



Article

A Hybrid Neural Ordinary Differential Equation Based Digital Twin Modeling and Online Diagnosis for an Industrial Cooling Fan

Chao-Chung Peng * and Yi-Ho Chen

Department of Aeronautics and Astronautics, National Cheng Kung University, Tainan 70101, Taiwan; p48101500@gs.ncku.edu.tw

* Correspondence: ccpeng@mail.ncku.edu.tw

Abstract: Digital twins can reflect the dynamical behavior of the identified system, enabling self-diagnosis and prediction in the digital world to optimize the intelligent manufacturing process. One of the key benefits of digital twins is the ability to provide real-time data analysis during operation, which can monitor the condition of the system and prognose the failure. This allows manufacturers to resolve the problem before it happens. However, most digital twins are constructed using discrete-time models, which are not able to describe the dynamics of the system across different sampling frequencies. In addition, the high computational complexity due to significant memory storage and large model sizes makes digital twins challenging for online diagnosis. To overcome these issues, this paper proposes a novel structure for creating the digital twins of cooling fan systems by combining with neural ordinary differential equations and physical dynamical differential equations. Evaluated using the simulation data, the proposed structure not only shows accurate modeling results compared to other digital twins methods but also requires fewer parameters and smaller model sizes. The proposed approach has also been demonstrated using experimental data and is robust in terms of measurement noise, and it has proven to be an effective solution for online diagnosis in the intelligent manufacturing process.

Keywords: neural ODE; system identification; data-driven modeling; digital twin; diagnosis



Citation: Peng, C.-C.; Chen, Y.-H. A Hybrid Neural Ordinary Differential Equation Based Digital Twin Modeling and Online Diagnosis for an Industrial Cooling Fan. *Future Internet* **2023**, *15*, 302. <https://doi.org/10.3390/fi15090302>

Academic Editor: Michael Sheng

Received: 6 August 2023

Revised: 27 August 2023

Accepted: 29 August 2023

Published: 4 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Cooling fan systems have been widely used in various industrial fields, such as nuclear power plants [1], automobiles [2], and workstation computers [3]. They are often used to maintain temperature and ensure an optimal operation environment. Due to their low development cost, efficient heat dissipation, and air circulation, a cooling fan system is often regarded as the primary solution for environmental temperature control use [4,5]. Despite playing an important role in the manufacturing process, the rotary mechanism of the cooling fan is highly susceptible to failure, making it one of the top 10 failing components in electronic products [6]. Therefore, it is crucial to monitor the condition of the cooling fan system to prevent drastic changes in the operation environment.

To achieve accurate condition monitoring during the manufacturing process, it is necessary to create the digital twins of a system, which provide the virtual representation of the identified physical entity. According to the data from sensors, digital twins can update their state to mirror the dynamic behavior and the condition of the system in real time. By incorporating other analysis tools, the digital twin model can be used not only for condition monitoring, but also for prognosis, optimization, and control in the digital world [7]. Therefore, digital twins are also regarded as the backbone of Industry 4.0 [8–11], providing the necessary and useful information to enhance productivity and making intelligent manufacturing smarter, more efficient, and more convenient [12].

Digital twins can be created using various kinds of methods, including using physical models or data-driven approaches. Physical methods require domain knowledge of the identified system. For an industrial cooling fan, aerodynamics, blade geometry, and the control theorem need to be considered. There have been several works analyzing its dynamics from previous studies [4,5,13]. Since the digital twin is modeled using physical principles, it has better robustness and interpretation of the anomalies. However, the result may still have a large deviation between the identified systems because the physical equation is usually built based on ideal conditions. Therefore, the digital twin may not be able to reflect the dynamic behavior in the real world.

In contrast, data-driven methods directly employ feature extraction to find the dynamical pattern from the historical data. Common methods include observer/Kalman filter identification [14,15], dynamic mode decomposition (DMD) [16,17], and sparse identification of nonlinear dynamics (SINDy) [18,19]. The most well-known and commonly used method is the neural network (NN), which provides a structure that can approximate any system dynamics based on the universal approximation theorem [20]. As the NN has evolved, different layers and structures have been developed. One of the NN types is called a recurrent neural network (RNN), which is renowned for modeling sequential data. Since system dynamics signals can also be seen as time-series data, RNN has been proven to perform well in system modeling [21–23]. However, applying RNN or any other data-driven method requires a large amount of fruitful inputs/outputs to avoid unstable model predictions [24,25]. Furthermore, most of the system dynamics are better described in the continuous-time domain. These reasons lead to excessively large model sizes and memory storage usage when employing RNN for robust modeling, which may pose a challenge in applying digital twins for real-time use.

To explore an alternative way to create the digital twins, this paper proposes using neural ordinary differential equations (neural ODEs) to approximate the governing equations of the system. The concept of neural ODEs establishes a viewpoint to connect the deep learning model with differential equations [26], offering a more flexible description and reducing the need for numerous parameters for modeling system dynamics. In addition, the continuous-time structure also allows the combination of pure physical models with neural ODEs, which can make the digital twin more accurate and interpretable [27].

This paper makes significant contributions to the field of digital twin modeling and intelligent manufacturing in terms of the following. (1) The neural ODE-based digital twin can approximate the governing equations of the industrial cooling fan system. By employing neural ODEs, the digital twin can achieve smoother system evolution, reducing abnormal discontinuities, and accommodate data from sensors with varying sampling frequencies for condition monitoring. (2) The proposed digital twin framework demonstrates impressive modeling accuracy while utilizing fewer parameters. This efficiency enables real-time condition monitoring during the manufacturing process, making it a practical and viable solution for industrial applications. (3) By incorporating the model with physically informed terms, the digital twins can be closer to real dynamical behavior and be more robust to unforeseen dynamical patterns compared to pure data-driven methods.

The remainder of the paper is organized as follows: Section 2 introduces the methodology used in this paper. The previous study on neural ODE modeling is briefly explained in Section 3. In Section 4, a comparison study is conducted to assess the fitting performance of the proposed digital twins in comparison to others. Section 5 demonstrates the practical applications of the proposed digital twin, where it effectively monitors the condition of the cooling fan and accurately detects the anomalies. The paper finally concludes in Section 6.

2. Methodology

2.1. Cooling Fan System Dynamics

An industrial cooling fan system typically comprises three essential components: a driving circuit, a rotary mechanism, and fan blades. Figure 1 briefly illustrates a cooling fan system driven by a DC motor. The driving circuit of the cooling fan system can be briefly

represented by an electric circuit with a resistor and an inductor, and its dynamics can be described as

$$L \frac{di}{dt} + iR + K_e \omega = V_{in} \tag{1}$$

where L is the inductance, R is the resistance, K_e is the back emf constant, V_{in} is the input voltage, i is the current, and ω is the rotor speed. For the motor mechanism, the dynamics can be written as

$$J_m \frac{d\omega}{dt} = K_t i - B_m \omega - T_L \tag{2}$$

where J_m is the motor inertia, K_t is the torque constant, B_m is the viscous constant, and T_L is the external load. Based on the law of energy conservation, where electrical power is equal to mechanical power, it can be expressed as

$$(K_e \omega) \cdot i = (K_t i) \cdot \omega \tag{3}$$

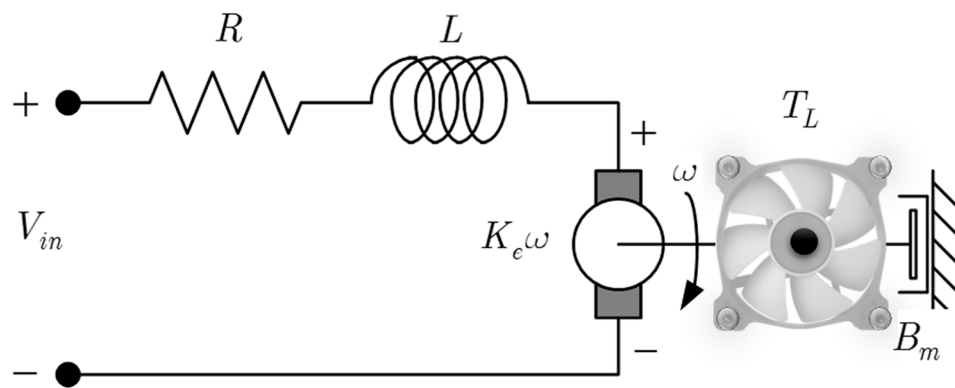


Figure 1. Illustration of a cooling fan driving system [13].

Equation (3) shows it is true only when the constants K_e and K_t have the same value, which can be defined as

$$K_e = K_t \equiv K \tag{4}$$

Substituting (4) to (1) and (2), the governing equation of the cooling fan system can be represented as

$$\begin{cases} L \frac{di}{dt} = -Ri - K\omega + V_{in} \\ J_m \frac{d\omega}{dt} = Ki - B_m \omega - T_L \end{cases} \tag{5}$$

The dynamics of the cooling fan system can be represented as a 2nd-order differential equation. Given that the response of the electrical dynamics of a cooling fan is much faster than its mechanical dynamics, the transient behavior of the electrical dynamics can be neglected. Therefore, the governing equation in (1) can be rewritten as

$$\begin{aligned} iR + K\omega &\approx V_{in} \\ \Rightarrow i &\approx \frac{V_{in} - K\omega}{R} \end{aligned} \tag{6}$$

Furthermore, according to aerodynamics, the external force on the fan blade mainly comes from the drag force, which is proportional to the square of the rotor speed. Therefore, the external load can be approximated as

$$T_L \approx C_d \omega^2 \tag{7}$$

where C_d represents the lumped aerodynamic drag coefficient. The details of the derivation can be seen in [5]. By substituting the result of (6) and (7) in the governing Equation (5), it can be expressed as

$$\Rightarrow \frac{J_m R}{K} \frac{d\omega}{dt} = V_{in} - \left(K + \frac{B_m R}{K} \right) \omega - \frac{C_d R}{K} \omega^2 \tag{8}$$

The equivalent coefficients can be defined as

$$J = \frac{J_m R}{K}, \alpha = K + \frac{B_m R}{K}, C_D = \frac{C_d R}{K} \tag{9}$$

Equation (8) can be expressed as

$$J\dot{\omega}(t) + \alpha\omega(t) + C_D\omega^2(t) = \tau(t) \tag{10}$$

After rearrangement, Equation (5) can be formed as a 1st-order dynamical differential equation. The symbol τ is denoted as the applied torque generated by the pulse-width modulator (PWM), which is proportional to the input voltage of the driving circuit; the equivalent coefficients J, α, C_D can be regarded as the lumped moment of inertia, viscous coefficient, and drag coefficient, respectively. After a few steps of simplification, the coefficients of the physical model (10) can be derived by measuring the rotor speed and applied torque, both of which are observable.

