Wind Turbine Fault-Tolerant Control via Incremental Model-Based Reinforcement Learning

Jingjie Xie, Hongyang Dong, Xiaowei Zhao, and Shuyue Lin

***Abstract*—A reinforcement learning (RL) based fault-tolerant control strategy is developed in this paper for wind turbine torque & pitch control under actuator & sensor faults subject to unknown system models. An incremental model-based heuristic dynamic programming (IHDP) approach, along with a critic-actor structure, is designed to enable fault-tolerance capability and achieve optimal control. Particularly, an incremental model is embedded in the critic-actor structure to quickly learn the potential system changes, such as faults, in real-time. Different from the current IHDP methods that need the intensive evaluation of the state and input matrices, only the input matrix of the incremental model is dynamically evaluated and updated by an online recursive least square estimation procedure in our proposed method. Such a design significantly enhances the online model evaluation efficiency and control performance, especially under faulty conditions. In addition, a value function and a target critic network are incorporated into the main critic-actor structure to improve our method's learning effectiveness. Case studies for wind turbines under various working conditions are conducted based on the fatigue, aerodynamics, structures, and turbulence (FAST) simulator to demonstrate the proposed method's solid fault-tolerance capability and adaptability.**

***Note to Practitioners*—This work achieves high-performance wind turbine control under unknown actuator & sensor faults. Such a task is still an open problem due to the complexity of turbine dynamics and potential uncertainties in practical situations. A novel data-driven and model-free control strategy based on reinforcement learning is proposed to handle these issues. The designed method can quickly capture the potential changes in the system and adjust its control policy in real-time, rendering strong adaptability and fault-tolerant abilities. It provides data-driven innovations for complex operational tasks of wind turbines and demonstrates the feasibility of applying reinforcement learning to handle fault-tolerant control problems. The proposed method has a generic structure and has the potential to be implemented in other renewable energy systems.**

***Index Terms*—Fault-tolerant control, Reinforcement learning, Wind turbine control, Intelligent control.**

I. Introduction

MODERN wind turbines are becoming large-scale and highly complicated to generate more power and meet diverse requirements. Due to the complex operation environment, multiple perturbations, and long-term harsh working conditions (especially for far offshore wind turbines), undesirable failures in sophisticated wind turbines will inevitably occur in practice. The main risks associated with wind turbine systems include actuator/sensor faults, measuring deviations, unknown dynamics, and external disturbances, which may lead to severe control performance degradation, power capturing efficiency reduction, and even structural damage. The demands for improving the reliability and safety of wind turbine control systems are significant and urgent.

To this end, fault-tolerant control (FTC) methods have emerged and received considerable attention in recent decades. FTC aims to compensate for the effects of undesirable failures and maintain the control performance at an acceptable level under faults [1], [2]. In general, it can be categorized into active FTC and passive FTC [3]. Active FTC methods react actively to failures by monitoring the system with a fault detection and diagnosis (FDD) mechanism [4] to reconfigure control systems for potential performance recovery with the remaining healthy components. They could provide the strong capability to handle serious faults but suffer from high design complexity and relatively poor real-time adaptability. In contrast, passive FTC methods are usually designed to maintain control performance under fault conditions by robust and/or adaptive control techniques without directly employing FDD [5]. Compared to active FTC, passive FTC has a more flexible and simple structure, less computational complexity, and is easier to implement. Passive FTC can instantaneously counteract many typical failures without the requirement to accurately detect faults as in active FTC methods.

Due to its self-healing and reconfigurable performance for control recovery under various faulty conditions, the FTC methods have also been applied in wind turbine operations. For example, Ref. [6] designed two wind turbine FTC methods – a passive FTC scheme based on a fuzzy model reference adaptive control and an active FTC with FDD methods, achieving superior fault-tolerance ability under the torque actuator offset faults. To particularly compensate for the pitch actuator faults of the wind turbine pitch control, Ref. [7] proposed an adaptive sliding mode observer-based active FTC method, and Ref. [8] developed an adaptive output feedback sliding mode control. Ref. [9] introduced a linear quadratic regulator-based wind turbine individual pitch control strategy combined with an active FTC for compensating different types of pitch sensor faults. Additionally, an adaptive fractional-based terminal back-stepping sliding mode method in Ref. [10] and an extended state observer-based active FTC method in Ref. [11]

This work has received funding from the UK Supergen Offshore Renewable Energy Hub ECR Research Fund and the UK Engineering and Physical Sciences Research Council under grant number EP/Y016297/1. J. Xie is with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong, China. Email: jingjie.xie@polyu.edu.hk. She was with the Intelligent Control & Smart Energy (ICSE) Research Group, School of Engineering, University of Warwick, Coventry CV4 7AL, UK. H. Dong and X. Zhao (Corresponding Author) are with the Intelligent Control & Smart Energy (ICSE) Research Group, School of Engineering, University of Warwick, Coventry CV4 7AL, UK. Emails: hongyang.dong@warwick.ac.uk, xiaowei.zhao@warwick.ac.uk. Shuyue Lin is with the Department of Engineering, University of Hull, Hull, HU6 7RX, UK. Email: S.Lin@hull.ac.uk.

were developed for wind turbine FTC control considering the actuator and sensor faults. For better understanding, Table I summarizes the properties, types of faults, and control methods to offer a comprehensive overview of several presentative wind turbine FTC technologies.

The above-mentioned wind turbine FTC methods have shown effectiveness in handling actuator and sensor faults, it is noteworthy that the problem is still challenging. Specifically, the strong fault tolerance and robustness for both active and passive FTC methods usually require much more accurate and richer system information to better understand faulty conditions. However, such expectations are challenging especially for the active FTC since the FDD for fault compensation is error-prone under unknown and harsh working conditions. As a result, more precise FDD requires much more model knowledge and computational resources with complex and tedious fault estimation structures, which is not preferable in practice. Another key concern of FTC for wind turbines is the optimality of control performance in faulty conditions, which makes the mainstream FTC methods incapable because most of those conventional FTC methods lack optimality consideration and thus would bring high costs and degraded transient performance. To our knowledge, there remains a research gap in achieving optimal fault-tolerant control for wind turbines with reduced model dependency. These bottlenecks motivate us to integrate the state-of-the-art reinforcement learning (RL) technique into the FTC framework for wind turbines. Because the RL is a powerful tool to learn optimal control policies for grey-box or black-box systems [12], [13], [14]. Moreover, an RL agent can interact with the environment and analyze the measured data in-depth to capture key system information instead of relying on accurate analytical models. Note that the RL-based FTC methods have shown advantages in various fields [15], such as unmanned aerial vehicles [16], highly flexible aircraft [17], and mobile robots [18]. To be brief, it is still an open area to design RL-based optimal FTC for wind turbines. Particularly, an improved RL-based FTC method with enhanced learning efficiency is urgently needed for wind turbine operation in complex and unknown environments.

Motivated by these features, this paper aims to design a novel RL-based fault-tolerant controller for wind turbines under actuator and sensor faults, uncertainties, and disturbances. Specifically, an incremental model-based heuristic dynamic programming (IHDP) method is designed for wind turbine control. Our strategy is data-driven and model-free since it does not rely on any pre-determined analytical wind turbine models. Instead, the essential system information can be captured by the critic network and also a specially-designed incremental model, which is updated online with real-time measurements. The incremental model can quickly capture the real-time changes of the system, rendering adaptability and fault-tolerant abilities to the controller. Based on the incremental model, a critic-actor structure is utilized to solve an optimal wind farm control problem to minimize long-term tracking errors. Particularly, the critic is to approximate the long-term performance metric built by tracking errors, and the actor is to learn optimal torque and pitch control strategies for wind turbines. The high-fidelity FAST (Fatigue, Aerodynamics, Structures, and Turbulence) simulator developed by the National Renewable Energy Laboratory (NREL) [19] is employed to validate the effectiveness of the proposed method. The main contributions of this paper are summarized as follows.

(1) In this paper, a novel incremental model-based RL method is proposed for wind turbine fault-tolerant control. It differs from mainstream conventional active wind turbine FTC methods that are highly dependent on accurate FDD techniques and analytical wind turbine dynamics. The proposed RL-based FTC for wind turbines is passive in that it not only relaxes the requirement on complex FDD procedures but also is data-driven and model-free to achieve optimal control performance and strong fault tolerance simultaneously without accurate analytical system dynamics.

(2) Different from the current methods that need to update the whole system dynamics, the proposed algorithm only approximates the partial system dynamics in an incremental domain, i.e., the input matrix of the incremental model in real-time. Such a design significantly enhances the online model evaluation efficiency and reliability with a rapid approximation process & a simplified design structure. Notably, the employed incremental model for wind turbine FTC applications can further result in reduced perturbation in each sampling step and thus will possess solid robustness and adaptability against uncertainties, faults, and disturbances.

The rest of this paper is organized as follows. The wind turbine control problem under actuator and sensor faults is formulated in Section II. The proposed RL-based FTC design is presented in Section III. The performance of the proposed method is validated using the FAST wind turbine simulator in Section IV. Finally, conclusions are given in Section V.

II. Problem Formulation

The wind turbine torque and pitch control problems are formulated in this section, followed by a description of actuator and sensor faults.

