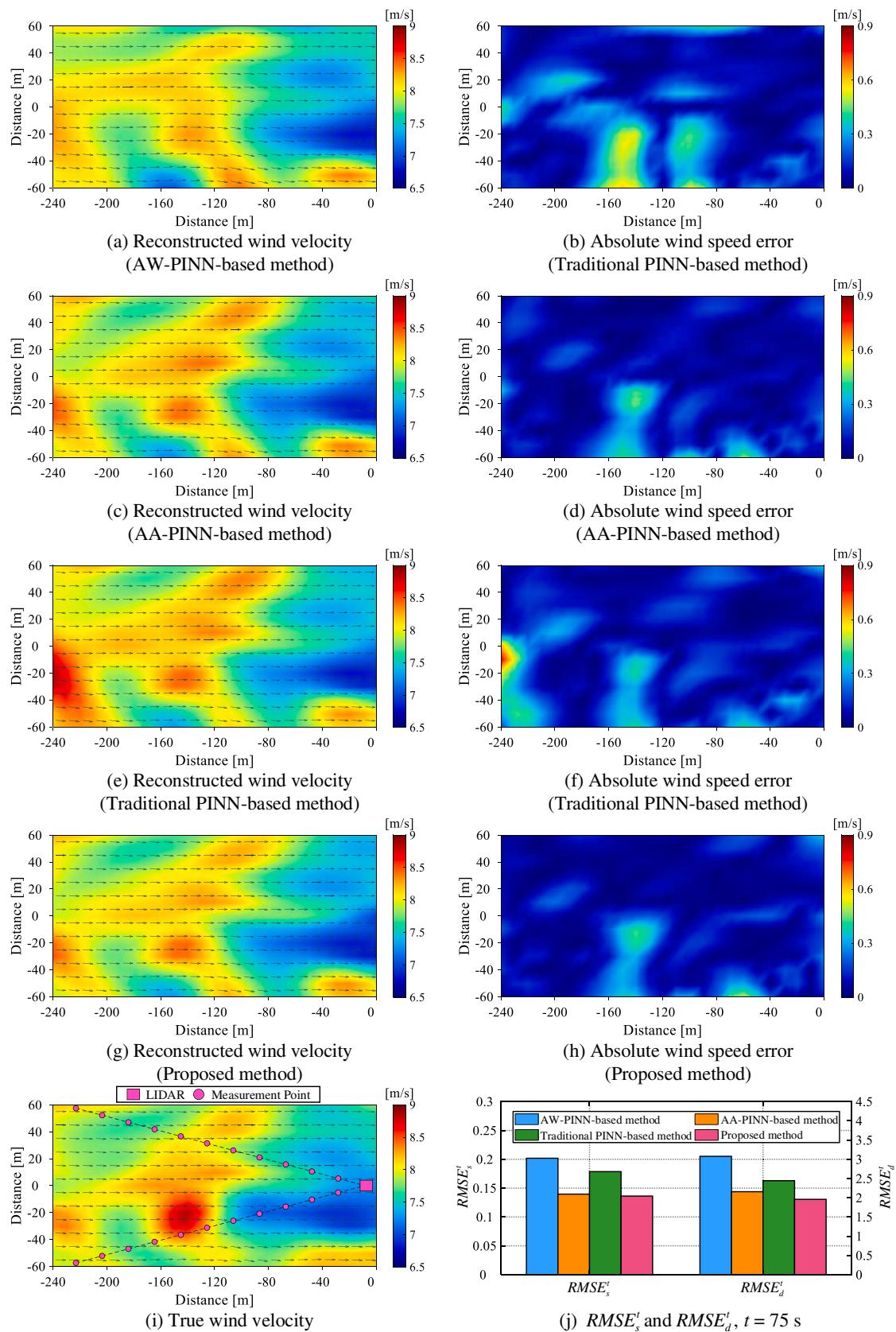


Fig. 11. Wind field reconstruction results ($t = 25$ s, noise level = 0.05 m/s).

Fig. 12. Wind field reconstruction results ($t = 75$ s, noise level = 0.05 m/s).