2.2. Filtering Operator Method

The simplified analytical Equation (10), defined as a “physical model”, can be utilized to predict the dynamics of the cooling fan system. To estimate these parameters, the filtering operator method is used. The details can be seen in [13]. Let analytical Equation (10) be rewritten as

$$\dot{\omega}(t) = -a\omega(t) - a_N\omega^2(t) + b\tau(t) \tag{11}$$

where

$$a = \frac{\alpha}{J}, a_N = \frac{C_D}{J}, b = \frac{1}{J} \tag{12}$$

Taking the Laplace transform of (11), the equation is represented as

$$s\Omega(s) - \omega(0) = -a\Omega(s) - a_N\Omega_N(s) + b\Gamma(s) \tag{13}$$

where

$$\Omega(s) = \mathcal{L}\{\omega(t)\}, \Omega_N(s) = \mathcal{L}\{\omega^2(t)\}, \Gamma(s) = \mathcal{L}\{\tau(t)\} \tag{14}$$

Next, multiply $1/(s + \lambda)$ on both sides of (13), where λ is a positive constant. The result can be written as

$$\Omega(s) + Y_1(s) = -a\Phi_\omega^f(s) - a_N\Phi_{\omega^2}^f(s) + b\Phi_\tau^f(s) \tag{15}$$

where

$$Y_1(s) = -\frac{s\Omega(s) + \omega(0)}{s + \lambda}, \Phi_\omega^f(s) = \frac{\Omega(s)}{s + \lambda}, \Phi_{\omega^2}^f(s) = \frac{\Omega_N(s)}{s + \lambda}, \Phi_\tau^f(s) = \frac{\Gamma(s)}{s + \lambda} \tag{16}$$

The inverse Laplace of $Y_1(s), \Phi_\omega^f(s), \Phi_{\omega^2}^f(s), \Phi_\tau^f(s)$ in (15) can be calculated by integrating the measurements

$$\begin{aligned}
 Y_1(s) = -\frac{\Omega(s)+\omega(0)}{s+\lambda} &\Rightarrow \begin{cases} \dot{y}_1(t) = -\lambda y_1(t) - \lambda\omega(t) \\ y_1(0) = -\omega(0) \end{cases} \\
 \Phi_\omega^f(s) = \frac{\Omega(s)}{s+\lambda} &\Rightarrow \begin{cases} \dot{f}_\omega(t) = -\lambda f_\omega + \omega(t) \\ f_\omega(0) = 0 \end{cases} \\
 \Phi_{\omega^2}^f(s) = \frac{\Omega_N(s)}{s+\lambda} &\Rightarrow \begin{cases} \dot{f}_{\omega^2}(t) = -\lambda f_{\omega^2} + \omega^2(t) \\ f_{\omega^2}(0) = 0 \end{cases} \\
 \Phi_\tau^f(s) = \frac{\Gamma(s)}{s+\lambda} &\Rightarrow \begin{cases} \dot{f}_\tau(t) = -\lambda f_\tau + \tau(t) \\ f_\tau(0) = 0 \end{cases}
 \end{aligned} \tag{17}$$

After integration, the parameters can be estimated by using the least-square method

$$\underbrace{\begin{bmatrix} \omega(0) + y_1(0) \\ \omega(\Delta t) + y_1(\Delta t) \\ \vdots \\ \omega(n\Delta t) + y_1(n\Delta t) \end{bmatrix}}_{\mathbf{Y}} = \underbrace{\begin{bmatrix} -\phi_\omega(0) & -\phi_{\omega^2}(0) & \phi_\tau(0) \\ -\phi_\omega(\Delta t) & -\phi_{\omega^2}(\Delta t) & \phi_\tau(\Delta t) \\ \vdots & \vdots & \vdots \\ -\phi_\omega(n\Delta t) & -\phi_{\omega^2}(n\Delta t) & \phi_\tau(n\Delta t) \end{bmatrix}}_{\mathbf{\Phi}} \underbrace{\begin{bmatrix} a \\ a_N \\ b \end{bmatrix}}_{\boldsymbol{\theta}} \tag{18}$$

$$\hat{\boldsymbol{\theta}} = \begin{bmatrix} \hat{a} \\ \hat{a}_N \\ \hat{b} \end{bmatrix} = (\mathbf{\Phi}^T \mathbf{\Phi})^{-1} \mathbf{\Phi}^T \mathbf{Y} \tag{19}$$

$$\hat{f} = 1/\hat{b}, \hat{\alpha} = \hat{a}\hat{f}, C_D = \hat{a}_N\hat{f} \tag{20}$$

2.3. Recurrent Neural Network

RNN is one type of NN with a loop structure, which can model the patterns with time-dependent behavior and approximate the transient response of the system. One type of RNN, the nonlinear autoregressive network with exogenous inputs (NARX), is used in this paper for the comparison study since it shows outstanding performance in modeling different system dynamics [28–30]. In this paper, the NARX model is constructed and trained by the Neural Net Time Series toolbox in MATLAB. Given the input and output of the system, $\mathbf{x}(k) \in \mathbb{R}^p$ and $\mathbf{y}(k) \in \mathbb{R}^r$, the structure of NARX is illustrated in Figure 2, which can also be expressed in the following mathematical form

$$\mathbf{y}(k) = \mathbf{B}_o + \mathbf{W}_o \cdot \tan sig \left(\begin{matrix} \mathbf{B}_i + \mathbf{W}_i \mathbf{X}_D(k) \\ + \mathbf{W}_l \mathbf{Y}_D(k) \end{matrix} \right) \tag{21}$$

where

$$\mathbf{X}_D(k) = [\mathbf{x}^T(k) \quad \mathbf{x}^T(k-1) \quad \dots \quad \mathbf{x}^T(k-d)]^T \tag{22}$$

$$\mathbf{Y}_D(k) = [\mathbf{y}^T(k-1) \quad \mathbf{y}^T(k-1) \quad \dots \quad \mathbf{y}^T(k-d)]^T \tag{23}$$

$$\tan sig(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{24}$$

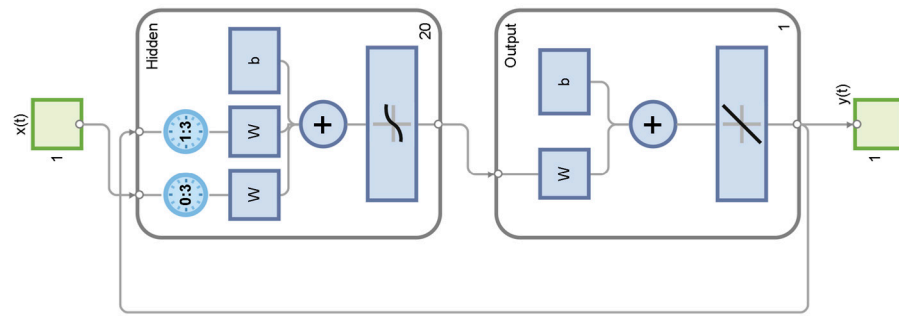


Figure 2. The structure of the NARX model.

The variable d represents the order of the model. The weights and bias of the NARX model ($\mathbf{W}_i \in \mathbb{R}^{n \times (p+1)d}$, $\mathbf{W}_l \in \mathbb{R}^{n \times rd}$, $\mathbf{W}_o \in \mathbb{R}^{d \times n}$, $\mathbf{B}_i \in \mathbb{R}^{n \times 1}$, $\mathbf{B}_o \in \mathbb{R}^{r \times 1}$) are trained using the measured data and updated using the Levenberg–Marquardt algorithm.