Without loss of generality, the dynamics of commonly used wind turbines can be described by:

$$\dot{\boldsymbol{s}} = f(\boldsymbol{s}) + G(\boldsymbol{s})\boldsymbol{a} \tag{1}$$

where $\boldsymbol{s} = [\theta \quad w_r \quad w_g \quad T_g \quad \beta]^T$ is the state vector, θ, w_r, w_g, T_g, and β represent the torsion angle, rotor speed, generator speed, generator torque, and pitch angle, respectively. Moreover, $\boldsymbol{a} = [T_{g,ref} \quad \beta_{ref}]^T$ is the control input, and here $T_{g,ref}$ and β_{ref} denote the control signals of generator torque and pitch angle, respectively. It is noteworthy that functions/matrices f and G can vary for different wind turbines, and their specific analytical expressions are not required in our control system design. Generally, wind turbine control tasks aim to extract as much power as possible in Region II via torque control by adjusting the rotor/generator speed to track the optimal reference, and maintain the power generation and rotor/generator speed at their rated level in Region III through pitch control [20]. These wind turbine control tasks can be

TABLE I
SUMMARY OF EXISTING STUDIES ABOUT WIND TURBINE FTC METHODS

Ref.	Fault types	Control types	Methods	FTC properties
[6]	Torque actuator fault	Torque & pitch control	Fuzzy model reference adaptive control with model identification	Passive & active FTC
[7]	Pitch actuator fault	Pitch control	Adaptive sliding mode estimation algorithm	Active FTC
[8]	Actuator fault	Pitch control	Adaptive output feedback sliding mode control	Active FTC
[9]	Pitch sensor fault	Pitch control	Linear quadratic regulator	Active FTC
[10]	Actuator and sensor faults	Pitch control	Adaptive fractional-based terminal back-stepping sliding mode control	Active FTC
[11]	Sensor and actuator faults	Torque & pitch control	Extended state observer based active FTC method	Active FTC

regarded as the optimal tracking control problems for a long-term reward function that can be described as follows:

$$\boldsymbol{a}^* = \arg\max \sum_{t}^{\infty} [(\boldsymbol{s} - \boldsymbol{s}^*)^T Q_W (\boldsymbol{s} - \boldsymbol{s}^*)] \tag{2}$$

subject to

$$w_r \leq 1.1 w_r^{\text{rated}} \tag{3}$$

$$w_g \leq 1.1 w_g^{\text{rated}} \tag{4}$$

$$T_g^{\min} \leq T_{g,ref} \leq T_g^{\max} \tag{5}$$

$$\beta^{\min} \leq \beta_{ref} \leq \beta^{\max} \tag{6}$$

$$\Delta T_g^{min} \leq \Delta T_{g,ref} \leq \Delta T_g^{\max} \tag{7}$$

$$\Delta\beta^{\min} \leq \Delta\beta_{ref} \leq \Delta\beta^{max} \tag{8}$$

where $\boldsymbol{s}^*$ is the desired state, $\boldsymbol{a}^*$ denotes the optimal control input, and Q_W is a weighting matrix. Eqs. (3) and (4) represent state constraints, and w_r^{rated} and w_g^{rated} denote the rated rotor speed and generator speed, respectively. Eqs. (5), (6), (7), and (8) are limitations of control inputs, where T_g^{min}, T_g^{max}, $\beta^{\min}$, and $\beta^{\max}$ are the lower & upper bounds of the generator torque and pitch angle, respectively. $\Delta T_{g,ref}$ and $\Delta\beta_{ref}$ denote the changing rates of generator torque and pitch angle, respectively. ΔT_g^{min}, ΔT_g^{max}, $\Delta\beta^{\min}$, and $\Delta\beta^{\max}$ are the lower & upper bounds of the changing rates of torque and pitch angle, respectively.

The above formulation describes a standard wind turbine control problem. However, due to harsh working environments and inevitable fatigue, wind turbines may face actuator and sensor faults, leading to significantly degraded control performance and reduced power production.

One of the most common wind turbine faults is the torque actuator fault, which can be caused by the failure in converter/generator electronics or an offset on the torque estimation as a result of design or manufacturing problems [6]. This failure will result in offset values of the generator torque. It can be modeled as:

$$T_{g,ref}^{\mathcal{F}} = T_{g,ref}^* + \mathcal{F}_{T_{g,ref}} \tag{9}$$

where $T_{g,ref}^{\mathcal{F}}$ is the total torque under the actuator fault, $T_{g,ref}^*$ denotes the torque under the healthy case, and the term $\mathcal{F}_{T_{g,ref}}$ represents the offset part.

Another type of fault considered in this paper is the sensor fault, such as the generator speed measuring failure, which usually originates from the electrical and mechanical breakdown [21]. This type of failure can behave either as the stuck fault that the output of the sensor is a fixed value or as a partial failure that a gain factor acts on the sensor measurements [9]:

$$\beta_{ref}^{\mathcal{F}} = \begin{cases} \mathcal{F}_\beta & Stuck \quad fault \\ \mathcal{F}_{\beta_{ref}} \beta_{ref}^* & Partial \quad failure \end{cases} \tag{10}$$

where $\beta_{ref}^{\mathcal{F}}$ is the pitch angle under the pitch sensor fault, β_{ref}^* denotes the nominal pitch angle, $\mathcal{F}_\beta$ is a constant fixed value in a stuck fault, and $\mathcal{F}_{\beta_{ref}}$ is the partial failure factor.

Therefore, the wind turbine dynamics in Eq. (1) under faults, uncertainties, and disturbances can be described by:

$$\begin{aligned} \dot{\boldsymbol{s}} = & f(\boldsymbol{s}) + (f_{\mathcal{F}}(\boldsymbol{s}) - f(\boldsymbol{s}))\delta_{\mathcal{F}} + \Delta f(\boldsymbol{s}) \\ & + (G(\boldsymbol{s}) + (G_{\mathcal{F}}(\boldsymbol{s}) - G(\boldsymbol{s}))\delta_{\mathcal{F}} + \Delta G(\boldsymbol{s}))\,\boldsymbol{a} + d \end{aligned} \tag{11}$$

where $f_{\mathcal{F}}(\boldsymbol{s})$ and $G_{\mathcal{F}}(\boldsymbol{s})$ are nominal forms of $f(\boldsymbol{s})$ and $G(\boldsymbol{s})$ under actuator or sensor faults, $\Delta f(\boldsymbol{s})$ and $\Delta G(\boldsymbol{s})$ are unmodelled dynamics, and d denotes disturbances. In addition, $\delta_{\mathcal{F}}$ is a binary constant (to be either 0 or 1) that is employed to denote the existence of faults. If $\delta_{\mathcal{F}} = 0$, Eq. (11) represents the healthy case. If $\delta_{\mathcal{F}} = 1$, Eq. (11) stands for faulty cases.

Fig. 1 provides a qualitative illustration of the wind turbine structure under potential faults. The faults, uncertainties, and disturbances altogether can significantly affect the power generation process, degrade the control performance, and eventually deteriorate the wind turbine life cycle. An RL-based FTC method will be proposed to solve this issue in the following section to achieve and maintain optimal control performance under faulty conditions.

III. RL-BASED WIND TURBINE FTC SCHEME

This section develops a novel RL-based optimal fault-tolerant control method for wind turbines.

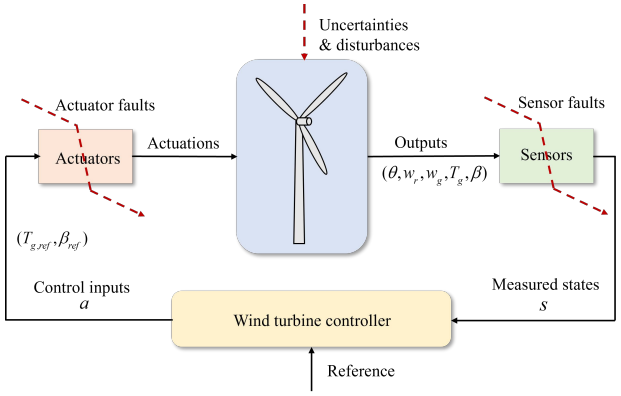

Fig. 1. The wind turbine control system structure with actuator and sensor faults.

A. IHDP Method

The reinforcement learning (RL) method aims to solve optimal control problems through interactions with the environment without using accurate system models. As a class of RL methods in the control field, the adaptive/approximate dynamic programming (ADP) method has been developed by utilizing a function approximator to approximate the value function [22]. It mainly incorporates the critic-actor structure, neural networks, and dynamic programming logic. The incremental model-based heuristic dynamic programming (IHDP) is a recently developed ADP & intelligent control method [23]. It is an effective strategy to solve nonlinear optimal control problems under partial/fully unknown dynamics. IHDP typically employs the critic-actor structure in RL. Specifically, it utilizes a critic network and an actor-network to approximate the state-value function and the optimal control policy, respectively. The specific learning strategy for updating network weights is introduced in the following.