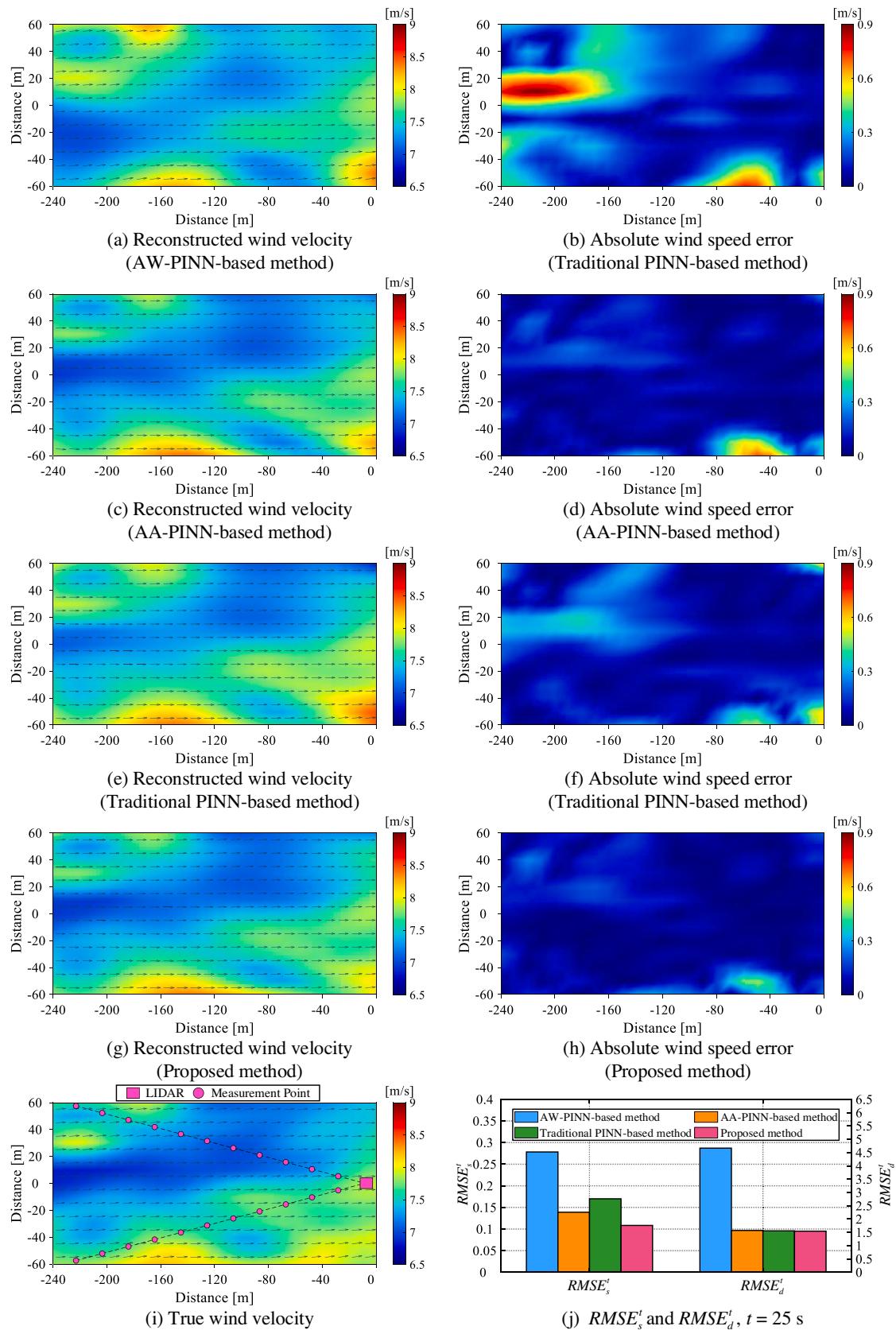


Fig. 13. Wind field reconstruction results ($t = 25$ s, noise level = 0.1 m/s).