2.4. Neural Ordinary Differential Equation

The core concept of a neural ODE is to provide a residual neural network (ResNet) architecture to describe a continuous-time dynamical behavior. Suppose the state of a system at the k^{th} sampling time is denoted as $\mathbf{h}(k)$; the dynamics can be described by ResNet as

$$\mathbf{h}(k + 1) = \mathbf{h}(k) + f(\mathbf{h}(k), \theta) \tag{25}$$

where $f(\mathbf{h}(k), \theta)$ is a nonlinear function modeled by a neural network and parameterized by θ . The iterative update equation in (25) can be seen as a Euclidean discretization result for a continuous-time model [31]. The output of the neural network can be regarded as the time derivative of the state when the time step Δt approaches zero:

$$\frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t), \theta) \tag{26}$$

where t represents the continuous time stamp. In a continuous-time structure, the update equation can be written as

$$\mathbf{h}(t) = \mathbf{h}(0) + \int_0^t f(\mathbf{h}(t'), \theta) dt' \tag{27}$$

The integral can be calculated by any kind of ODE solver. In this paper, the 4th order Runge–Kutta (RK4) method is applied for numerical integration because of its conventional usage. As the discrete time step becomes smaller and a more precise ODE solver is used for discretization, the neural network model can approximate a differential equation closely. Modeling fan dynamics using neural ODE has been applied in the previous study [25]. Instead of only depending on the initial value, such as (27), the state of the cooling fan system is also affected by the forced input. Therefore, the neural ODE for the cooling fan system is defined as

$$\begin{aligned} \dot{\omega}(t) &= f(\omega(t), \tau(t), \theta) = \mathbf{W}_o \tanh(\mathbf{W}_i \cdot [\omega(t) \quad \tau(t)]^T + \mathbf{b}_i) + b_o \\ \omega(t) &= \omega(0) + \int_0^t f(\omega(t'), \tau(t'), \theta) dt' \end{aligned} \tag{28}$$

where $\mathbf{W}_i \in \mathbb{R}^{n \times 2}$, $\mathbf{W}_o \in \mathbb{R}^{1 \times n}$ are weights, and $\mathbf{b}_i \in \mathbb{R}^{n \times 1}$, $b_o \in \mathbb{R}$ are biases, of the neural network. The neural network used in this paper only contains one hidden layer, which is displayed in Figure 3. The inputs of the neural network include the state of the system, the rotor speed, and the applied torque. The parameters $\theta = \{\mathbf{W}_i, \mathbf{W}_o, \mathbf{b}_i, b_o\}$ can be estimated

by applying gradient-based optimization to reduce the value of the loss function. In this paper, the loss function is set as the mean absolute error (MAE).

$$L = \frac{1}{N} \sum_{k=1}^N |\omega(t) - \hat{\omega}(t)| \tag{29}$$

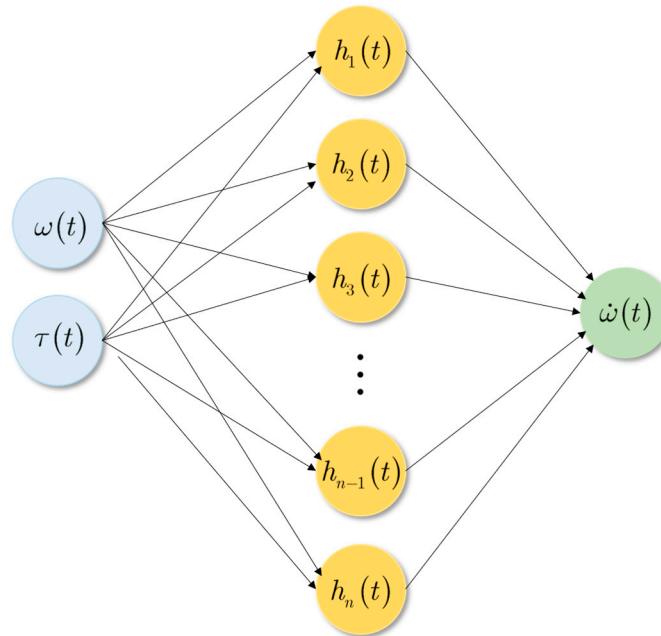


Figure 3. Structure of the neural ODE model for the cooling fan system.

After calculating the loss, the gradient-based optimization updates the parameters using the following equation

$$\theta^{(i+1)} = \theta^{(i)} - \alpha \nabla_{\theta} L \tag{30}$$

The subscript (*i*) represents the number of iterations. α is the learning rate, which is adjusted based on specific applications and optimization methods used. In this paper, ADAM optimization is applied to train the parameters of the neural network.

2.5. Hybrid Neural Ordinary Differential Equation

The architecture of the neural ODE makes it possible to combine with the physically based ODE. Therefore, this paper proposes a hybrid neural ODE model to create the digital twin of the system that contains both physics-informed terms and a neural network in one differential equation, which can be represented as

$$\dot{\omega}(t) = \frac{1}{J} \left[-\alpha \omega(t) - C_D \omega^2(t) + \tau(t) \right] + f(\omega(t), \tau(t), \theta) \tag{31}$$

The differential equation of the hybrid neural ODE model is the combination of Equations (10) and (28). The training process is displayed in Figure 4. The physical parameters J, α, C_D are initially estimated using filtering operator method and kept fixed while using gradient-based optimization to train the overall model. The hybrid architecture can be regarded as using physics-informed terms to describe the linear dynamic behavior and using neural network to fit the nonlinearity and uncertainties that occur in the real world. This not only increases the fitting accuracy but also enhances the interpretability of the model.

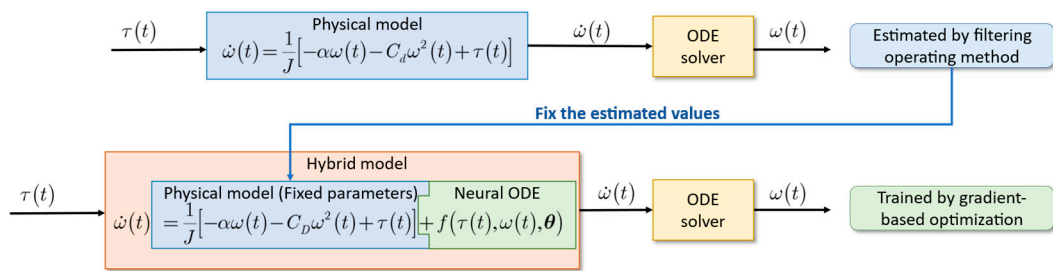


Figure 4. The training process of the hybrid neural ODE model.

3. Literature Review

The previous study in [25] has already utilized neural ODE to model a motor-driven propeller dynamical system via simulation. In this research, the fitting performance of the neural ODE model is compared with a physical model and a NARX model, which represent physically based and discrete-time data-driven methods, respectively. Furthermore, NARX models of the 1st order and 10th order are both applied to highlight that the neural ODE model can be closer to real system dynamics. The system dynamics is simulated using MATLAB/Simulink with Equation (10), employing RK4 as the solver with a numerical time step of 5×10^{-4} s. The measurement sampling frequency is set to 0.01 s and measurement noise follows normal distribution with a mean of 0 and a standard deviation of $\sqrt{10}$. The reference parameters are $J = 0.15, \alpha = 0.012, C_D = 0.0025$. Figures 5 and 6 show the simulation data for training and testing. The fitting results of different models are displayed in Figure 7, which shows the neural ODE model has a better fitting performance and generalization compared to others. Therefore, the previous study has proved that the continuous structure in the neural network can not only enhance the modeling accuracy but also reduce the model’s complexity. To further improve the feasibility and interpretability, this paper combines neural ODE with physics-informed terms to investigate its fitting performance on real-world data.

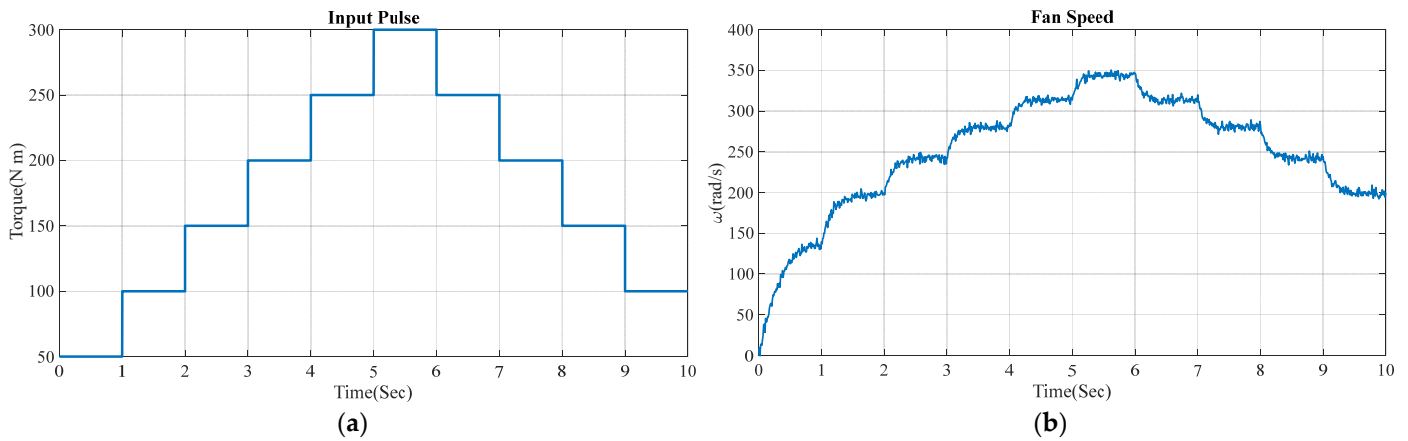


Figure 5. Simulation data for training. (a) Excitation input torque; (b) the response of fan speed.