1) Learning Strategy:

The critic network is used to directly approximate the state-value function $\mathcal{Q}(\boldsymbol{s}_k)$, which is a cumulative summation of future rewards from any initial state $\boldsymbol{s}_k$:

$$\mathcal{Q}(\boldsymbol{s}_k) = \sum_{l=k}^{\infty} \gamma^{l-k} r_l \tag{12}$$

where $\gamma \in (0,1)$ is the discount factor, r_l is a one-step reward function, which is designed as a quadratic function of tracking errors between the current state $\boldsymbol{s}_k$ and the desired state $\boldsymbol{s}_k^*$, formulated at the step k as:

$$r_k = (\boldsymbol{s}_k - \boldsymbol{s}_k^*)^T Q_c (\boldsymbol{s}_k - \boldsymbol{s}_k^*) \tag{13}$$

where $Q_c \in \mathcal{R}^{n\times n}$ is a positive-definite matrix. In the IHDP algorithm, an on-policy Temporal Difference (TD) method is applied to iteratively update the critic network, which is defined as:

$$\mathbf{e}_k^c = \widehat{\mathcal{Q}}(\boldsymbol{s}_k) - r_k - \gamma \widehat{\mathcal{Q}}(\boldsymbol{s}_{k+1}) \tag{14}$$

where $\widehat{\mathcal{Q}}(\boldsymbol{s}_k)$ and $\widehat{\mathcal{Q}}(\boldsymbol{s}_{k+1})$ are the estimations of the state-value function. Then, the critic network is updated by minimizing the following error function:

$$\mathbf{E}_k^c = \frac{1}{2} \mathbf{e}_k^{c\,T} \mathbf{e}_k^c \tag{15}$$

Accordingly, the gradient-descent algorithm is employed to update the weights of critic network $\mathbf{w^c}$:

$$\mathbf{w}_{k+1}^c = \mathbf{w}_k^c - \eta_c \cdot \frac{\partial \mathbf{E}_k^c}{\partial \mathbf{w}_k^c} \tag{16}$$

where $\eta_c > 0$ is a user-defined learning rate, and

$$\frac{\partial \mathbf{E}_k^c}{\partial \mathbf{w}_k^c} = \frac{\partial \mathbf{E}_k^c}{\partial \widehat{\mathcal{Q}}(\boldsymbol{s}_k)} \cdot \frac{\partial \widehat{\mathcal{Q}}(\boldsymbol{s}_k)}{\partial \mathbf{w}_k^c} = \mathbf{e}_k^c \cdot \frac{\partial \widehat{\mathcal{Q}}(\boldsymbol{s}_k)}{\partial \mathbf{w}_k^c} \tag{17}$$

We employ the actor-network to find the optimal control policy $\boldsymbol{a}_k^*$ via the estimated state-value function $\widehat{\mathcal{Q}}(\boldsymbol{s}_k)$. Please notice that the optimal state-value function is expected to be zero because the control objective is to minimize the long-term tracking error, i.e. the state-value function. Therefore, the error of the actor network $\mathbf{e}_k^a$ can be formalized by the following equation with $\mathcal{Q}^*(\boldsymbol{s}_{k+1}) = 0$.

$$\mathbf{e}_k^a = \widehat{\mathcal{Q}}(\boldsymbol{s}_{k+1}) - \mathcal{Q}^*(\boldsymbol{s}_{k+1}) = \widehat{\mathcal{Q}}(\boldsymbol{s}_{k+1}) \tag{18}$$

Then, the error function for updating the actor-network is defined as:

$$\mathbf{E}_k^a = \frac{1}{2} (\mathbf{e}_k^a)^T \mathbf{e}_k^a \tag{19}$$

Accordingly, the gradient-descent algorithm is applied to update the weights of actor-network $\mathbf{w^a}$ [24]:

$$\mathbf{w}_{k+1}^a = \mathbf{w}_k^a - \eta_a \cdot \frac{\partial \mathbf{E}_k^a}{\partial \mathbf{w}_k^a} \tag{20}$$

where $\eta_a > 0$ is the learning rate, and

$$\begin{aligned} \frac{\partial \mathbf{E}_k^a}{\partial \mathbf{w}_k^a} &= \frac{\partial \mathbf{E}_k^a}{\partial \widehat{\mathcal{Q}}(\boldsymbol{s}_{k+1})} \cdot \frac{\partial \widehat{\mathcal{Q}}(\boldsymbol{s}_{k+1})}{\partial \boldsymbol{s}_{k+1}} \cdot \frac{\partial \boldsymbol{s}_{k+1}}{\partial \boldsymbol{a}_k} \cdot \frac{\partial \boldsymbol{a}_k}{\partial \mathbf{w}_k^a} \\ &= \mathbf{e}_k^a \cdot \frac{\partial \widehat{\mathcal{Q}}(\boldsymbol{s}_{k+1})}{\partial \boldsymbol{s}_{k+1}} \cdot \frac{\partial \boldsymbol{s}_{k+1}}{\partial \boldsymbol{a}_k} \cdot \frac{\partial \boldsymbol{a}_k}{\partial \mathbf{w}_k^a} \end{aligned} \tag{21}$$

2) Incremental technique:

Conventional HDP methods usually utilize artificial neural networks to approximate the whole dynamics/models of systems [23]. However, evaluating the whole dynamics/models online can be inefficient, especially under faulty conditions requiring quick responses. In contrast, our IHDP method can avoid approximating the whole system model. Specifically, it can approximate nonlinear dynamics using the first-order Taylor series expansion around the latest condition of the system, and only requires the online updates of input matrices instead of whole models.

The dynamics of wind turbines subject to actuator and sensor faults, uncertainties, and disturbances can be rewritten in a continuous-time form as

$$\dot{\boldsymbol{s}}(t) = \bar{f}(\boldsymbol{s}(t), \delta_{\mathcal{F}}(t)) + \bar{G}(\boldsymbol{s}(t), \delta_{\mathcal{F}}(t))\boldsymbol{a}(t) + d(t) \tag{22}$$

where $\boldsymbol{s}(t) \in \mathcal{R}^n$ and $\boldsymbol{a}(t) \in \mathcal{R}^m$ are the system state and control action at the time t, $\bar{f}(\boldsymbol{s}(t), \delta_{\mathcal{F}}(t)) = f(\boldsymbol{s}) + (f_{\mathcal{F}}(\boldsymbol{s}) - f(\boldsymbol{s}))\delta_{\mathcal{F}} + \Delta f(\boldsymbol{s})$ and $\bar{G}(\boldsymbol{s}(t), \delta_{\mathcal{F}}(t)) = G(\boldsymbol{s}) + (G_{\mathcal{F}}(\boldsymbol{s}) - G(\boldsymbol{s}))\delta_{\mathcal{F}} + \Delta G(\boldsymbol{s})$.

By taking the Taylor series expansion at the latest sampling time t_0, the system can be approximated around the operating point $[\boldsymbol{s}(t_0), \boldsymbol{a}(t_0)]$ as follows [25],

$$\begin{aligned}\dot{\boldsymbol{s}}(t) = &\dot{\boldsymbol{s}}\left(t_0\right) + \bar{\mathbf{G}}\left[\boldsymbol{s}\left(t_0\right), \delta_{\mathcal{F}}\left(t_0\right)\right]\Delta\boldsymbol{a}(t) \\ &+ \left.\frac{\partial[\bar{f}(\boldsymbol{s}(t), \delta_{\mathcal{F}}(t)) + \bar{G}(\boldsymbol{s}(t), \delta_{\mathcal{F}}(t))\boldsymbol{a}(t)]}{\partial \boldsymbol{s}(t)}\right|_{\boldsymbol{s}(t_0), \boldsymbol{a}(t_0)} \Delta\boldsymbol{s}(t) \\ &+ \left.\frac{\partial[\bar{f}(\boldsymbol{s}(t), \delta_{\mathcal{F}}(t)) + \bar{G}(\boldsymbol{s}(t), \delta_{\mathcal{F}}(t))\boldsymbol{a}(t)]}{\partial \delta_{\mathcal{F}}(t)}\right|_{\boldsymbol{s}(t_0), \boldsymbol{a}(t_0)} \\ &\cdot \Delta\delta_{\mathcal{F}}(t) + \Delta d(t) + \mathcal{O}\left(\Delta\boldsymbol{s}^2(t)\right)\end{aligned} \tag{23}$$

where $\Delta\boldsymbol{a}(t) = \boldsymbol{a}(t) - \boldsymbol{a}(t_0)$ and $\Delta\boldsymbol{s}(t) = \boldsymbol{s}(t) - \boldsymbol{s}(t_0)$ depict the increments of the control action and system state, respectively. Moreover, $\Delta\delta_{\mathcal{F}}(t) = \delta_{\mathcal{F}}(t) - \delta_{\mathcal{F}}(t_0)$ and $\Delta d(t) = d(t) - d(t_0)$. In addition, $\mathcal{O}\left(\Delta\boldsymbol{s}^2(t)\right)$ represents the higher-order residual terms. By omitting the state variation-related nonlinear terms and higher-order terms, and considering that the fault trigger variation only affects the system around the fault occurrence instants, the incremental model can be further simplified as [26]

$$\Delta\dot{\boldsymbol{s}}(t) \approx \bar{\mathrm{G}}\left[\boldsymbol{s}\left(t_0\right), \delta_{\mathcal{F}}\left(t_0\right)\right]\Delta\boldsymbol{a}(t) + \Delta d(t) \tag{24}$$

After that, the states and control inputs and their relationship can be rewritten in a discrete form. We employ $\boldsymbol{s}_k$ and $\boldsymbol{a}_k$ to denote the discrete form of $\boldsymbol{s}$ and $\boldsymbol{a}$ at the k-th step, respectively. With a proper data sampling frequency, the above-mentioned continuous systems can be transferred back into the discrete form, which is approximated at $\boldsymbol{s}_k$ via the Taylor series expansion as

$$\boldsymbol{s}_{k+1} \approx \boldsymbol{s}_k + \bar{\mathbf{G}}_k \cdot \Delta\boldsymbol{a}_k + \Delta d_k \tag{25}$$

where $\bar{\mathbf{G}}_k = \bar{G}(\boldsymbol{s}_k, \delta_{\mathcal{F}_k})$ is the input matrix at the latest sampling instant under faults and uncertainties, $\Delta\boldsymbol{a}_k = \boldsymbol{a}_k - \boldsymbol{a}_{k-1}$ and $\Delta d_k = d_k - d_{k-1}$ are the increments of control action and disturbance from $k-1$ to k, respectively.