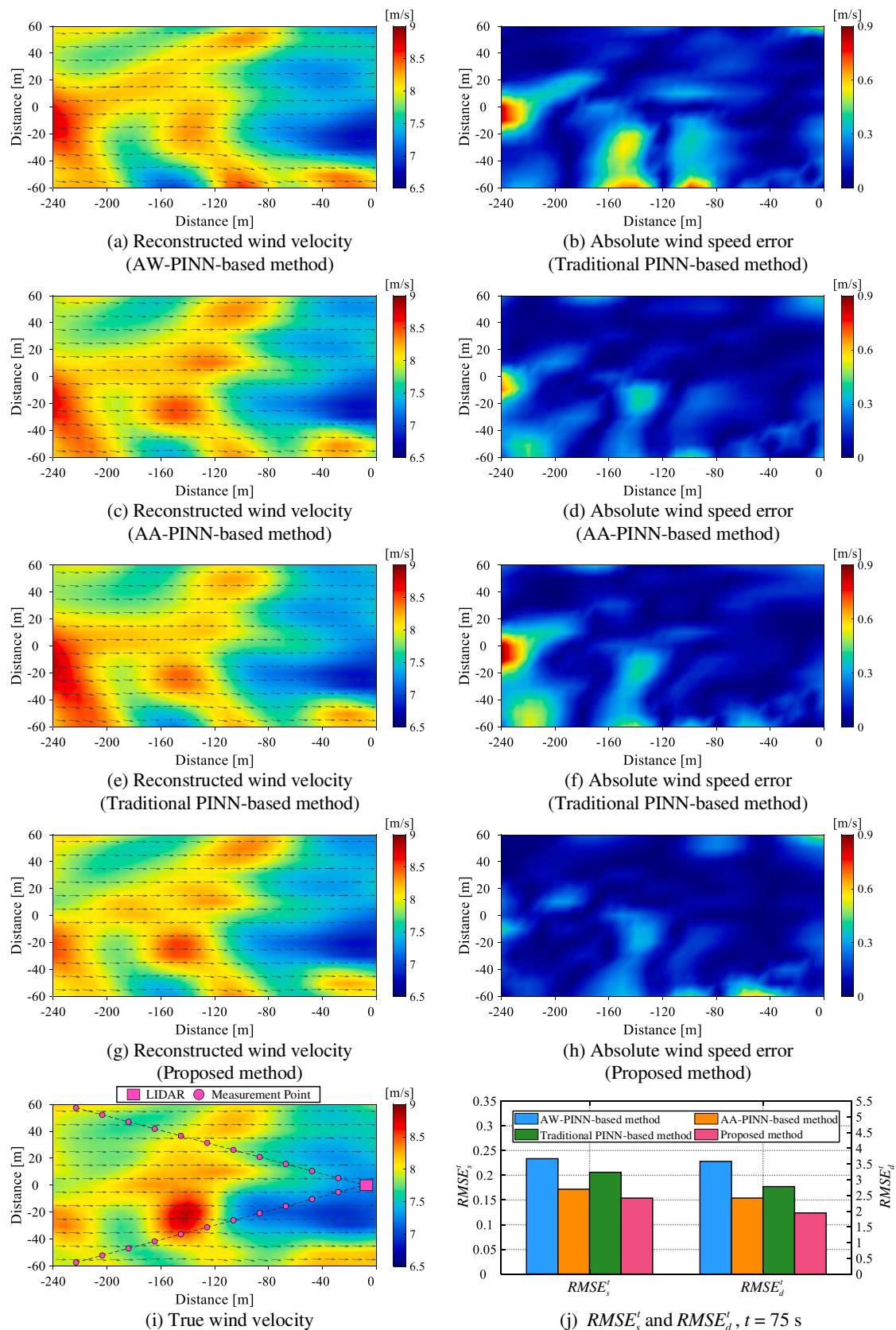


Fig. 14. Wind field reconstruction results ($t = 75$ s, noise level = 0.1 m/s).

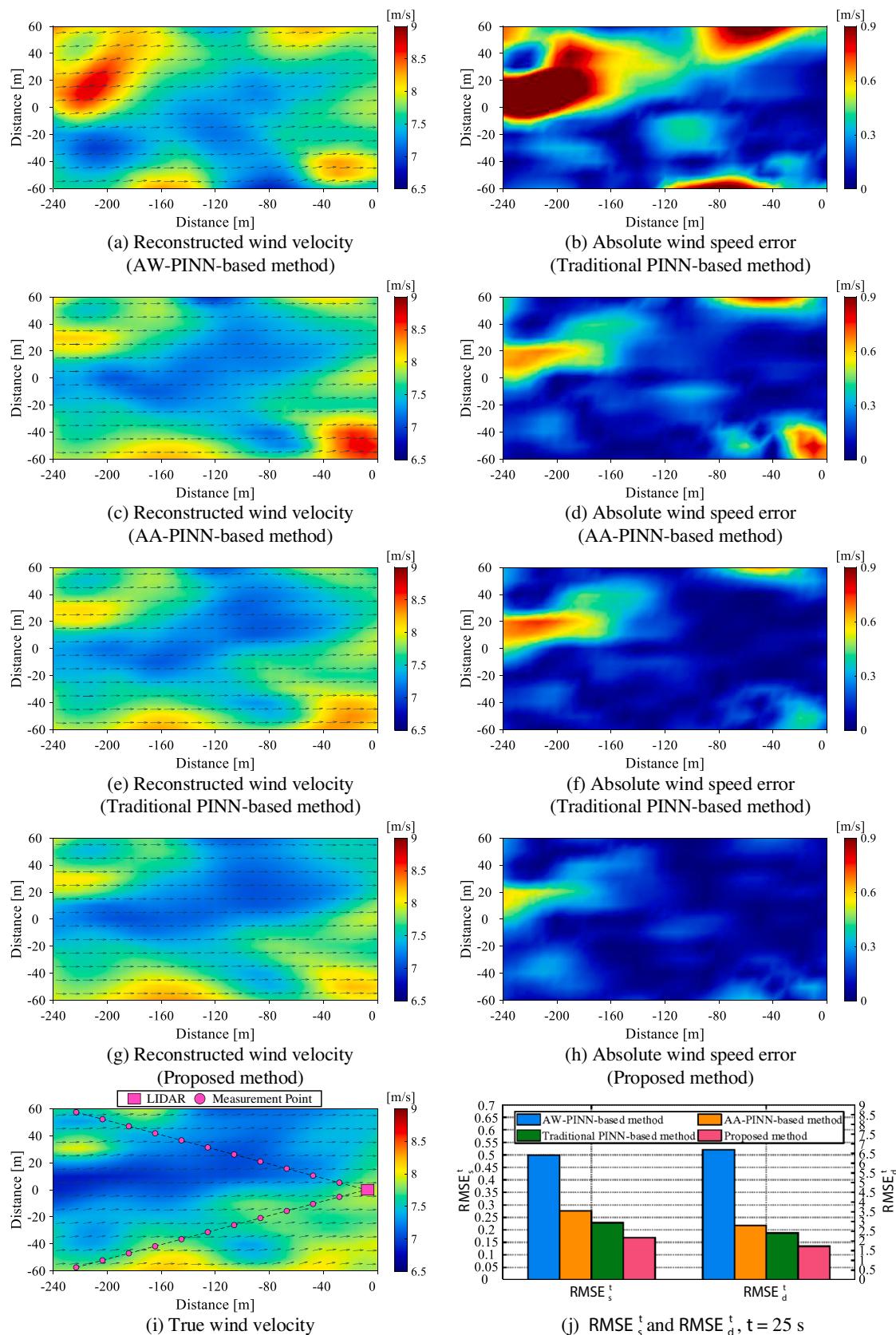


Fig. 15. Wind field reconstruction results ($t = 25$ s, noise level = 0.2 m/s).