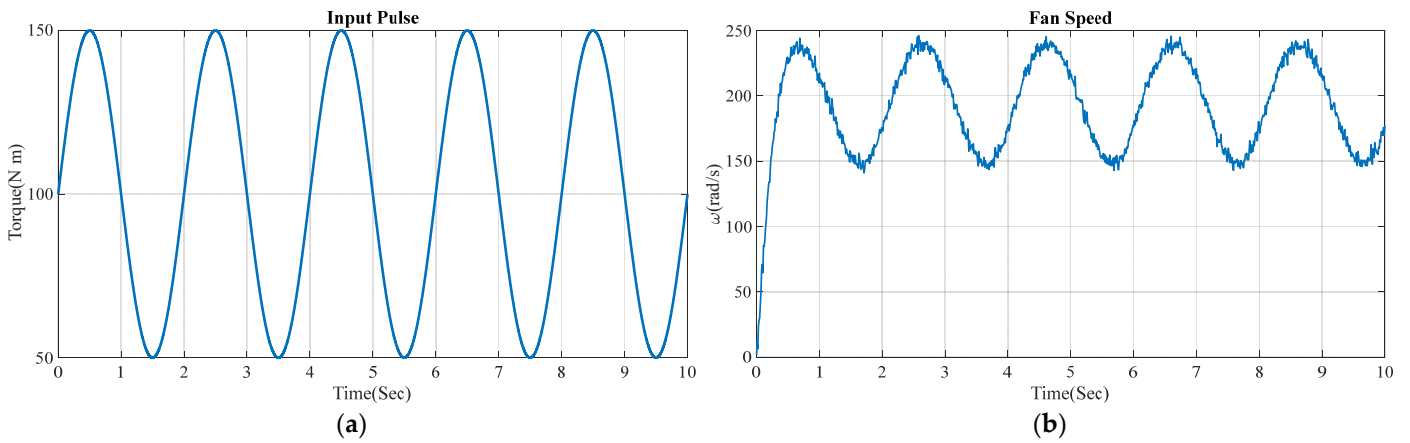


Figure 6. Simulation data for testing. (a) Excitation input torque; (b) the response of fan speed.

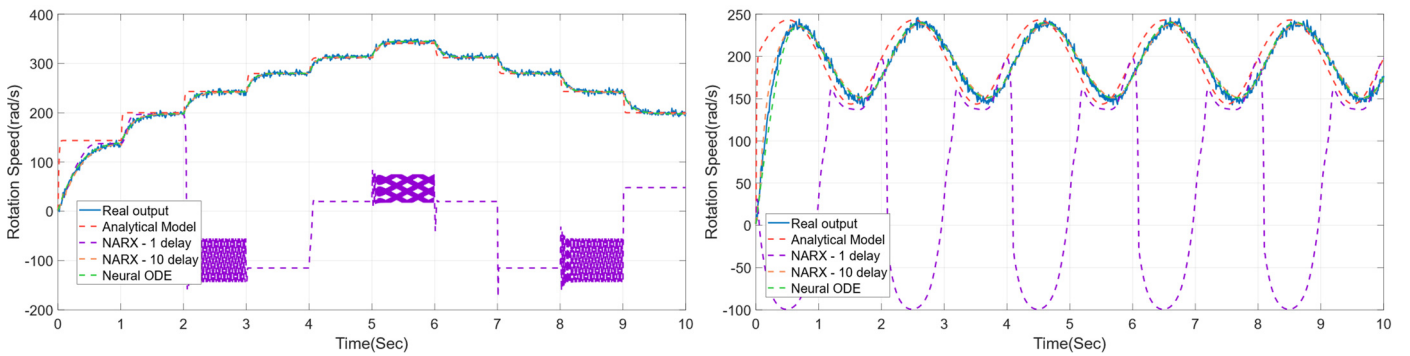


Figure 7. The fitting performance of different fan dynamics models in [25].

4. Digital Twins Derivation

4.1. Problem Formulation

The study of physically based digital twins of a cooling fan system and a filtering operator method has been given in [4,5,13]. The results have shown that the model has great performance and uses very few parameters. Figure 8 displays the fitting results on experimental data, but it is also evident that there is a decline in the accuracy of fan speed prediction as the fan speed decreases. The result indicates that the physics employed in the model is only suitable for high-speed conditions. However, there are several uncertainties that need to be considered when determining the low-speed dynamics. To address these uncertain conditions, data-driven methods are utilized in this paper to improve the model’s accuracy. In this section, different types of neural network-based models are applied and compared with physically based models. The model that can best deal with these uncertainties and provide the best-fitting results will be selected and chosen as the digital twin for the cooling fan system.

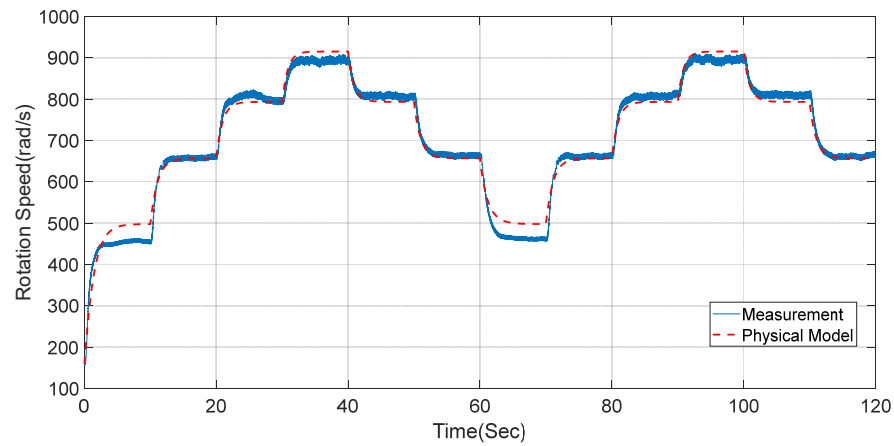


Figure 8. Physically based model fitting performance in high-speed conditions.

4.2. Structure of Digital Twins

This paper utilizes and compares four types of different methods to construct the digital twins of the cooling fan system and evaluate each modeling performance to find the best. Four types of models are the physical model, the NARX model, the neural ODE model, and the hybrid neural ODE model. The models’ structure and number of parameters can be seen in Table 1. All the neural network models only include a single hidden layer. Although increasing the number of hidden layers may improve fitting results, it also presents challenges in real-time implementation with limited memory storage. In addition, the paper only sets applied torque and previous fan speed as the inputs of the digital twins. Since the proposed ODE model is designed to describe the dynamics of the cooling fan, the inputs of the model also have to follow the physical laws governing the identified system. Although increasing more input features may enhance the fitting performance, it runs the risk of reducing the model’s ability to generalize the fan dynamics effectively. Furthermore, according to the practical scenario, fan speed and applied torque are the only available measurements from the experimental device. These are the main reasons why this research only selected these key parameters that have physical meanings as the inputs of the model. The directly measurable information not only reduces the model’s complexity but also ensures the interpretability of the digital twin modeling. In this study, two sets of data are used for training and testing: training data are used to estimate the parameters of the digital twins, and testing data are utilized to examine the fitting result.

Table 1. The details of four types of cooling fan digital twins.

Model Type	Parameter Estimation Method	Structure	No. of Parameters	Details
Physical model	Filtering integral operator	$\dot{\omega}(t) = \frac{1}{J} [-\alpha\omega(t) - C_D\omega^2(t) + \tau(t)]$ $\omega(t) = \omega(0) + \int_0^t \dot{\omega}(t') dt'$	3	–
NARX model	Levenberg–Marquardt algorithm	$\omega(k) = \mathbf{B}_o + \mathbf{W}_o \cdot \tan sig \left(\begin{matrix} \mathbf{B}_i + \mathbf{W}_i \mathbf{X}_D(k) \\ + \mathbf{W}_l \mathbf{Y}_D(k) \end{matrix} \right)$	2051	10th order with 50 hidden states
Neural ODE	ADAM optimization	$\dot{\omega}(t) = f(\omega(t), \tau(t), \theta) = \mathbf{W}_o \tanh(\mathbf{W}_i \cdot [\omega(t) \quad \tau(t)]^T + \mathbf{b}_i) + b_o$ $\omega(t) = \omega(0) + \int_0^t f(\omega(t'), \tau(t'), \theta) dt'$	81	1st order with 20 hidden states
Hybrid neural ODE	ADAM optimization	$\dot{\omega}(t) = \frac{1}{J} [-\alpha\omega(t) - C_D\omega^2(t) + \tau(t)] + f(\omega(t), \tau(t), \theta)$ $\omega(t) = \omega(0) + \int_0^t \dot{\omega}(t') dt'$	34	1st order with 10 hidden states for NN part

4.3. Numerical Simulation

Before the experiment, this study first evaluates the modeling performance of digital twins by using simulation data. The simulation environment and data patterns of fan dynamics are the same as mentioned in Section 3. To evaluate the fitting performance, three criteria are used: root-mean-square error (RMSE), maximum error (Error max), and R-squared (R^2), which are calculated as follows

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \tag{32}$$

$$Error\ max = \max_i |y_i - \hat{y}_i| \tag{33}$$

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y})^2}{\sum_{i=1}^N (y_i - \bar{y})^2}, \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i \tag{34}$$

where $\{y_i\}_{i=1}^N$ is a set of measurements and $\{\hat{y}_i\}_{i=1}^N$ represents the model output. A model with a good fitting result tends to have less RMSE, less Error max, and higher R^2 .

The fitting result can be seen in Figures 9 and 10, and Table 2. Since the physical model has the same dynamic equation as the simulation, the fitting performance is good. Furthermore, the estimated parameters for the physical model are $\hat{J} = 0.1444$, $\hat{\alpha} = 0.0167$, and $\hat{C}_D = 0.0025$, which is close to the reference parameters values. On the other hand, the pure data-driven models, NARX model, and neural ODE model also perform well on training and testing data, but the fitting results do not surpass those of the physical model. It is noted that both show similar fitting performance, while neural ODE uses much fewer parameters. It is proved that the continuous-time model structure has advantages in modeling the system dynamics. Finally, the hybrid neural ODE model uses the same physical parameters \hat{J} , $\hat{\alpha}$, \hat{C}_D while training. The fitting result shows a slight improvement compared to the physical model, which means that the fitting performance on simulation data still accounts for physics-informed terms.