Accordingly, the incremental technique leads to a time-varying linear incremental model for the control algorithms instead of directly utilizing a nonlinear system. The time-varying input matrix $\bar{\mathbf{G}}_k$ can be estimated by an online system evaluation scheme for the implementation of IHDP. The estimated matrix is denoted as $\widehat{\mathbf{G}}_k$, and thus the predicted system state is

$$\hat{\boldsymbol{s}}_{k+1} = \boldsymbol{s}_k + \hat{\mathbf{G}}_k\Delta\boldsymbol{a}_k + \Delta d_k \tag{26}$$

Notably, the designed incremental model shown in Eq. (26) can provide the term $\partial\widehat{\boldsymbol{s}}_{k+1}$ for approximating $\partial\boldsymbol{s}_{k+1}$ in Eq. (21) to update the actor network. The term $\frac{\partial\widehat{\boldsymbol{s}}_{k+1}}{\partial\boldsymbol{a}_k}$ can be expressed as

$$\frac{\partial\widehat{\boldsymbol{s}}_{k+1}}{\partial\boldsymbol{a}_k} \approx \widehat{\mathbf{G}}_k \tag{27}$$

Considering Eq. (27) and Eq. (21) yields

$$\frac{\partial\mathbf{E}_k^a}{\partial\mathbf{w}_k^a} = \mathbf{e}_k^a \cdot \frac{\partial\widehat{\mathcal{Q}}(\widehat{\boldsymbol{s}}_{k+1})}{\partial\widehat{\boldsymbol{s}}_{k+1}} \cdot \widehat{\mathbf{G}}_k \cdot \frac{\partial\boldsymbol{a}_k}{\partial\mathbf{w}_k^a} \tag{28}$$

Note that the estimation of the unknown system dynamics can be approximated by various techniques, such as adaptive neural network [27] and fuzzy logic [28]. In this work, we utilize the recursive least square (RLS) strategy to online estimate the input matrix $\widehat{\mathbf{G}}_k$ of the incremental model due to its simple calculation and good convergence properties, which has the following format

$$\epsilon_k = \Delta\boldsymbol{s}_{k+1}^T - \Delta\widehat{\boldsymbol{s}}_{k+1}^T \tag{29}$$

$$\widehat{\mathbf{G}}_k^T = \widehat{\mathbf{G}}_{k-1}^T + \frac{\mathrm{P}_{k-1}\Delta\boldsymbol{a}_k}{\lambda + \Delta\boldsymbol{a}_k^T\mathrm{P}_{k-1}\Delta\boldsymbol{a}_k}\epsilon_k \tag{30}$$

$$\mathrm{P}_k = \frac{1}{\lambda}\left(\mathrm{P}_{k-1} - \frac{\mathrm{P}_{k-1}\Delta\boldsymbol{a}_k\Delta\boldsymbol{a}_k^T\mathrm{P}_{k-1}}{\lambda + \Delta\boldsymbol{a}_k^T\mathrm{P}_{k-1}\Delta\boldsymbol{a}_k}\right) \tag{31}$$

where $\Delta\widehat{\boldsymbol{s}}_{k+1} = \widehat{\boldsymbol{s}}_{k+1} - \boldsymbol{s}_k$, $\epsilon_k \in \mathcal{R}^n$ is the prediction error, $\Delta\boldsymbol{s}_{k+1} = \boldsymbol{s}_{k+1} - \boldsymbol{s}_k$, $\lambda \in (0, 1]$ is the forgetting factor, $\mathrm{P}_k \in \mathcal{R}^{(m\times m)}$ is the estimation covariance matrix, which stands for the confidence of the estimations.

Remark 1: By incorporating the above-mentioned incremental model and the corresponding RLS updating strategy into the conventional HDP method, the resulting IHDP algorithm can adaptively solve the optimal control problem without prior knowledge of system dynamics or an offline training process. Compared with the traditional FTC methods that are developed based on accurate nominal models, the proposed IHDP method is data-driven and model-free.

Remark 2: The proposed IHDP utilizes an online partial model evaluation technique to update the incremental model (only the input matrix) in real-time according to a series of instantaneous measurements. In this case, the data under the unknown environment that reflects the information on undesirable faults and uncertainties can be collected for the controller, which brings adaptability and fault-tolerant ability.

Remark 3: The proposed IHDP method has the potential to reduce computational complexity compared to the conventional HDP algorithm and the current IHDP methods. This can be manifested in the following aspects: 1) The proposed IHDP method learns the model information by the RLS technique while the conventional HDP algorithm estimates the model information via a neural network - the latter typically requires much data and is more computationally expensive. 2) Unlike the current IHDP methods that estimate the global model information (both the state and input matrices), the proposed HDP method only estimates the partial dynamics information (the input matrix $\widehat{\mathbf{G}}_k$) and thus provides a simple way to lower the computational consumption.

In summary, the proposed IHDP method mainly involves the following three stages: incremental model learning for providing real-time wind turbine dynamics updates, actor-network learning for policy improvement, and critic network learning for policy evaluation. Specifically,

1) The incremental model learning stage approximates the next state $\boldsymbol{s}_{k+1}$ with the estimated input matrix $\widehat{\mathbf{G}}_k$. It directly affects the update of the actor-network. The incremental model learning stage dynamically receives the real-time environmental variations and the information on potential system faults.

2) Based on the incremental model and the currently learned critic, the actor-network learning stage updates and generates actions to be taken. The weights of the actor-network, in

turn, affect the state-value function estimated by the critic, i.e. $\widehat{\mathcal{Q}}(\boldsymbol{s}_{k+1})$, by influencing the action $\boldsymbol{a}_k$ and the next state measurement $\boldsymbol{s}_{k+1}$.

3) The critic network learning stage is related to the designed reward function and the TD technique, which iteratively updates the weights of the critic network based on updated measurements. The effect of the critic network is to approximate the state-value function and therefore guide the actor network in the following learning iteration.

B. RL-Based Wind Turbine FTC Architecture

In this subsection, the RL-based wind turbine FTC method will be developed using the IHDP scheme designed above. Based on the wind turbine dynamics shown in Eq. (1) and the constraints listed in Eqs. (3)-(8), the state space and action space are defined as $\boldsymbol{s} = \{\theta, w_r, w_g, T_g, \beta \mid \theta \geq \theta^{\min}, 0 \leq w_r \leq 1.1 w_r^{\text{rated}}, 0 \leq w_g \leq 1.1 w_g^{\text{rated}}, T_g^{\min} \leq T_g \leq T_g^{\max}, \beta^{\min} \leq \beta \leq \beta^{\max}\}$ and $\boldsymbol{a} = \{T_{g,ref}, \beta_{ref} \mid T_g^{\min} \leq T_{g,ref} \leq T_g^{\max}, \beta^{\min} \leq \beta_{ref} \leq \beta^{\max}, \Delta T_g^{min} \leq \Delta T_{g,ref} \leq \Delta T_g^{\max}, \Delta\beta^{\min} \leq \Delta\beta_{ref} \leq \Delta\beta^{max}\}$, respectively.

1) Target Critic Network Integration:

In our design, a so-called target critic network is integrated into IHDP. Specifically, it is known that the TD error evaluation directly based on the main critic network might diverge, and the learning instability is inevitable with a large and nonlinear state-value function. One solution to alleviate this problem is to employ a target critic network with the evaluation $\widehat{\mathcal{Q}}'(\boldsymbol{s}_k)$ and weights $\mathbf{w^t}$ such that it is updated slower than the main critic network. For this aim, a soft replacement strategy is utilized to update the weights of the target network, defined as

$$\mathbf{w}_{k+1}^t = \tau \mathbf{w}_{k+1}^c + (1-\tau)\mathbf{w}_k^t \tag{32}$$

where $\tau \in (0, 1]$ is a user-defined scalar factor. This strategy means that the weights of the target network will be slowly tracking the main critic network, providing a possible way for improving the training stability and reliability.

Then, the TD error described in Eq. (14) can be rewritten with the target critic network as

$$\mathbf{e}_k^{c'} = \widehat{\mathcal{Q}}(\boldsymbol{s}_k) - r_k - \gamma \widehat{\mathcal{Q}}'(\boldsymbol{s}_{k+1}) \tag{33}$$

2) Design of Neural Networks:

We take the actor-network as an example to explain the general settings for neural networks in our design. The weights of the actor-network from the input layer to the hidden layer and from the hidden layer to the output layer are depicted as $\mathbf{w^{ai}}$ and $\mathbf{w^{ao}}$, respectively. Then, based on Eqs. (20) and (28), the update rules of the actor-network are

$$\mathbf{w}_{k+1}^{ai} = \mathbf{w}_k^{ai} - \eta_{ai} \cdot \frac{\partial \mathbf{E}_k^a}{\partial \mathbf{w}_k^{ai}} \tag{34}$$

$$\mathbf{w}_{k+1}^{ao} = \mathbf{w}_k^{ao} - \eta_{ao} \cdot \frac{\partial \mathbf{E}_k^a}{\partial \mathbf{w}_k^{ao}} \tag{35}$$

where $\eta_{ai} > 0$ and $\eta_{ao} > 0$ are the learning rates, and

$$\frac{\partial \mathbf{E}_k^a}{\partial \mathbf{w}_k^{ai}} = \mathbf{e}_k^a \cdot \frac{\partial \widehat{\mathcal{Q}}(\widehat{\boldsymbol{s}}_{k+1})}{\partial \widehat{\boldsymbol{s}}_{k+1}} \cdot \widehat{\mathbf{G}}_k \cdot \frac{\partial \boldsymbol{a}_k}{\partial \mathbf{w}_k^{ai}} \tag{36}$$

$$\frac{\partial \mathbf{E}_k^a}{\partial \mathbf{w}_k^{ao}} = \mathbf{e}_k^a \cdot \frac{\partial \widehat{\mathcal{Q}}(\widehat{\boldsymbol{s}}_{k+1})}{\partial \widehat{\boldsymbol{s}}_{k+1}} \cdot \widehat{\mathbf{G}}_k \cdot \frac{\partial \boldsymbol{a}_k}{\partial \mathbf{z}_k^a} \cdot \frac{\partial \mathbf{z}_k^a}{\partial \mathbf{w}_k^{ao}} \tag{37}$$

where $\mathbf{z}_k^a$ is the output value of the hidden layer.