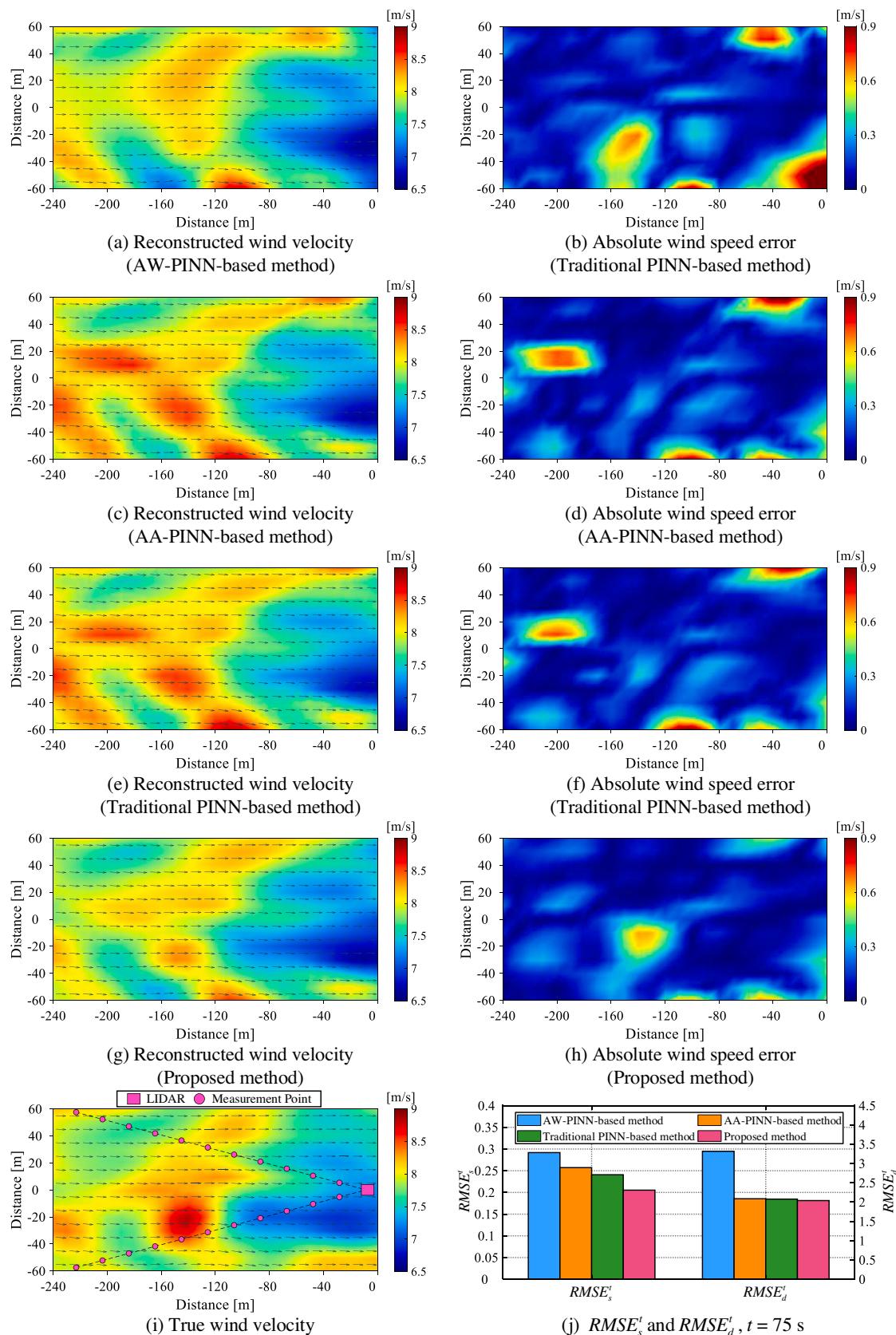


Fig. 16. Wind field reconstruction results ($t = 75$ s, noise level = 0.2 m/s).

conducted based on all methods, respectively, using the noisy LIDAR measurement datasets with a noise level of 0.1 m/s. Based on the reconstruction results, a standard error bar diagram is plotted in Fig. 17. From this figure, one can see that, compared to the benchmark methods, the proposed method provides the lowest mean values for E_s and E_d , as well as the lowest standard deviations for E_s and E_d . This indicates that, under the noise level of commercially available LIDAR, compared with benchmark methods, the uncertainty of reconstruction accuracy given by the proposed method is the lowest. That is, at the typical noise level of LIDAR, the proposed method exhibits the highest reconstruction accuracy while maintaining the greatest stability in reconstruction accuracy.