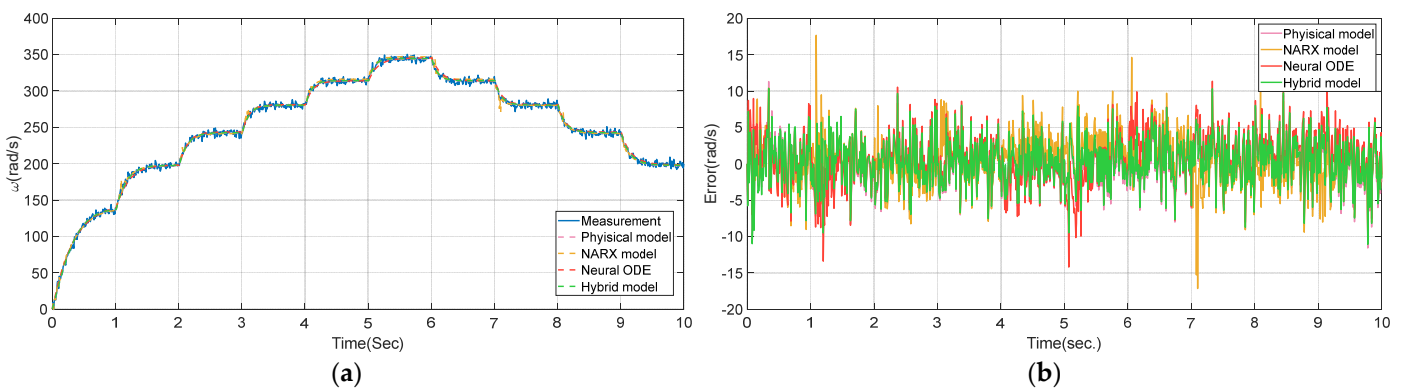


Figure 9. Training fitting result for simulation data. (a) Fan speed output. (b) Modeling error.

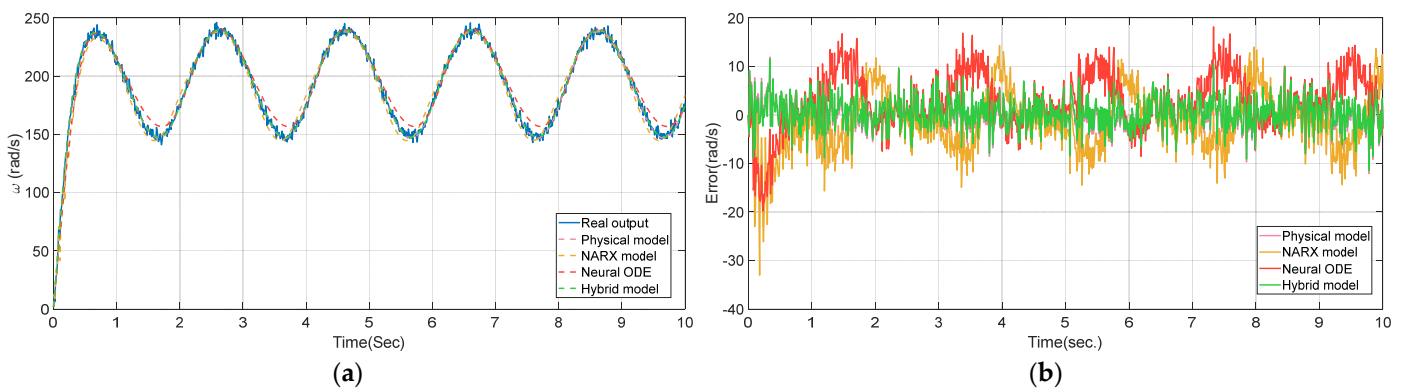


Figure 10. Testing fitting results for simulation data. (a) Fan speed output. (b) Modeling error.

Table 2. The overall fitting result on simulation data.

Model Type	Training Data			Testing Data		
	RMSE	Error Max	R ²	RMSE	Error Max	R ²
Physical model	3.227	11.563	0.9979	3.219	12.078	0.9925
NARX model	3.633	17.663	0.9973	6.313	33.009	0.9712
Neural ODE	3.631	14.169	0.9973	6.532	19.823	0.9692
Hybrid neural ODE	3.232	11.062	0.9979	3.194	11.521	0.9926

4.4. Experiment Validation

To validate the practical application of the proposed digital twins in a real fan system, this paper collects data from an actual cooling fan system, which can be seen in Figure 11. The fan tray system consists of eight cooling fans with eight knobs for speed control. A Cortex-M4 microprocessor is used to give the input command. Figures 12 and 13 are the training and testing data, respectively. Since the applied torque is assumed to be proportional to the input voltage, the voltage signal is directly used for system identification.

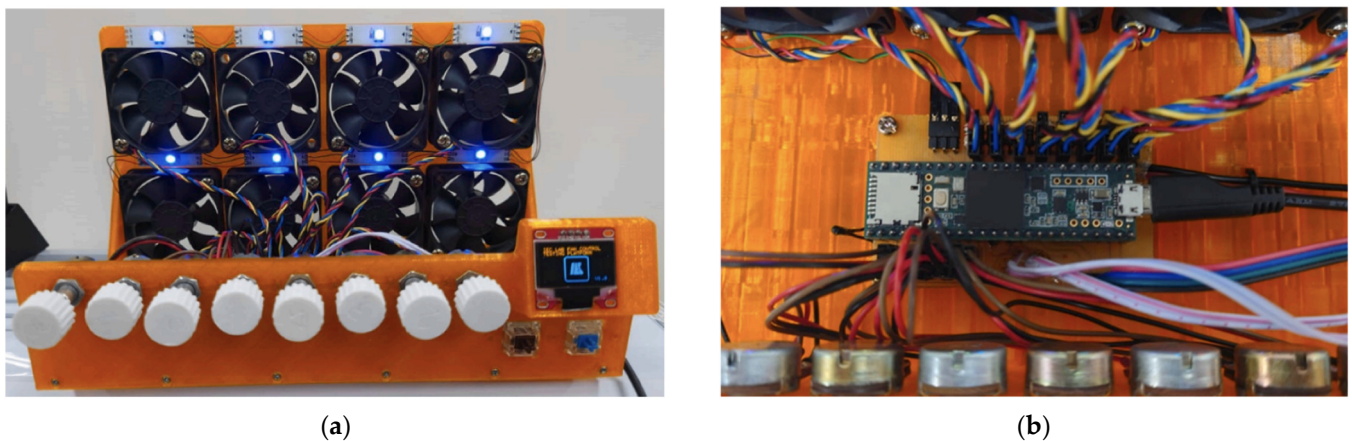


Figure 11. Developed fan tray system used for the experiment [13]. (a) Front view of the fan system; (b) microprocessor for input excitation.

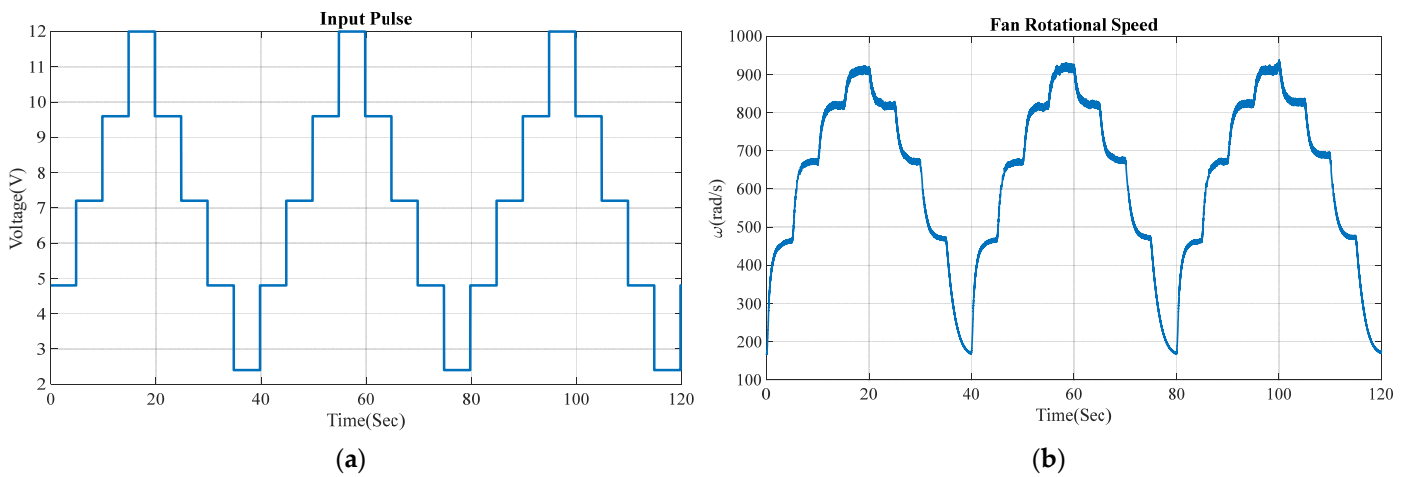


Figure 12. Experimental data for training. (a) Excitation input voltage. (b) The response of fan speed.