3) FTC Architecture Elaboration:

For better understanding, the procedure and the block diagram of our IHDP algorithm for wind turbine FTC are given in Algorithm 1 and Fig. 2, respectively. It can be seen that the proposed control scheme has an incremental model that requires only the input matrix to be estimated online and serves to update the actor-network. Such a special design is of significance to complex wind turbine systems since it is difficult to acquire accurate wind turbine models, especially under faults and uncertainties. Another benefit is that less model knowledge is updated by the RLS technique, which leads to reduced computational burden compared with the current methods that dig out whole model information under uncertain and faulty conditions. In addition, our IHDP utilizes a target critic network to improve the training stability, which provides a more reliable learning process for the wind turbine FTC in a wide range of time-varying unknown environments.

Remark 4: To achieve fault-tolerance and optimal control performance simultaneously, this work integrates the RL technique into the FTC framework for wind turbines such that the optimal action will be generated in each step under a given reward optimization task. To further improve the learning efficiency, this work embeds the incremental wind turbine model into the well-known RL structure such that only partial system changes are required to be identified/updated in real-time. In other words, it only dynamically evaluates and updates the input matrix of the incremental wind turbine model, which can not only enhance the online model evaluation efficiency but also ensure optimal control performance with guaranteed fault tolerance and robustness.

Remark 5: The discount factor in the reward function ranges from 0 to 1 (inclusive), which usually aims to reduce the weight of future rewards and balance the short-term and long-term goals. The lower values of the discount factor mean more emphasis is placed on the immediate rewards [29]. The forgetting factor ($0 < \lambda \leq 1$) in the RLS strategy means giving greater weight to recent data and less weight to older data. It is regarded as a weight that decreases as remote data increases [30]. The scalar factor for soft replacement is usually chosen following the rule $\tau \ll 1$. In this way, the target critic network can slowly track the critic network for learning stability improvement.

In summary, the proposed RL-based FTC scheme for wind turbines has the following advantages: 1) The dependency on accurate nominal wind turbine models is released. 2) The target critic network is employed to improve the learning stability for the wind turbine FTC task under time-varying environmental conditions. 3) The optimization ability, robustness, and safety are achieved by the RL-based FTC scheme with ensured fault-tolerance capability.

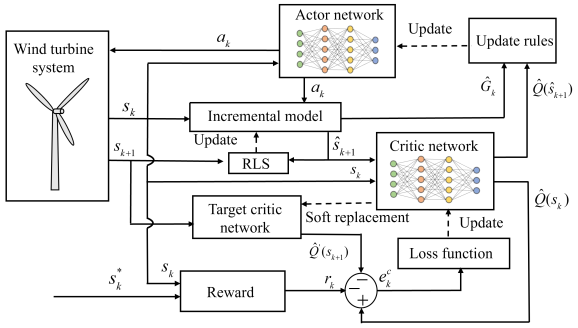

Fig. 2. The block diagram of the proposed IHDP algorithm for wind turbines.

ALGORITHM 1 IHDP Algorithm for Wind Turbine Fault-Tolerant Control

1 : Provide the initial state $\mathbf{s}_0$ and action $\mathbf{a}_0$
2 : Initialize the weights of critic, actor, and target critic networks $\mathbf{w}_k^c$, $\mathbf{w}_k^a$ and $\mathbf{w}_k^T$
3 : **for** $k = 0$ **to** k_{max} **do**
4 : Collect the measured state at next step $\mathbf{s}_{k+1}$ and the output of the critic network $\hat{\mathbf{s}}_{k+1}$
5 : Compute the estimated input matrix $\hat{\mathbf{G}}_k$ according to Eqs. (29)-(31)
6 : Calculate the reward shown in Eq. (13) and the TD error for the loss function in Eq. (33)
7 : Update the critic network according to the Eq. (17)
8 : Update the actor network according to the Eqs. (28)
9 : Update the target critic network according to the Eq. (32)
10 : $k = k + 1$
11 : **end for**

IV. Numerical Simulations

In this section, the performance and effectiveness of the proposed method are validated for wind turbine torque and pitch control tasks considering actuator and sensor faults. The FAST (Fatigue, Aerodynamics, Structures, and Turbulence) simulator developed by NREL (National Renewable Energy Laboratory) is adopted [19]. The forgetting factor λ, scalar factor τ, and discount factor γ are set as 1, 0.01, and 0.9 respectively. The desired control references are given by the lookup table scheme [31]. Two different wind profiles with the mean speeds of 9 m/s and 18 m/s, generated by the NREL software TurbSim [32], are employed for verification. The simulation time in each test is set as $200s$. The rated rotor speed w_r^{rated} and rated generator speed w_g^{rated} are set as 1.2671 rad/s and 122.9096 rad/s, respectively. The values of T_g^{max}, T_g^{min}, β^{max}, β^{min}, ΔT_g^{max}, ΔT_g^{min}, $\Delta\beta^{max}$, and $\Delta\beta^{min}$ are set as 47,403 $N \cdot m$, 200 $N \cdot m$, 90°, 0°, 1500 $N \cdot m$, 1500 $N \cdot m$, 8°, and $-8°$, respectively.

To validate the effectiveness and performance of the proposed method, the existing HDP strategy [24], the model predictive (MPC) method [33], and the baseline controller [34] are carried out in simulations for comparison. The scalar factor and discount factor in the HDP method are respectively set as 0.01 and 0.9. In the MPC strategy, the state and control input weight matrices in the objective function are selected as $[1, 10^4, 10, 10^3, 10^3]^T$ and $[1, 10^4]^T$, respectively. In baseline control, the blade-pitch controller proportional, integral, and derivative gains are set as 0.025, 0.015, and 0.18, respectively. The proportional gain in the torque control is set as 2.33.

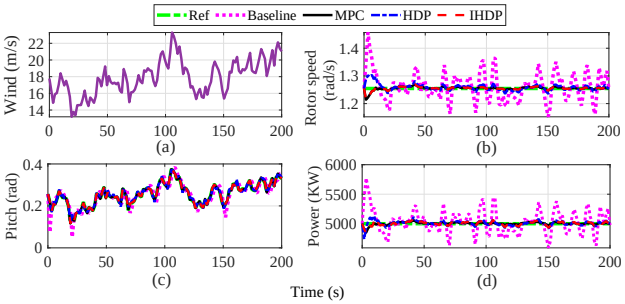

Fig. 3. Simulation results under the nominal condition by the IHDP method, HDP algorithm, MPC strategy, and baseline controller with the mean wind speed of 18 m/s. (a) Wind speed profiles with the mean of 18 m/s. (b) Rotor speeds. (c) Pitch angles. (d) Power outputs.

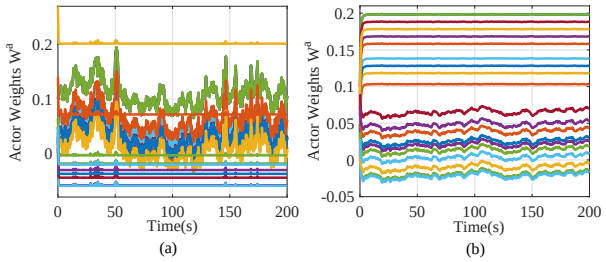

Fig. 4. Simulation results of actor weights. (a) Under the mean wind speed of 9 m/s. (b) Under the mean wind speed of 18 m/s.

A. Simulation Results Under Nominal Conditions

The first set of simulations is conducted under nominal conditions (no faults). Fig. 3 presents the simulation results of the different controllers under a wind profile with the mean speed of 18 m/s, as shown in Fig. 3(a). Therein, from Fig. 3(b) and (c), all the controllers can track the reference rotor speeds and pitch commands, while the baseline controller exhibits considerable fluctuations compared with the IHDP, HDP, and MPC methods. Fig. 3(d) displays the power production results. Compared with the baseline controller, the power fluctuation amplitudes under the IHDP, HDP, and MPC are significantly decreased and thus the power quality is improved. This is because fluctuating transient power trajectories can damage the wind turbine converter. It is worth noting that the control performance of the proposed method is similar to that of MPC under the nominal case, illustrating the optimization ability of our RL-based method.

The updating process of $\mathbf{w}^{\mathbf{a}}$ under the proposed method is presented in Fig. 4. Furthermore, Fig. 5 displays the online estimated results of several entries in the input matrix $\hat{\mathbf{G}}_k$, i.e., $\frac{\partial \hat{\mathbf{s}}_{k+1}}{\partial \boldsymbol{a}_k}$. For the wind turbine, we have

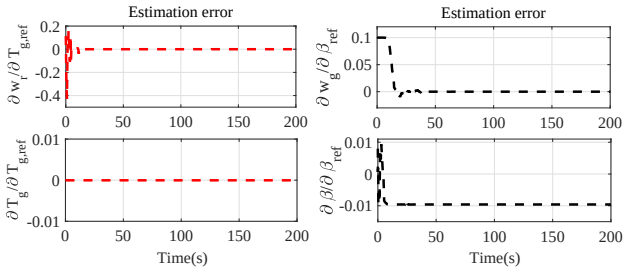

Fig. 5. Estimated terms in the matrix $\widehat{\mathbf{G}}_k$. (a) Under the mean wind speed of 9 m/s. (b) Under the mean wind speed of 18 m/s.

$$\frac{\partial \widehat{\boldsymbol{s}}_{k+1}}{\partial \boldsymbol{a}_k} = \begin{bmatrix} \frac{\partial \theta}{\partial T_{g,ref}} & \frac{\partial w_r}{\partial T_{g,ref}} & \frac{\partial w_g}{\partial T_{g,ref}} & \frac{\partial T_g}{\partial T_{g,ref}} & \frac{\partial \beta}{\partial T_{g,ref}} \\ \frac{\partial \theta}{\partial \beta} & \frac{\partial w_r}{\partial \beta} & \frac{\partial w_g}{\partial \beta} & \frac{\partial T_g}{\partial \beta_{ref}} & \frac{\partial \beta}{\partial \beta_{ref}} \end{bmatrix}^T \tag{38}$$

It can be observed from Fig. 5 that the estimation errors quickly converge to zero, which demonstrates the effectiveness of the proposed incremental model-based RLS estimator in providing accurate system information to the proposed wind turbine controller.