Then, to evaluate the reconstruction accuracy uncertainty of the proposed method at all four selected noise levels, using noisy LIDAR measurement datasets with the other three noise levels, three additional wind field reconstructions are conducted. The reconstruction results are presented in Fig. 18. From this figure, one can see that, under all four selected noise levels, the standard deviations of both E_s and E_d are relatively low. This indicates that, under all four noise levels, the proposed method consistently maintains low uncertainty in reconstruction accuracy. The above analysis demonstrates the superiority of the proposed method in the uncertainty of reconstruction accuracy. This superiority is attributed to the utilization of RC-PINN, which enhances the constraining effect of the NS equations, thereby mitigating the adverse impact of measurement noise on reconstruction accuracy.

To demonstrate the improvement of the proposed method in convergence speed, Fig. 19 shows the reconstruction error of both the proposed method and benchmark methods as a function of training iterations.

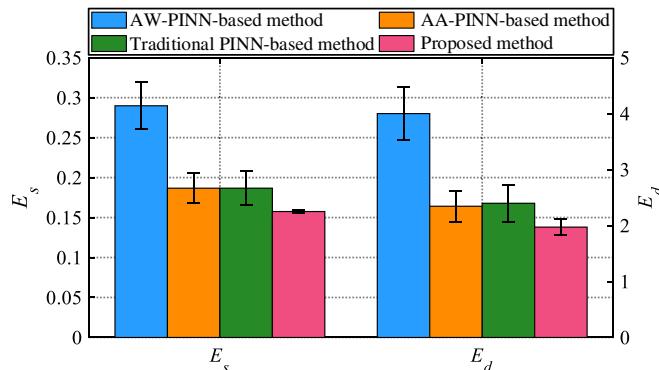


Fig. 17. The standard error bar diagram of the reconstruction results given by the proposed method and benchmark methods under the noise level of commercially available LIDAR.

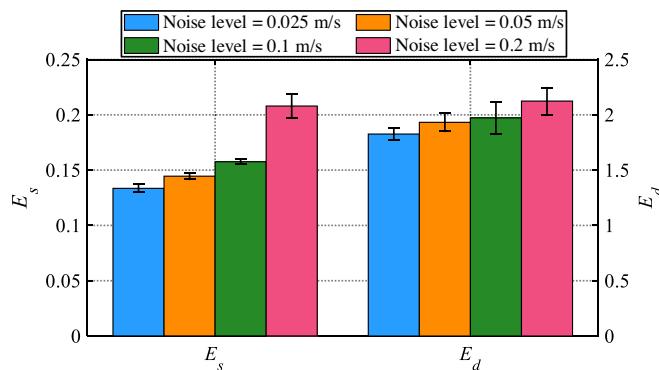


Fig. 18. The standard error bar diagram of the reconstruction results given by the proposed method under all selected noise levels.

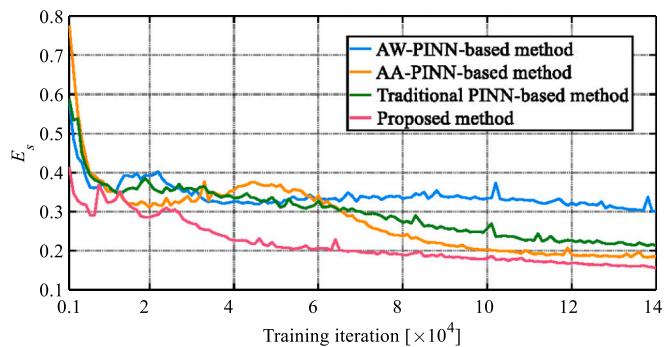


Fig. 19. Wind speed reconstruction error at each training iteration, noise level = 0.1 m/s.

From this figure, it is clear that in each training iteration, the reconstruction error produced by the proposed method is the lowest. This indicates that the proposed method has the fastest convergence speed. Our simulation results demonstrate that this fact also holds true for other noise levels, but the results are omitted for saving space. This superiority of the proposed method is attributed to its ability to fully leverage the NS equations, which optimizes its utilization of wind dynamics and makes the training process more conform to the fluid physical law, thereby improving the training efficiency.

As for the computational cost, firstly, the proposed method does not introduce any additional training parameters compared to the traditional PINN-based method. Secondly, compared to the traditional PINN-based method, the proposed method only adds some simple additional operations, i.e., addition. As a result, the computational time of the proposed method is almost the same as the traditional PINN-based method and the AA-PINN-based method (which introduces one additional training parameter) and faster than the AW-PINN-based method (which introduces additional gradient calculation). Specifically, for the proposed method, each training iteration requires about 0.33 s, and the reconstruction of the wind field at any time instant requires about 0.08 s. Referring back to Fig. 19, this indicates that the proposed method requires less time for the same accuracy.

Finally, based on the above simulation results and analysis, we may summarize the following conclusion: 1) compared to benchmark methods, the proposed method can achieve the best wind field reconstruction performance, especially in the presence of LIDAR measurement noise. 2) Compared to benchmark methods, the stability of reconstruction accuracy in the time dimension given by the proposed method is the highest. 3) As for the reconstruction accuracy uncertainty induced by the stochastic nature of LIDAR measurement noise, compared to benchmark methods, at the noise level of commercially available LIDAR, the proposed method demonstrates the lowest uncertainty. Furthermore, it consistently maintains this low uncertainty across all selected noise levels. 4) Compared to benchmark methods, the proposed method has the fastest convergence speed and requires the least computation time. 5) The above superiorities of the proposed method are attributed to the use of RC-PINN, which makes the proposed method able to fully leverage the NS equations during training.