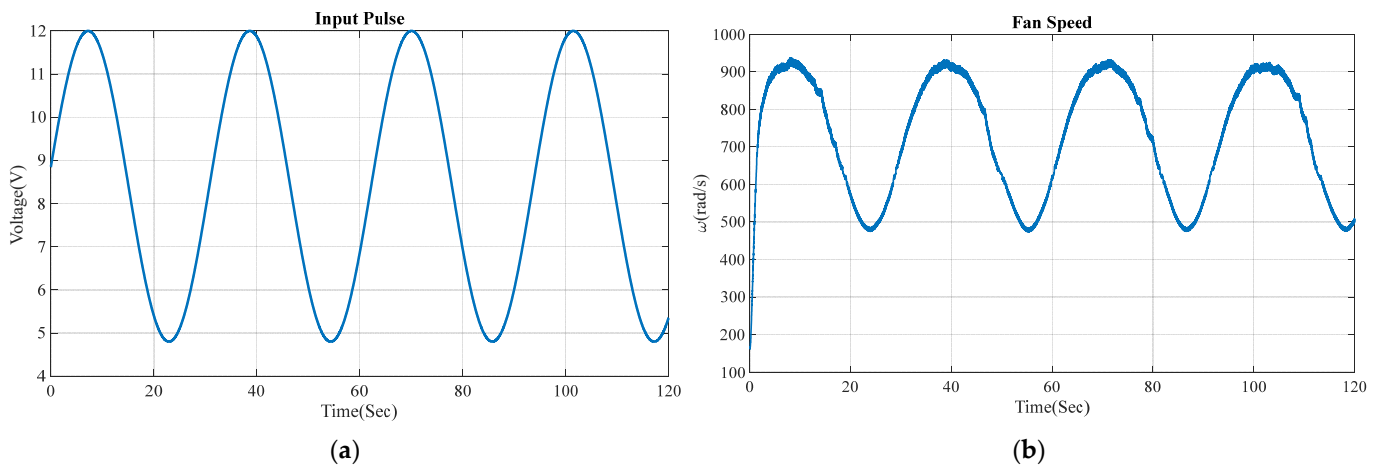


Figure 13. Experimental data for testing. (a) Excitation input voltage. (b) The response of fan speed.

The fitting results are shown in Figures 14–16, and Table 3. From the results shown in Figure 14, it is more obvious that the prediction of the fan speed from the physical model has a low accuracy in training data. The complexity of the low-speed fan dynamics makes it challenging to describe accurately using just a physical model, especially in transient behavior. The estimated parameter values are $\hat{f} = 0.0148$, $\hat{\alpha} = 0.0074$, $\hat{C}_D = 5.4772 \times 10^{-6}$. For the NARX model, although the fitting results on training data are good, the model has an overfitting problem on testing data, which is shown in Figure 16. It can be inferred that the discrete-time model has difficulty capturing the continuous dynamics of the system, making it susceptible to unforeseen data. In contrast, the neural ODE model can achieve a better fitting performance than the NARX model with fewer parameters. The outstanding capability of system identification for continuous-time neural networks is again proven. In addition, the fitting result of the neural ODE model in Figure 14 performs better than the physical model at a low speed. This indicates that the data-driven method is effective in addressing low-speed dynamics. The hybrid neural ODE model, which only sets 10 hidden states, has the best-fitting performance among all models. This shows that the inclusion of physics-informed terms can efficiently reduce the complexity of the NN required to fit the data. From another perspective, it can also be said that the physical model can enhance its fitting accuracy by incorporating a neural ODE structure, which makes it able to describe the nonlinearity of the real dynamics. According to the fitting results on both simulation and experimental data, it is proven that the hybrid neural ODE model has the best fitting performance and can be the digital twins of a healthy cooling fan system.

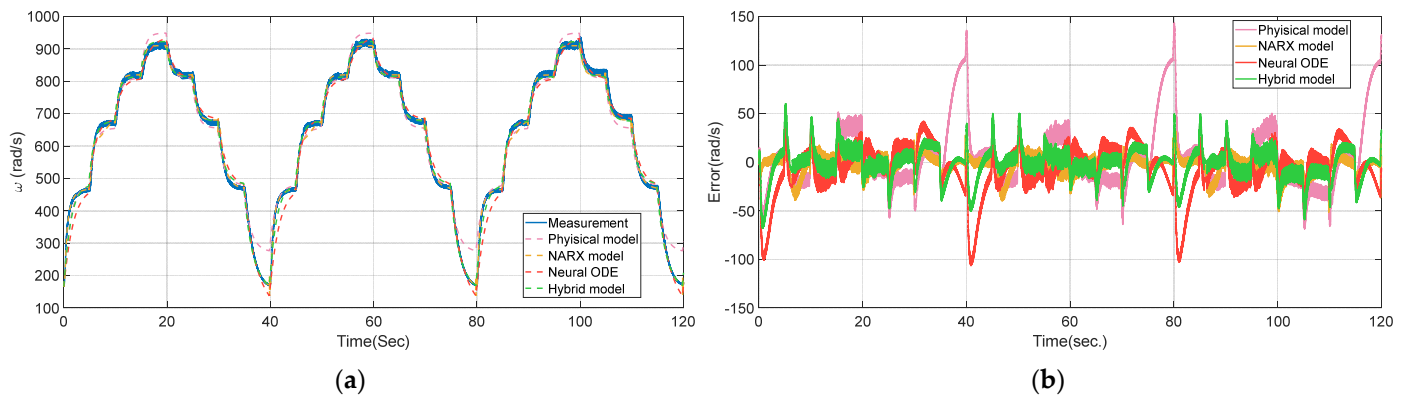


Figure 14. Training fitting result on experimental data. (a) Fan speed output. (b) Modeling error.

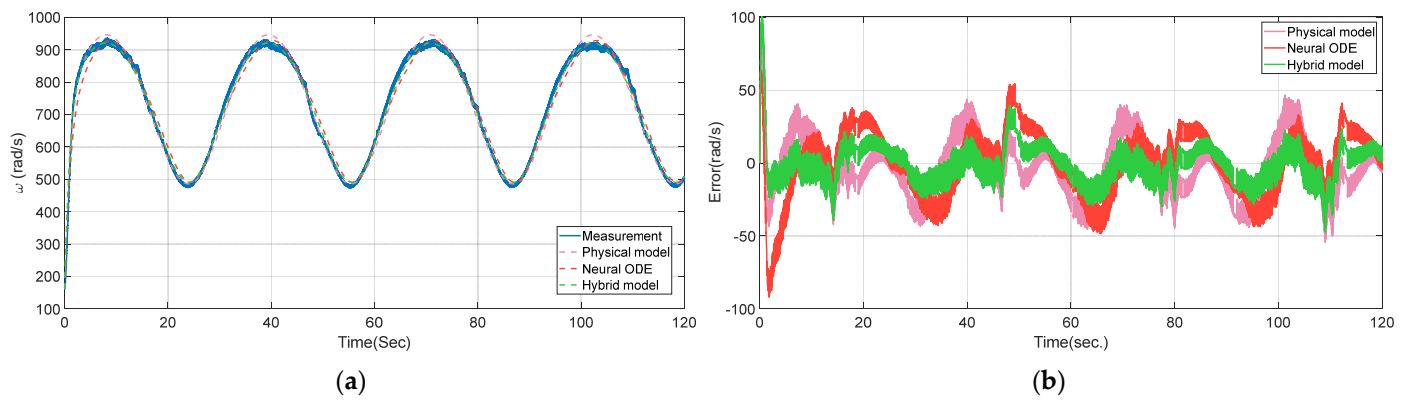


Figure 15. Testing fitting results on experimental data. (a) Fan speed output. (b) Modeling error.

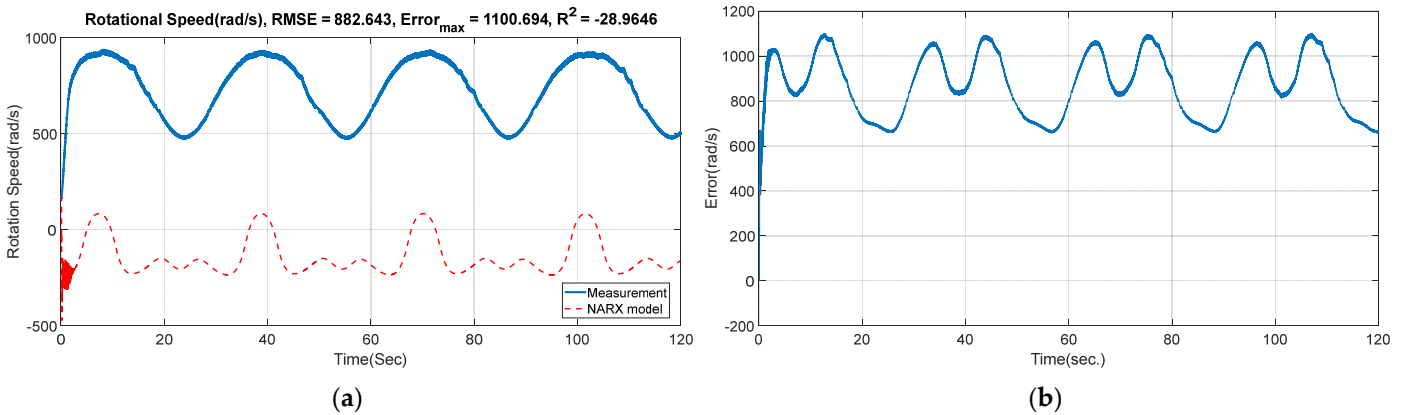


Figure 16. Testing fitting result for NARX model on experimental data. (a) Fan speed output. (b) Modeling error.

Table 3. The overall fitting results based on experimental data.

Model Type	Training Data			Testing Data		
	RMSE	Error Max	R ²	RMSE	Error Max	R ²
Physical model	35.051	143.455	0.9734	20.761	83.704	0.9834
NARX model	11.458	51.635	0.9972	882.643	1100.964	-28.9646
Neural ODE	26.236	106.411	0.9851	23.239	91.775	0.9792
Hybrid model	17.081	67.99	0.9937	14.582	100.685	0.9918

5. Anomaly Detection Result

In this section, the proposed digital twin hybrid neural ODE model is applied to monitor the condition and detect the anomalies of the cooling fan system. The study uses the fan tray system seen in Figure 11 and simulates two kinds of anomalies, an inlet covered by an object and a disturbance occurring on the rotor, which are shown in Figure 17. The proposed digital twin first evaluates the fan speed response in a healthy condition and assess the cooling fan when an anomaly happens.