B. Simulation Results Under Torque Actuator Faults

To evaluate the proposed method's fault-tolerance ability against actuator failures, a fault with a $+5000\ Nm$ torque offset from 60 s to 120 s is tested. In addition, parameter uncertainties and measuring noises are also considered. Specifically, the parameters for stiffness coefficient and torsion damping coefficient in the turbine are set to be 30% of their true values. Gaussian noises with zero means and standard deviations of 0.01 rad/s and 0.1 rad/s are considered in the rotor and generator speeds, respectively. Fig. 6 depicts the generator torques and tracking errors under the four controllers considering actuator offset faults. For better presentation, some local magnifications are given. It can be observed that the torque values under the baseline controller leave away from the reference with the largest errors compared with the other three methods. The proposed method achieves the best performance with the smallest errors, followed by the traditional HDP method and then the MPC Method.

The power responses of the four controllers under the faulty conditions are shown in Fig. 7. It can be seen that our IHDP method performs best in power generation as it approaches near the optimal value. The conventional HDP method has larger errors than the proposed IHDP method, but it performs better than the MPC strategy and the baseline controller. In contrast, the response of the baseline controller can not accurately follow the optimal trajectories. As for the MPC method, there exists remarkable power tracking accuracy deterioration during the initial faulty period. In contrast, the power output under the proposed method is quickly driven back to the optimal power trajectory. Therefore, the proposed method has strong fault tolerance under faulty conditions. Notably, after the faulty period, our IHDP controller still performs expected behaviors while the others fail to achieve the desired control goals.

The torque changes under additional offset faults situations ($+3000\ Nm$ and $+7000\ Nm$) are compared in Fig. 8. It can be observed that the IHDP method and the conventional HDP can maintain better generator torque performance than the MPC and baseline controllers. Moreover, the generator torque under the IHDP method is closer to the reference values than in the traditional HDP method. As for MPC and baseline controllers, the larger the fault offset value is, the greater torque deviation from the desired value is created.

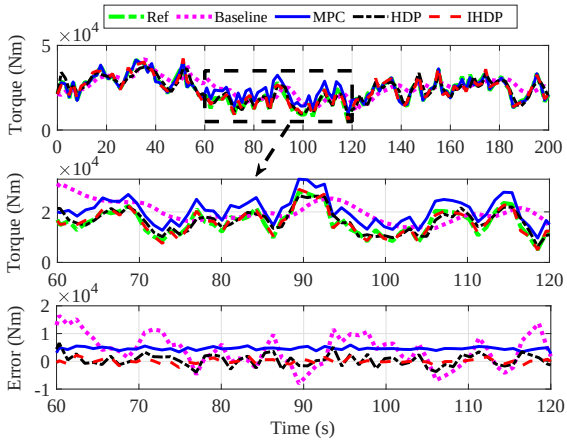

Fig. 6. Generator torques under different controllers considering the torque actuator offset fault ($+5000\ Nm$).

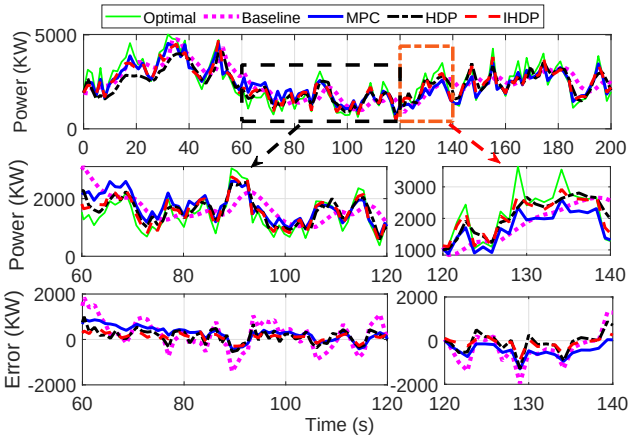

Fig. 7. Power outputs under different controllers considering the torque actuator offset fault ($+5000\ Nm$).

Moreover, the mean square error (MSE) [30] is employed for the characterization of fault-tolerant control accuracy, which is defined as the average of the squares of errors:

$$MSE = \frac{1}{M}\sum_{i=1}^{M}(y_i - y_i^*)^2 \tag{39}$$

where y_i denotes the measured data, y_i^* is the reference data, and M is the total number of data.

The quantitative comparison results of MSE values of the generator torque and power under different controllers considering torque actuator offset faults are listed in Table II.

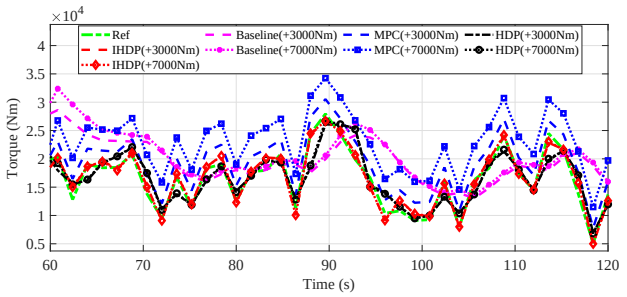

Fig. 8. Torque variations under different controllers under offset faults ($+3000\ Nm$ and $+7000\ Nm$).

It can be seen that the MSE values under the proposed IHDP method are the smallest among all controllers.

TABLE II
MSE VALUES OF THE GENERATOR TORQUE AND POWER UNDER DIFFERENT CONTROLLERS CONSIDERING TORQUE ACTUATOR OFFSET FAULTS.

Cases	Methods	Baseline	MPC	HDP	IHDP
+3000Nm	**Torque**($*10^7$)	3.79	1.26	0.28	0.26
	Power($*10^5$)	6.42	3.69	1.27	1.16
+5000Nm	**Torque**($*10^7$)	3.96	1.27	0.68	0.27
	Power($*10^5$)	6.50	3.69	1.53	1.26
+7000Nm	**Torque**($*10^7$)	4.15	1.30	1.28	0.26
	Power($*10^5$)	6.61	3.68	3.69	1.22

C. Simulation Results Under Sensor Faults

In this sub-section, two types of sensor faults are considered, as listed in Table III. Additionally, the parameter uncertainties and measuring noises mentioned in the previous case study are also considered.

TABLE III
TWO TYPES OF POSSIBLE SENSOR FAULTS

Sensor	Type	Value	Faulty period
Generator speed sensor	Partial failure	-50%	60s-110s
Generator speed sensor	Stuck fault	100 rad/s	100s-150s

Figs. 9 presents the generator speed, pitch angle, and power production under the partial failure of the generator speed sensor and uncertainties. As expected, the generator speed and the pitch angle can precisely follow the desired trajectory under the proposed method. The conventional HDP method exhibits larger tracking errors compared with the proposed method. These two methods have better control performance than the MPC and baseline controller, which suffer from large deviations from the reference trajectories. In addition, the optimal power expectation is closely maintained by our IHDP method, which performs better than the conventional HDP algorithm. In contrast, the power production trajectories under the MPC and baseline controllers are seriously affected

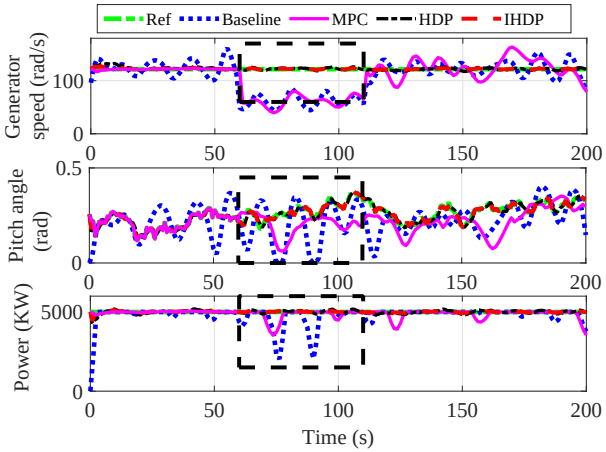

Fig. 9. Simulation results under different controllers considering the partial failure of sensors and uncertainties.

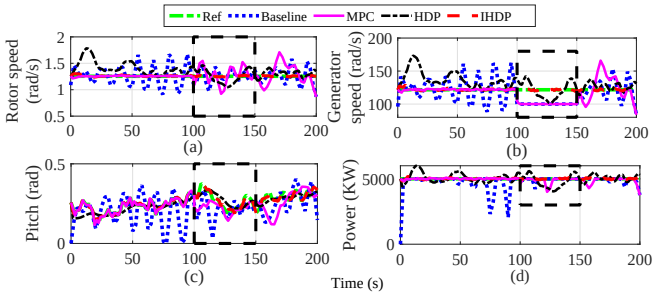

Fig. 10. Simulation results under different controllers considering the stuck faults of sensors and uncertainties.

by the sensor's partial failure even after the faulty conditions are recovered.