However, the proposed method still has some limitations that merit further research to solve. First, similar to the conventional PINN and its various improved versions, the proposed RC-PINN also relies on automatic differentiation to incorporate physical information. In wind field reconstruction, the automatic differentiation allows the proposed method to directly incorporate the NS equations. Unfortunately, it also leads to relatively high computational costs. Second, the proposed RC-PINN is a neural network framework. Similar to other types of neural networks, the training of RC-PINN fundamentally involves solving a non-convex optimization problem. Currently, iterative methods are commonly used to solve this problem, but they do not guarantee the

attainment of a global optimal solution. In wind field reconstruction, compared to benchmark methods, the use of RC-PINN allows the proposed method to leverage the NS equations more adequately, thus leading to a more efficient iterative searching strategy. However, this can only ensure that the training of the RC-PINN converges faster to a better solution (a higher reconstruction accuracy). The global optimality of the solution still cannot be guaranteed.

6. Conclusion

In this work, we propose a novel anti-noise method to reconstruct the wind field based on real-world noisy LIDAR measurements. To this end, a RC-PINN framework is first established to alleviate the gradient vanishing problem, which arises in the automatic differentiation process and prevents the NS equations from being fully leveraged in the wind field reconstruction. The superiority of RC-PINN is then mathematically proven. Subsequently, based on RC-PINN, the anti-noise wind field reconstruction method is proposed. Through a hybrid approach involving both theoretical analysis and numerical simulation, the anti-noise mechanism behind the proposed method is demonstrated. Specifically, the utilization of the RC-PINN enables the proposed method to leverage the NS equations more adequately, thereby leading to enhanced anti-noise capability. Numerical simulation is carried out on the spatiotemporal wind data generated from the high-fidelity wind farm solver SOWFA. The results show that, compared with the benchmark methods, at a wide range of noise levels, the proposed method achieves a significant improvement in reconstruction performance. Especially, at the noise level of commercially available LIDAR, the reconstruction results are quite satisfactory. This gives the possibility of achieving a stable and highly accurate reconstruction of real-world wind field based on commercially available LIDAR. Furthermore, the enhancement of reconstruction accuracy stability in the time dimension, as well as the mitigation of the reconstruction accuracy uncertainty induced by the stochastic nature of measurement noise, make the proposed method more reliable in real-world applications. Moreover, the more adequate utilization of the NS equations also makes the proposed method converge faster in training.

Based on the real-world spatiotemporal wind speed data reconstructed by the proposed method, the wind energy assessment can be conducted, thereby providing guidance for decisions related to wind energy development. Besides, this data can also serve as data support for wind turbine and wind farm control.

Furthermore, since the proposed RC-PINN, like PINN, is a generic framework, its application is not limited to anti-noise wind field reconstruction. Specifically, by incorporating relevant physical information, the proposed RC-PINN can be applied to simulations in other energy fields. For instance, it can be applied to power system simulation in electrical energy, porous medium fluid flow simulation in geothermal energy, and tidal simulation in tidal energy.

Future work involves conducting wind field reconstruction based on measurements from multiple wind speed measuring instruments. The reason lies in the fact that, in addition to LIDAR, there are other types of wind speed measuring instruments in wind farms. Measurements from multi-type instruments can provide more real-time wind field information, thereby enhancing the accuracy of wind field reconstruction. Furthermore, utilizing detailed wind velocity data obtained through wind field reconstruction to guide wind farms' participation in power system frequency regulation is also one of the future research directions.

CRediT authorship contribution statement

Runze Tian: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis. **Peng Kou:** Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition, Conceptualization. **Yuanhang Zhang:** Writing – review & editing, Data curation. **Mingyang Mei:** Writing –

review & editing. **Zhihao Zhang:** Writing – review & editing. **Deliang Liang:** Resources.

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.

Acknowledgements

The authors gratefully acknowledge the financial support received from the National Natural Science Foundation of China under Grant 52077165. (Corresponding author: Peng Kou).