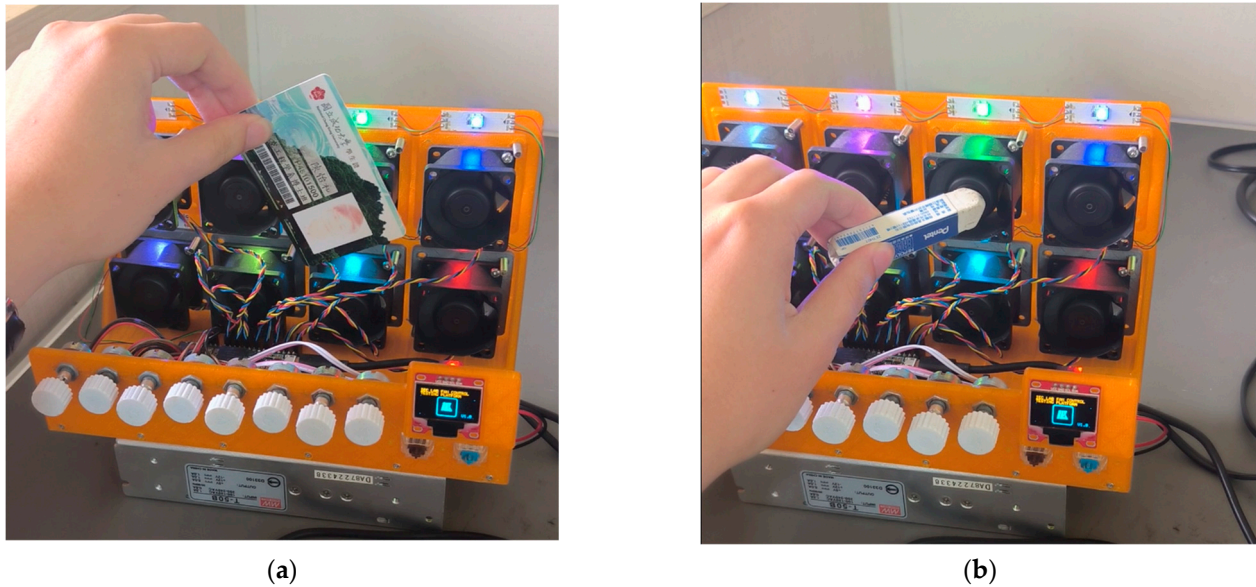


Figure 17. The anomalies that occurred in the fan system. (a) Fan inlet covered by an object. (b) Disturbance of the rotor.

To inspect the status of a cooling fan system through a digital twin, it is essential to establish a reliable mechanism to determine its health status. Since the digital twin represents the response of a healthy cooling fan system, the status can be defined by the error between the digital twin. Figure 18 illustrates the error distribution between the hybrid neural ODE model and measurement under healthy conditions, which is the error seen in Figures 14b and 15b. The error distribution can be regarded as a normal distribution, in which over 95% of values fall within two standard deviations. Based on the characteristic, the “safe range” that determines the system under healthy conditions can be established by an upper bound (UB) and a lower bound (LB):

$$UB = \hat{\omega}(t) + 2\sigma \quad (35)$$

$$LB = \hat{\omega}(t) - 2\sigma \quad (36)$$

where $\hat{\omega}(t)$ represents the digital twins’ output calculated by the hybrid neural ODE model and σ is the standard deviation. Calculation shows that the standard deviation of the error distribution is $\sigma = 15.8483$. According to the above definition, a cooling fan system is considered healthy when its fan speed remains within the upper and lower bound described in Equations (35) and (36). Conversely, if the fan speed exceeds the safe range, it indicates an anomaly in the fan system.

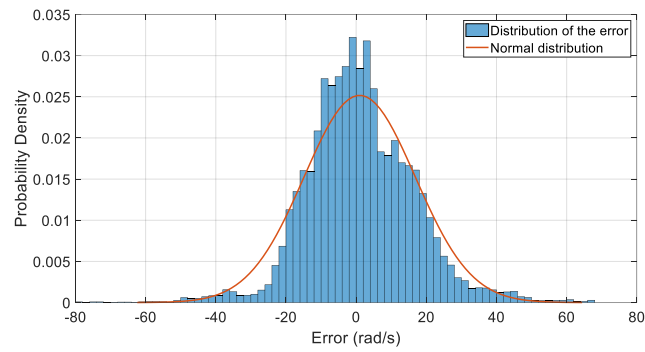


Figure 18. Error distribution between the digital twin and measurement under healthy conditions.

After establishing the criterion for a healthy condition, the anomaly detection process can be initiated. Figure 19 presents the results by using anomaly detection based on the proposed digital-twin-based anomaly detection. The green area in the figures represents the safe range defined by digital twin outputs and the standard deviation of the error in healthy conditions, while the red area indicates the occurrence of the anomalies. Most of the time, the fan speed remains within the green area, representing that the fan system is under healthy conditions. However, when anomalies do occur, the fan speed experiences sudden changes and exceeds the safe range. The anomalies are detected using the defined safe range criteria. Once the objects are removed, which means the anomalies are addressed, the fan speed quickly returns to the safe range, representing that the system restores the normal condition. Figure 19a,b demonstrate the anomaly conditions when the inlet was covered and when there was a disturbance on the rotor. Figure 19c shows the monitoring result when the speed command was generated manually, reflecting the applicability in unpredictable commands. The experiments demonstrate accurate detection when faults occur and rapid response to anomalies. Moreover, the digital twin has significantly fewer parameters than a conventional NN, facilitating real-time implementation with limited memory storage. These results validate the practical application capabilities of the proposed digital twin.

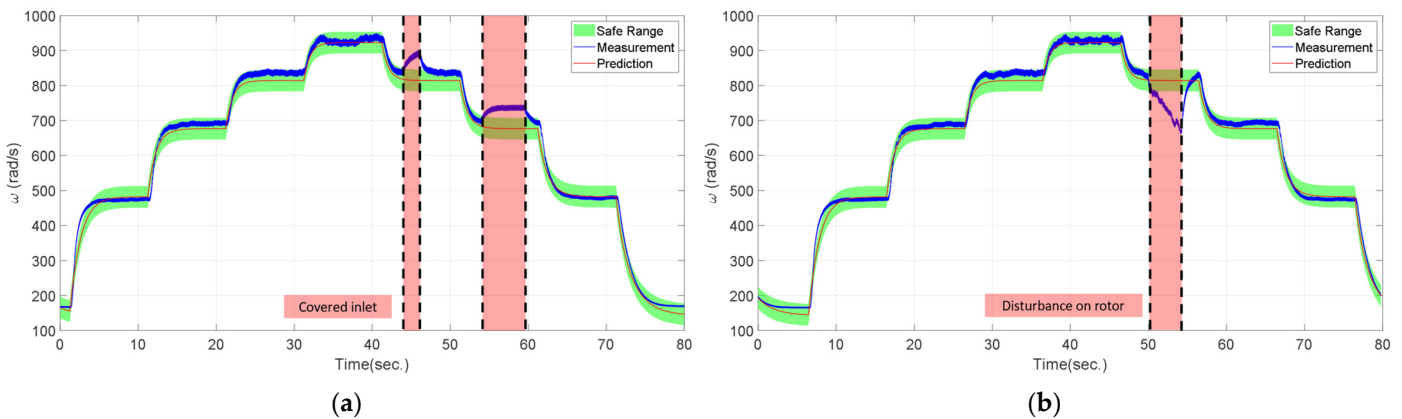


Figure 19. Cont.

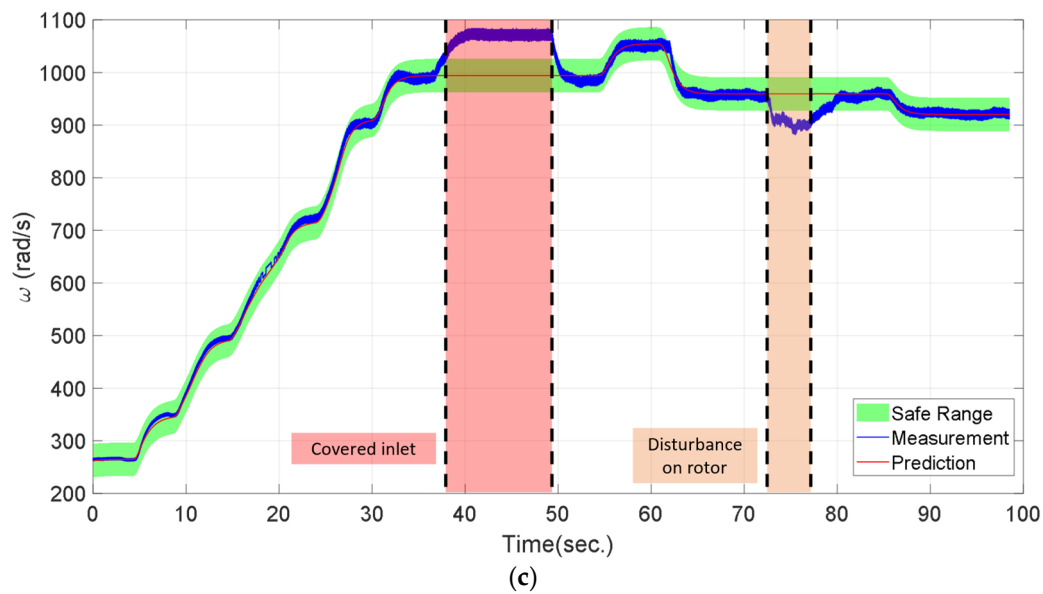


Figure 19. Anomaly detection result. (a) Covered inlet. (b) Disturbance on the rotor. (c) Manual modulation with anomalies.