The simulation results under the sensor stuck fault and uncertainties are presented in terms of rotor & generator speeds, pitch angles, and power outputs in Fig. 10, where the local magnification displays the results under the MPC and IHDP methods during the faulty period. It can be observed that the proposed method can restore the rotor & generator speeds, pitch angles, and power outputs to their references, which illustrates the effective fault compensation and strong robustness of the proposed method. Although the conventional HDP algorithm also shows a certain level of robustness to sensor faults, it creates larger errors than the proposed method. In addition, the results under the MPC scheme can eventually converge around the references, but large errors are created after the fault occurs. As for the baseline controller, it can be seen that the wind turbine control performance and power generation severely deteriorate. It can be concluded that the proposed IHDP method achieves acceptable performance under generator speed sensor faults, followed by the conventional HDP controller, while the MPC and the baseline algorithms fail to recover the control performance under the selected faulty conditions.

Table IV provides MSE values of the generator speed and power generation under different controllers. It can be observed that our method's MSE values are the smallest in all

cases. The results listed in both Tables II and IV indicate that our method achieves higher control accuracy, especially under faulty conditions.

TABLE IV
MSE VALUES OF THE GENERATOR SPEED AND POWER UNDER DIFFERENT CONTROLLERS CONSIDERING SENSOR FAULTS.

Cases \ Methods		Baseline	MPC	HDP	IHDP
Sensor partial failure	**Generator speed** ($*10^3$)	6.93	1.16	0.26	0.00016
	Pitch ($*10^{-3}$)	121.2	4.30	1.30	0.083
	Power ($*10^6$)	17.1	1.13	0.50	0.35
Sensor stuck fault	**Generator speed** ($*10^3$)	30.30	0.26	0.18	0.0028
	Pitch ($*10^{-3}$)	293.5	1.65	1.42	0.094
	Power ($*10^6$)	17.4	0.91	0.67	0.28

D. Simulation Results Under Strong Uncertainties

This subsection aims to test the performance of the proposed method under strong uncertainties. In addition to uncertain stiffness & torsion damping coefficients, extra uncertainties are considered. Specifically, the rotor inertia and generator inertia uncertainties in the wind turbine models are set to be 30% of their true values. Moreover, the pitch angle is polluted by normally distributed random noises with a standard deviation of 0.01 deg. Considering these uncertainties, simulation results under different controllers are shown in Fig. 11. We can see the generator speeds and powers under the IHDP, HDP, and MPC methods stay much closer to the reference values than the baseline controller. Moreover, IHDP and MPC achieve comparable performance, and both of them perform better than the HDP method. As for power generation, our IHDP demonstrates the best power behavior among the four controllers. The baseline controller fails to guarantee a desirable tracking performance and leads to significant fluctuations.

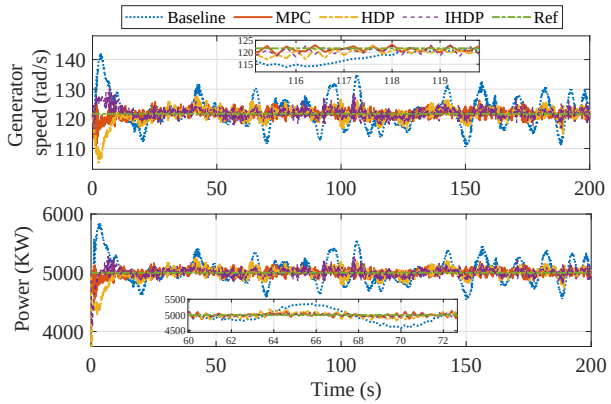

Fig. 11. Simulation results under different controllers considering strong uncertainties.

E. Simulation Results Under Intermittent Faults

This subsection tests the performance of the proposed method under intermittent faults that occur irregularly and unpredictably. Specifically, in Case 1, the generator torque is set to have intermittent offset faults with +5000 Nm in 50 s-90 s and -5000 Nm in 130 s-170 s. In Case 2, the generator speed is set to have stuck faults at 100 rad/s in 50 s-90 s and partial faults with -20 % in 130 s-170 s. Simulation results under different controllers are shown in Fig. 12 and Fig. 13. From Fig. 12, it can be seen that the torque and power under the IHDP and HDP methods stay closer to the reference values than the MPC and baseline controllers, indicating the strong fault-tolerance of the proposed method. As for the results under the sensor's intermittent faults, it can be seen from Fig. 13 that the IHDP and HDP have comparable performance in generator speed and power generation, and both of them perform better than the MPC and baseline methods.

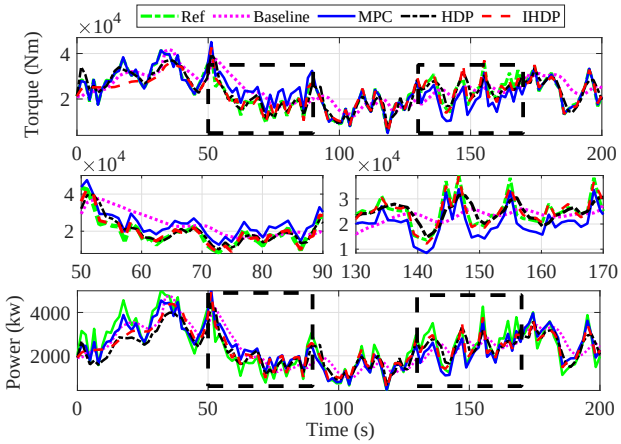

Fig. 12. Simulation results under different controllers considering intermittent faults in torque control.

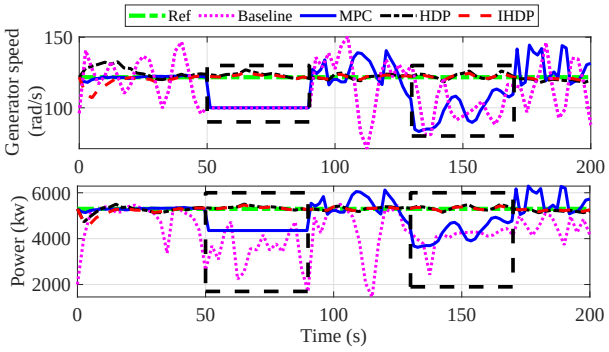

Fig. 13. Simulation results under different controllers considering intermittent faults in pitch control.

F. Comparison With the Model-Free Adaptive Control Method

In addition to the previous comparisons with the well-known MPC, HDP, and baseline controllers, this subsection further evaluates the performance of the proposed IHDP method compared to an advanced model-free turbine control method, called the model-free adaptive control (MFAC) (Ref. [35]). Simulation results for both methods in a faulty-free (nominal) case are presented in Fig. 14. It can be observed

that both controllers perform comparable control behaviors in the nominal case. Considering the torque actuator offset faults with +5000 Nm in 60 s - 100 s, the wind turbine torque control performance under these two controllers is shown in Fig. 15. It can be seen that the generator torque under the proposed IHDP method stays closer to its desired values under the chosen faults than the MFAC algorithm.

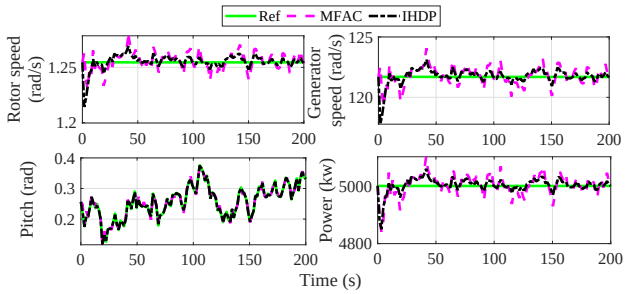

Fig. 14. Simulation results under the proposed IHDP and the MFAC algorithm in faulty-free condition.

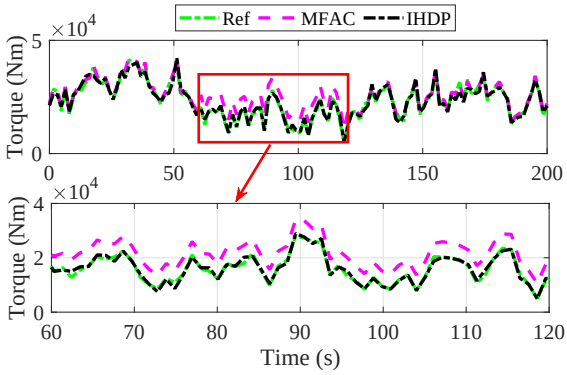

Fig. 15. Simulation result of the generator torque under the proposed IHDP and the MFAC algorithm in a faulty condition.

In summary, extensive simulation results indicate the effectiveness and strong fault-tolerance ability of our IHDP method. In nominal cases, the results of IHDP are highly close to the optimal solutions and similar to the conventional HDP algorithm and MPC strategy. In faulty conditions, the proposed IHDP method has better performance than the conventional HDP algorithm, MPC strategy, baseline controller, and the MFAC method. The key features of the proposed algorithm compared with other methods are summarized in Table V.

V. Conclusion

A reinforcement learning (RL) based passive fault-tolerant controller was proposed in this work for wind turbine torque & pitch control tasks under actuator & sensor faults and uncertainties. To release the dependency on the accurate and complex turbine model knowledge, an incremental model-based heuristic dynamic programming (IHDP) algorithm was developed. It was built upon a main critic-actor structure with an incremental model that only estimates the input matrix in real-time. This strategy enhanced the efficiency of the system approximation and simplified the implementation process with reduced computational complexity. Simulation results with the FAST simulator verified the effectiveness and adaptability of the proposed algorithm. Comparison studies showed that our IHDP-based wind turbine control method has strong fault-tolerance ability and robustness in comparison with several existing methods.

It is worth mentioning that the proposed technique would be negatively affected by unexpected large noises and system delays because of the incremental model derivation process. Therefore, our future research will focus on eliminating the influences of noises and delays on incremental models. Another key attention would also be paid to bringing out novel physics-informed neural network techniques for wind turbine fault-tolerant control designs such that the training efficiency and overall control performance can be further enhanced. Notably, the RL-based FTC methods can be applied not only to the wind turbine and farm control task but also to a wide range of autonomous systems, such as unmanned aerial vehicles, spacecraft, autonomous vehicles, and robots in various control missions.