References

- [1] He R, Yang H, Sun S, et al. A machine learning-based fatigue loads and power prediction method for wind turbines under yaw control. *Appl Energy* 2022;326:120013.
- [2] Sun S, Wang T, Yang H, et al. Condition monitoring of wind turbine blades based on self-supervised health representation learning: a conducive technique to effective and reliable utilization of wind energy. *Appl Energy* 2022;313:118882.
- [3] Wang J, Wang S, Zeng B, et al. A novel ensemble probabilistic forecasting system for uncertainty in wind speed. *Appl Energy* 2022;313:118796.
- [4] Antonini EGA, Romero DA, Amon CH. Improving CFD wind farm simulations incorporating wind direction uncertainty. *Renew Energy* 2019;133:1011–23.
- [5] Santoni C, Ciri U, Rotea M, Leonardi S. Development of a high fidelity CFD code for wind farm control. In: 2015 American control conference. IEEE; 2015. p. 1715–20.
- [6] Yan BW, Li QS. Coupled on-site measurement/CFD based approach for high-resolution wind resource assessment over complex terrains. *Energ Convers Manage* 2016;117:351–66.
- [7] Harris M, Hand M, Wright A. Lidar for turbine control. Report no. NREL/TP-500-39154. National Renewable Energy Laboratory: Golden, CO; 2016.
- [8] Carcangiu CE, Pujana-Arrese A, Mendizabal A, Pinede I, Landaluze J. Wind gust detection and load mitigation using artificial neural networks assisted control. *Wind Energy* 2014;17(7):957–70.
- [9] Wang N, Johnson KE, Wright AD. Comparison of strategies for enhancing energy capture and reducing loads using LIDAR and feedforward control. *IEEE Trans Control Syst Technol* 2013;21(4):1129–42.
- [10] Mahdizadeh A, Schmid R, Oetomo D. Lidar-assisted exact output regulation for load mitigation in wind turbines. *IEEE Trans Control Syst Technol* 2020;29(3):1102–16.
- [11] Li J, Yu XB. LiDAR technology for wind energy potential assessment: demonstration and validation at a site around Lake Erie. *Energ Convers Manage* 2017;144:252–61.
- [12] Sharma PK, Warudkar V, Ahmed S. Application of lidar and measure correlate predict method in offshore wind resource assessments. *J Clean Prod* 2019;215:534–43.
- [13] Dolatabadi A, Abdeltawab H, Mohamed YARI. Deep spatial-temporal 2-D CNN-BLSTM model for ultrashort-term LiDAR-assisted wind turbine's power and fatigue load forecasting. *IEEE Trans on Industr Inform* 2021;18(4):2342–53.
- [14] Zhang J, Zhao X. Spatiotemporal wind field prediction based on physics-informed deep learning and LiDAR measurements. *Appl Energy* 2021;288:116641.
- [15] Wagner R, Courtney MS, Pedersen TF, Davoust S. Uncertainty of power curve measurement with a two-beam nacelle-mounted lidar. *Wind Energy* 2016;19(7):1269–87.
- [16] Towers P, Jones BL. Real-time wind field reconstruction from LiDAR measurements using dynamic wind model and state estimation. *Wind Energy* 2016;19(1):133–50.
- [17] Mercieca J, Aram P, Jones BL, Kadirkamanathan V. A spatiotemporal estimation framework for real-world lidar wind speed measurements. *IEEE Trans Control Syst Technol* 2019;28(4):1595–602.
- [18] Borracino A, Schlipf D, Haizmann F, Wagner R. Wind field reconstruction from nacelle-mounted lidar short-range measurements. *Wind Energy Sci* 2017;2(1):269–83.
- [19] Sun S, Liu S, Liu J, et al. Wind field reconstruction using inverse process with optimal sensor placement. *IEEE Trans Sustain Energy* 2018;10(3):1290–9.
- [20] 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.
- [21] Baydin AG, Pearlmutter BA, Radul AA, Siskind JM. Automatic differentiation in machine learning: a survey. *J Mach Learn Res* 2018;18:1–43.
- [22] Nazari LF, Camponogara E, Seman LO. Physics-informed neural networks for modeling water flows in a river channel. *IEEE Trans Artif Intell* 2022. <https://doi.org/10.1109/TAI.2022.3200028>. Early access. [Online].