6. Conclusions

This paper introduces a novel method to construct a digital twin of a cooling fan system by combining physically based and data-driven models. Based on the continuous-time structure, neural ODE can incorporate the physics-informed terms derived from the governing equation of the cooling fan. A comparison study shows that the proposed hybrid digital twin model can have outstanding fitting performances with the usage of fewer parameters. Furthermore, the fitting results on experimental data reveal that the neural ODE part of the hybrid neural ODE model can address the uncertainties and nonlinear behaviors of the real fan dynamics, enhancing both the modeling accuracy and interpretability. To validate its capability in practical applications, the proposed digital twin is employed to establish an anomaly-detection process. Represented as the fan speed under healthy conditions, the digital twin's output is utilized to estimate the status of a fan system based on the error between the digital twin's output and measurements. A fan system is identified as faulty when its fan speed overpasses the safe range defined by the digital twin output and the standard deviation of the error distribution of model output and measurement under healthy conditions. The conducted experiments simulate anomalies, and the results demonstrate that the proposed digital twin-based anomaly detection effectively responds to the faults, thereby validating the feasibility of the study presented in this paper. The study shows the outstanding modeling performance of a hybrid neural ODE. By establishing a precise relationship between the system's inputs and outputs through continuous evolution, neural ODE is proven to be suitable for the mathematical framework used in control theory, such as Lyapunov stability analysis. Therefore, future study will focus on utilizing neural ODE in control applications, thereby enabling a more intelligent and robust manufacturing process.

Author Contributions: Conceptualization, C.-C.P.; methodology, C.-C.P. and Y.-H.C.; software, Y.-H.C.; validation, C.-C.P. and Y.-H.C.; formal analysis, Y.-H.C.; investigation, C.-C.P. and Y.-H.C.; resources, C.-C.P.; data curation, Y.-H.C.; writing—original draft preparation, Y.-H.C.; writing—review and editing, C.-C.P.; visualization, Y.-H.C.; supervision, C.-C.P.; project administration, C.-C.P.; funding acquisition, C.-C.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Science and Technology under Grant No. MOST 111-2221-E-006-170.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author, upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Guillen, D.P.; Anderson, N.; Krome, C.; Boza, R.; Griffel, L.; Zouabe, J.; Al Rashdan, A. A RELAP5-3D/LSTM model for the analysis of drywell cooling fan failure. *Prog. Nucl. Energy* **2020**, *130*, 103540. [[CrossRef](#)]
2. Liu, C.; Wang, Z.; Fan, C.; Zhang, R.; Man, X. A Joint Control Strategy for Automobile Active Grille Shutter and Cooling Fan. *Int. J. Automot. Technol.* **2021**, *22*, 1675–1682. [[CrossRef](#)]
3. Wiryasart, S.; Hommalee, C.; Naphon, P. Thermal cooling enhancement of dual processors computer with thermoelectric air cooler module. *Case Stud. Therm. Eng.* **2019**, *14*, 100445. [[CrossRef](#)]
4. Peng, C.-C.; Su, C.-Y. Modeling and parameter identification of a cooling fan for online monitoring. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–14. [[CrossRef](#)]
5. Peng, C.-C.; Lin, Y.-I. Dynamics modeling and parameter identification of a cooling fan system. In Proceedings of the 2018 IEEE International Conference on Advanced Manufacturing (ICAM), Yunlin, Taiwan, 16–18 November 2018; pp. 257–260.
6. Schroeder, B.; Gibson, G.A. Understanding disk failure rates: What does an MTTF of 1,000,000 hours mean to you? *ACM Trans. Storage (TOS)* **2007**, *3*, 8-es. [[CrossRef](#)]
7. Tao, F.; Xiao, B.; Qi, Q.; Cheng, J.; Ji, P. Digital twin modeling. *J. Manuf. Syst.* **2022**, *64*, 372–389. [[CrossRef](#)]
8. Jiang, Y.; Yin, S.; Li, K.; Luo, H.; Kaynak, O. Industrial applications of digital twins. *Philos. Trans. R. Soc. A* **2021**, *379*, 20200360. [[CrossRef](#)] [[PubMed](#)]
9. Colombo, A.W.; Karnouskos, S.; Kaynak, O.; Shi, Y.; Yin, S. Industrial cyberphysical systems: A backbone of the fourth industrial revolution. *IEEE Ind. Electron. Mag.* **2017**, *11*, 6–16. [[CrossRef](#)]
10. Melesse, T.Y.; Di Pasquale, V.; Riemma, S. Digital twin models in industrial operations: A systematic literature review. *Procedia Manuf.* **2020**, *42*, 267–272. [[CrossRef](#)]
11. Jones, D.; Snider, C.; Nassehi, A.; Yon, J.; Hicks, B. Characterising the Digital Twin: A systematic literature review. *CIRP J. Manuf. Sci. Technol.* **2020**, *29*, 36–52. [[CrossRef](#)]
12. He, B.; Bai, K.-J. Digital twin-based sustainable intelligent manufacturing: A review. *Adv. Manuf.* **2021**, *9*, 1–21. [[CrossRef](#)]
13. Peng, C.-C.; Chen, T.-Y. A recursive low-pass filtering method for a commercial cooling fan tray parameter online estimation with measurement noise. *Measurement* **2022**, *205*, 112193. [[CrossRef](#)]
14. Prakash, N.P.S.; Chen, Z.; Horowitz, R. System identification in multi-actuator hard disk drives with colored noises using observer/Kalman filter identification (OKID) framework. *arXiv* **2021**, arXiv:2109.12460.
15. Fan, L.; Liu, X.; Cai, G.-p. Dynamic modeling and modal parameters identification of satellite with large-scale membrane antenna. *Adv. Space Res.* **2019**, *63*, 4046–4057. [[CrossRef](#)]
16. Mauroy, A.; Goncalves, J. Koopman-based lifting techniques for nonlinear systems identification. *IEEE Trans. Autom. Control* **2019**, *65*, 2550–2565. [[CrossRef](#)]
17. Erichson, N.B.; Brunton, S.L.; Kutz, J.N. Compressed dynamic mode decomposition for background modeling. *J. Real-Time Image Process.* **2019**, *16*, 1479–1492. [[CrossRef](#)]
18. Brunton, S.L.; Proctor, J.L.; Kutz, J.N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. USA* **2016**, *113*, 3932–3937. [[CrossRef](#)] [[PubMed](#)]
19. Fukami, K.; Murata, T.; Zhang, K.; Fukagata, K. Sparse identification of nonlinear dynamics with low-dimensionalized flow representations. *J. Fluid Mech.* **2021**, *926*, A10. [[CrossRef](#)]
20. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366. [[CrossRef](#)]
21. Wen, S.; Wang, Y.; Tang, Y.; Xu, Y.; Li, P.; Zhao, T. Real-time identification of power fluctuations based on LSTM recurrent neural network: A case study on Singapore power system. *IEEE Trans. Ind. Inform.* **2019**, *15*, 5266–5275. [[CrossRef](#)]
22. Jiao, M.; Wang, D.; Qiu, J. A GRU-RNN based momentum optimized algorithm for SOC estimation. *J. Power Sources* **2020**, *459*, 228051. [[CrossRef](#)]
23. Kim, B.-H.; Pyun, J.-Y. ECG identification for personal authentication using LSTM-based deep recurrent neural networks. *Sensors* **2020**, *20*, 3069. [[CrossRef](#)] [[PubMed](#)]
24. Peng, C.-C.; Chen, Y.-H. Digital twins-based online monitoring of TFE-731 turbofan engine using Fast orthogonal search. *IEEE Syst. J.* **2021**, *16*, 3060–3071. [[CrossRef](#)]
25. Peng, C.-C.; Chen, Y.-H. Modeling of a Motor-driven Propeller Dynamics System by Neural Ordinary Differential Equation. In Proceedings of the 2023 Sixth International Symposium on Computer, Consumer and Control (IS3C), Taichung, Taiwan, 3–30 July 2023; pp. 284–287.
26. Chen, R.T.; Rubanova, Y.; Bettencourt, J.; Duvenaud, D.K. Neural ordinary differential equations. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 6572–6583. [[CrossRef](#)]
27. Lai, Z.; Mylonas, C.; Nagarajiah, S.; Chatzi, E. Structural identification with physics-informed neural ordinary differential equations. *J. Sound Vib.* **2021**, *508*, 116196. [[CrossRef](#)]

28. Di Nunno, F.; Granata, F. Groundwater level prediction in Apulia region (Southern Italy) using NARX neural network. *Environ. Res.* **2020**, *190*, 110062. [[CrossRef](#)] [[PubMed](#)]
29. Takyi-Aninakwa, P.; Wang, S.; Zhang, H.; Xiao, Y.; Fernandez, C. A NARX network optimized with an adaptive weighted square-root cubature Kalman filter for the dynamic state of charge estimation of lithium-ion batteries. *J. Energy Storage* **2023**, *68*, 107728. [[CrossRef](#)]
30. Shahbaz, M.; Taqvi, S.A.A.; Inayat, M.; Inayat, A.; Sulaiman, S.A.; McKay, G.; Al-Ansari, T. Air catalytic biomass (PKS) gasification in a fixed-bed downdraft gasifier using waste bottom ash as catalyst with NARX neural network modelling. *Comput. Chem. Eng.* **2020**, *142*, 107048. [[CrossRef](#)]
31. Lu, Y.; Zhong, A.; Li, Q.; Dong, B. Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 3276–3285.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.