References

[1] F. Farivar and M. N. Ahmadabadi, "Continuous reinforcement learning to robust fault tolerant control for a class of unknown nonlinear systems," *Applied Soft Computing*, vol. 37, pp. 702–714, 2015.

[2] L. Liu, Y.-J. Liu, and S. Tong, "Neural networks-based adaptive finite-time fault-tolerant control for a class of strict-feedback switched nonlinear systems," *IEEE transactions on cybernetics*, vol. 49, no. 7, pp. 2536–2545, 2018.

[3] X. Shao, Q. Hu, Y. Shi, and B. Jiang, "Fault-tolerant prescribed performance attitude tracking control for spacecraft under input saturation," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 2, pp. 574–582, 2018.

[4] B. Xiao and S. Yin, "A deep learning based data-driven thruster fault diagnosis approach for satellite attitude control system," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 10, pp. 10 162–10 170, 2020.

[5] Q. Zhang, X. Zhang, B. Zhu, and V. Reppa, "Fault tolerant control for autonomous surface vehicles via model reference reinforcement learning," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 1536–1541.

[6] H. Badihi, Y. Zhang, and H. Hong, "Wind turbine fault diagnosis and fault-tolerant torque load control against actuator faults," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1351–1372, 2014.

[7] J. Lan, R. J. Patton, and X. Zhu, "Fault-tolerant wind turbine pitch control using adaptive sliding mode estimation," *Renewable Energy*, vol. 116, pp. 219–231, 2018.

[8] A. Azizi, H. Nourisola, and S. Shoja-Majidabad, "Fault tolerant control of wind turbines with an adaptive output feedback sliding mode controller," *Renewable energy*, vol. 135, pp. 55–65, 2019.

[9] Y. Liu, R. J. Patton, and S. Shi, "Wind turbine asymmetrical load reduction with pitch sensor fault compensation," *Wind Energy*, vol. 23, no. 7, pp. 1523–1541, 2020.

[10] M. Mazare, M. Taghizadeh, and P. Ghaf-Ghanbari, "Fault tolerant control of wind turbines with simultaneous actuator and sensor faults using adaptive time delay control," *Renewable Energy*, vol. 174, pp. 86–101, 2021.

[11] F. Shi and R. Patton, "An active fault tolerant control approach to an offshore wind turbine model," *Renewable Energy*, vol. 75, pp. 788–798, 2015.

[12] T. Sadamoto and A. Chakrabortty, "Fast real-time reinforcement learning for partially-observable large-scale systems," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 3, pp. 206–218, 2020.

[13] A. Tittaferrante and A. Yassine, "Multi-advisor reinforcement learning for multi-agent multi-objective smart home energy control," *IEEE Transactions on Artificial Intelligence*, 2021.

[14] L. Qian, X. Zhao, P. Liu, Z. Zhang, and Y. Lv, "Design of observer-based control with residual generator using actor-critic reinforcement learning," *IEEE Transactions on Artificial Intelligence*, 2022.

TABLE V
QUALITATIVE COMPARISON AMONG DIFFERENT CONTROLLERS.

Items \ Methods	Baseline	MPC	MFAC	HDP	IHDP-G	IHDP (this work)
Model-free	Yes	None	Yes	Yes	Yes	Yes
Learning ability	None	None	None	Yes	Yes	Yes
Real-time dynamics identification	None	None	None	Global	Global	Partial
Reduced perturbation layer analysis	None	None	None	None	None	Yes
Control optimality	None	Yes	Yes	Yes	Yes	Yes
Control design concept	PI	Prediction & optimization	Estimation & optimization	RL, NN	RL, NN, RLS	RL, NN, RLS

[15] L. Liu, Z. Wang, and H. Zhang, "Adaptive fault-tolerant tracking control for mimo discrete-time systems via reinforcement learning algorithm with less learning parameters," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 299–313, 2016.

[16] Y. Sohège, M. Quiñones-Grueiro, and G. Provan, "A novel hybrid approach for fault-tolerant control of uavs based on robust reinforcement learning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 10 719–10 725.

[17] J. Ma and C. Peng, "Adaptive model-free fault-tolerant control based on integral reinforcement learning for a highly flexible aircraft with actuator faults," *Aerospace Science and Technology*, vol. 119, p. 107204, 2021.

[18] C. Zhang, X. Xu, and X. Zhang, "Dual heuristic programming with just-in-time modeling for self-learning fault-tolerant control of mobile robots," *Optimal Control Applications and Methods*, 2021.

[19] J. M. Jonkman and M. L. Buhl Jr, "Fast user's guide-updated august 2005," National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2005.

[20] H. Jafarnejadsani, J. Pieper, and J. Ehlers, "Adaptive control of a variable-speed variable-pitch wind turbine using radial-basis function neural network," *IEEE transactions on control systems technology*, vol. 21, no. 6, pp. 2264–2272, 2013.

[21] P. F. Odgaard, J. Stoustrup, and M. Kinnaert, "Fault-tolerant control of wind turbines: A benchmark model," *IEEE Transactions on control systems Technology*, vol. 21, no. 4, pp. 1168–1182, 2013.

[22] Y. Zhou, E. van Kampen, and Q. Chu, "Incremental model based heuristic dynamic programming for nonlinear adaptive flight control," in *Proceedings of the International Micro Air Vehicles Conference and Competition*, 2016.

[23] B. Sun and E.-J. van Kampen, "Incremental model-based heuristic dynamic programming with output feedback applied to aerospace system identification and control," in *2020 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2020, pp. 366–371.

[24] Y. Zhou, E. van Kampen, and Q. P. Chu, "Launch vehicle adaptive flight control with incremental model based heuristic dynamic programming," in *68th International Astronautical Congress (IAC)*, 2017.

[25] X. Wang, E.-J. van Kampen, and Q. Chu, "Quadrotor fault-tolerant incremental nonsingular terminal sliding mode control," *Aerospace Science and Technology*, vol. 95, p. 105514, 2019.

[26] X. Wang and S. Sun, "Incremental fault-tolerant control for a hybrid quad-plane uav subjected to a complete rotor loss," *Aerospace Science and Technology*, vol. 125, p. 107105, 2022.

[27] Y. Li, Y. Liu, and S. Tong, "Observer-based neuro-adaptive optimized control of strict-feedback nonlinear systems with state constraints," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 7, pp. 3131–3145, 2021.

[28] Y.-m. Li, X. Min, and S. Tong, "Adaptive fuzzy inverse optimal control for uncertain strict-feedback nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 10, pp. 2363–2374, 2019.

[29] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.

[30] A. Vahidi, A. Stefanopoulou, and H. Peng, "Recursive least squares with forgetting for online estimation of vehicle mass and road grade: theory and experiments," *Vehicle System Dynamics*, vol. 43, no. 1, pp. 31–55, 2005.

[31] Y. Xia, K. H. Ahmed, and B. W. Williams, "Wind turbine power coefficient analysis of a new maximum power point tracking technique," *IEEE transactions on industrial electronics*, vol. 60, no. 3, pp. 1122–1132, 2012.

[32] B. J. Jonkman and M. L. Buhl Jr, "Turbsim user's guide," National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2006.

[33] X. Feng and R. Patton, "A model-based predictive control for ftc for wind turbine wind speed sensor fault," in *2013 Conference on Control and Fault-Tolerant Systems (SysTol)*. IEEE, 2013, pp. 504–509.

[34] J. Jonkman, S. Butterfield, W. Musial, and G. Scott, "Definition of a 5-mw reference wind turbine for offshore system development," National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2009.

[35] S. Lin, P. Qi, and X. Zhao, "Power generation control of a hydrostatic wind turbine implemented by model-free adaptive control scheme," *Wind energy*, vol. 23, no. 4, pp. 849–863, 2020.

Jingjie Xie is currently a Research Assistant Professor with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong, China. She received the Ph.D. degree in Engineering at the School of Engineering, University of Warwick, Coventry, UK, in 2023. She received the B.S. degree in information engineering from Northwestern Polytechnical University, Xian, China, in 2016, and the M.S. degree in control science and engineering from Beijing University of Aeronautics and Astronautics, Beijing, China, in 2019. Her current research interests include reinforcement learning, deep learning, and intelligent control.

Hongyang Dong is an Assistant Professor with the School of Engineering, University of Warwick, Coventry, U.K.. He received the Ph.D. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2018. He was a Research Fellow in Machine Learning and Intelligent Control with the University of Warwick from 2019 to 2022, before he became an Assistant Professor in November 2022. His current research interest is control theories and machine learning methods with their applications in complex systems, including offshore renewable energy systems and autonomous systems.

Xiaowei Zhao is Professor of Control Engineering with the School of Engineering, University of Warwick, Coventry, U.K.. He received the Ph.D. degree in control theory from Imperial College London, London, U.K., in 2010. He was a Postdoctoral Researcher with the University of Oxford, Oxford, U.K., for three years before joining the University of Warwick, in 2013. His main research interests include control theory and machine learning with applications in offshore renewable energy systems, smart grids, and autonomous systems.

Shuyue Lin is a lecturer in electrical and electronic engineering at the University of Hull, UK. Before that, she was an assistant professor at Fuzhou University, China. She obtained her PhD in engineering (University of Warwick), MSc in control systems (Imperial College London), and BEng in electrical power engineering (the University of Bath and North China Electric Power University) in 2019, 2013, and 2012, respectively. Her current research interests include power generation and grid integration of renewable energies, fault detection and diagnosis in power systems.