- [23] Cheng C, Zhang GT. Deep learning method based on physics informed neural network with resnet block for solving fluid flow problems. *Water* 2021;13(4):423.
- [24] Aliakbari M, Mahmoudi M, Vadasz P, Arzani A. Predicting high-fidelity multiphysics data from low-fidelity fluid flow and transport solvers using physics-informed neural networks. *Int J Heat Fluid Flow* 2022;96:109002.
- [25] Cai S, Mao Z, Wang Z, et al. Physics-informed neural networks (PINNs) for fluid mechanics: a review. *Acta Mech Sin* 2021;37(12):1727–38.
- [26] Huang B, Wang J. Applications of physics-informed neural networks in power systems-a review. *IEEE Trans Power Syst* 2022;38(1):572–88.
- [27] Son S, Lee H, Jeong D, et al. A novel physics-informed neural network for modeling electromagnetism of a permanent magnet synchronous motor. *Adv Eng Inform* 2023;57:102035.
- [28] Mao Z, Jagtap AD, Karniadakis GE. Physics-informed neural networks for high-speed flows. *Comput Methods Appl Mech Eng* 2020;360:112789.
- [29] Jin X, Cai S, Li H, et al. NSFnets (Navier-Stokes flow nets): physics-informed neural networks for the incompressible Navier-stokes equations. *J Comput Phys* 2021; 426:109951.
- [30] Khan A, Lowther DA. Physics informed neural networks for electromagnetic analysis. *IEEE Trans Magn* 2022;58(9):1–4.
- [31] Baldan M, Di Barba P, Lowther DA. Physics-informed neural networks for inverse electromagnetic problems. *IEEE Trans Magn* 2023;59(5):1–5.
- [32] Mohammadian M, Baker K, Fioretto F. Gradient-enhanced physics-informed neural networks for power systems operational support. *Electr Pow Syst Res* 2023;223: 109551.
- [33] Zhang W, Li J. CPINNs: a coupled physics-informed neural networks for the closed-loop geothermal system. *Comput Math Appl* 2023;132:161–79.
- [34] Wang J, Wang Y, Gao X, et al. Solution of tidal equations and inversion of bottom friction coefficient based on neural network. In: Proceedings of OCEANS 2022. IEEE; 2022. p. 1–4.
- [35] Zhang C, Shafeezadeh A. Nested physics-informed neural network for analysis of transient flows in natural gas pipelines. *Eng Appl Artif Intell* 2023;122:106073.
- [36] Zhang J, Zhao X. Three-dimensional spatiotemporal wind field reconstruction based on physics-informed deep learning. *Appl Energy* 2021;300:117390.
- [37] Wang Y, Yuan Z, Xie C, Wang J. Temporally sparse data assimilation for the small-scale reconstruction of turbulence. *Phys Fluids* 2022;34(6):065115.
- [38] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE; 2016. p. 770–8.
- [39] He K, Zhang X, Ren S, Sun J. Identity mappings in deep residual networks. In: European conference on computer vision. Springer; 2016. p. 630–45.
- [40] Hayou S, Clerico E, He B, Deligiannidis G, Doucet A, Rousseau J. Stable resnet. In: International conference on artificial intelligence and statistics. PMLR; 2021. p. 1324–32.
- [41] Roy SK, Manna S, Song T, Bruzzone L. Attention-based adaptive spectral-spatial kernel ResNet for hyperspectral image classification. *IEEE Trans Geosci Remote Sens* 2020;59(9):7831–43.
- [42] Yang L, Zhong J, Zhang Y, Bai S, Li G, Yang Y, et al. An improving faster-RCNN with multi-attention ResNet for small target detection in intelligent autonomous transport with 6G. *IEEE Trans Intell Transp Syst* 2023;24(7):7717–25.
- [43] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: Advances in neural information processing systems 30 (NIPS2017); 2017. p. 5998–6008.
- [44] Yildiz C, Acikgoz H, Korkmaz D, Budak U. An improved residual-based convolutional neural network for very short-term wind power forecasting. *Energ Convers Manage* 2021;228:113731.
- [45] Acikgoz H, Budak U, Korkmaz D, Yildiz C. WSFNet: an efficient wind speed forecasting model using channel attention-based densely connected convolutional neural network. *Energy* 2021;233:121121.
- [46] Zhang K, Tang B, Deng L, Liu X. A hybrid attention improved ResNet based fault diagnosis method of wind turbines gearbox. *Measurement* 2021;179:109491.
- [47] Liu Z, et al. A review of progress and applications of pulsed Doppler wind LiDARs. *Remote Sens (Basel)* 2019;11(21):2522.
- [48] VAISALA. Wind cube nacelle. Accessed March 6, 2023, <https://www.vaisala.com/en/products/wind-energy-windcube-nacelle>.
- [49] ZXLiDARS. ZXTM. Accessed March 6, 2023, <https://www.zxlidars.com/wind-lidar/zx-tm>.
- [50] MOVELASER. Molas NL. Accessed March 6, 2023, <https://www.move laser.com/index/windinner2>.
- [51] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans Neural Netw Learn Syst* 1994;5(2):157–66.
- [52] Yilmaz A, Poli R. Successfully and efficiently training deep multi-layer perceptrons with logistic activation function simply requires initializing the weights with an appropriate negative mean. *Neural Netw* 2022;153:87–103.
- [53] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics; 2010. p. 249–56.
- [54] He K, Zhang X, Ren S, et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. IEEE; 2015. p. 1026–34.
- [55] Mehta D, Van Zuijlen AH, Koren B, Holierhoek JG, Bijl H. Large Eddy simulation of wind farm aerodynamics: a review. *J Wind Eng Ind Aerodyn* 2014;133:1–17.
- [56] Wang S, Teng Y, Perdikaris P. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM J Sci Comput* 2021;43(5): A3055–81.
- [57] Churchfield M, Lee S. Nwtc design codes-sowfa. URL, http://wind.nrel.gov/design_codes/simulators/SOWFA; 2012.
- [58] Yin X, Zhao X. Deep neural learning based distributed predictive control for offshore wind farm using high-fidelity LES data. *IEEE Trans Ind Electron* 2020;68(4):3251–61.
- [59] Li R, Zhang J, Zhao X. Multi-fidelity modeling of wind farm wakes based on a novel super-fidelity network. *Energ Convers Manage* 2022;270:116185.
- [60] Fleming PA, Gebraad PMO, Lee S, van Wingerden JW, Johnson K, Churchfield M, et al. Evaluating techniques for redirecting turbine wakes using SOWFA. *Renew Energy* 2014;70:211–8.
- [61] Bartl J, Mühlé F, Schottler J, Saetran L, Peink J, Adaramola M, et al. Wind tunnel experiments on wind turbine wakes in yaw: effects of inflow turbulence and shear. *Wind Energy Sci* 2018;3(1):329–43.
- [62] Jagtap AD, Kawaguchi K, Karniadakis GE. Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *J Comput Phys* 2020; 404:109136.