Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Black-box adversarial attack on deep reinforcement learning-based PID controller for load frequency control*

Wei WANG[1], Zhenyong ZHANG[†‡1,2], Xin WANG[2], Xuguo JIAO[3,4]

[1]*State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China*

[2]*Key Laboratory of Computing Power Network and Information Security, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China*

[3]*School of Information and Control Engineering, Qingdao University of Technology, Qingdao 266061, China*

[4]*State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310058, China*

[†]E-mail: zhangzy@gzu.edu.cn

Received Nov. 23, 2024; Revision accepted July 25, 2025; Crosschecked

Unedited

Abstract: Load frequency control (LFC) is usually managed by traditional proportional integral derivative (PID) controllers. Recently, deep reinforcement learning (DRL)-based adaptive controllers have been widely studied for their superior performance. However, the DRL-based adaptive controller exhibits inherent vulnerability due to adversarial attacks. To develop more robust control systems, this study conducts a deep analysis of DRL-based adaptive controller vulnerability under adversarial attacks. First, an adaptive controller is developed based on the DRL algorithm. Subsequently, considering the limited capability of attackers, the DRL-based LFC is evaluated under adversarial attacks using the zeroth-order optimization (ZOO) method. Finally, we use adversarial training to enhance the robustness of DRL-based adaptive controllers. Extensive simulations are conducted to evaluate the performance of the DRL-based PID controller with and without adversarial attacks.

Key words: Adaptive controller; Deep reinforcement learning; Load frequency control; Adversarial attacks
https://doi.org/10.1631/FITEE.2401021 **CLC number:**

1 Introduction

The power system is a critical infrastructure in modern society. A stable supply of electricity is crucial for maintaining world development. Load frequency control (LFC) is responsible for power system stability by maintaining the grid frequency according to the reference value (Shangguan et al., 2021). Typically, LFC relies on automatic generation control to stabilize the frequency of the entire network. The control approaches include the proportional integral derivative (PID) controller (Shabani et al., 2013), heuristic intelligent algorithm (Raju and Srikanth, 2024), and fuzzy logic (Doan et al., 2022). The PID controller is widely used because of its simple structure, effectiveness, and practicality. However, implementing a high-performance traditional PID controller often requires complex system model analysis. This process is both time-consuming and technically demanding(Muduli et al., 2025). In addition, the increasing penetration of renewable energy has increased the uncertainty in power system modeling, and traditional PID methods, which rely on precise system models, struggle to cope with load fluctuations and generation uncertainties (Chen et al.,

‡ Corresponding author

* Project supported by the Natural Science Foundation of China (Nos. 62303126, 62362008, and 62203249), the Major Scientific and Technological Special Project of Guizhou Province (No. [2024]014), the Key Laboratory of Computing Power Network and Information (No. 2023ZD037), the Taishan Scholars Program (No. tsqn202408239), and the Open Research Project of the State Key Laboratory of Industrial Control Technology of Zhejiang University (No. ICT2025B39)

ORCID: Zhenyong ZHANG, https://orcid.org/0000-0003-0950-1525

© Zhejiang University Press 2025

2024).

Recently, deep reinforcement learning (DRL) has been regarded as a promising solution because of its powerful exploration capabilities, adaptability, and the fact that it does not require physical modeling (Yan and Xu, 2019). Xue et al. (2024) studied the wind disturbance resistance strategy for quadrotor UAVs using DRL technology. Shi et al. (2023) explore how DRL can achieve sim-to-real transfer, enabling strategies learned in simulation environments to be transferred to the real world, and many researchers have combined DRL with the PID controller to develop intelligent adaptive controllers.Dogru et al. (2022) addressed the real-time control issue by applying DRL to tune the PID controller autonomously. Muduli et al. (2025) presented an actor-critic DRL-based adaptive controller to handle nonlinearity, uncertainties, and parameter variations without requiring pre-learning or prior knowledge. Shuprajhaa et al. (2022) proposed a modified proximal policy optimization DRL algorithm to design an adaptive controller and achieved excellent performance in managing unstable complex systems.

However, because the DRL-based adaptive controllers are essentially deep neural networks (DNNs), they are inherently vulnerable (Behzadan and Munir, 2017; Michel et al., 2022; Zhang et al., 024a, 2024). Guo et al. (2024) demonstrate that QMIX models experience significant performance degradation when subjected to adversarial attacks. Chen et al. (2024) reveal the vulnerability of DRL models under complex perturbations by introducing a nature player to generate adversarial training environments. Zhou et al. (2024b) show that the mean-field actor-critic model is highly sensitive to state perturbations, with adversarial attacks leading to a substantial decline in team rewards. Zhou et al. (2024a) explore adversarial attacks on multi-agent systems in continuous action spaces. When the DRL model is subjected to adversarial attacks, it can cause significantly different outputs from DRL-based application (Gleave et al., 2020; Hao and Tao, 2022; Qiaoben et al., 2024).

Modern power systems (e.g., smart grids) transmit a large amount of information in real time through communication networks to ensure coordinated generation, transmission, distribution, and consumption (Zhang et al., 023b, 024c). Many communication protocols used in power systems, such as Modbus and DNP3, have security vulnerabilities (Saxena et al., 2021; Moldovan and Ayyanar, 2024; Muhammad et al., 2024). In 2015, an attacker gained network access to a Ukrainian power company through phishing emails. They sent unencrypted control commands to shut down circuit breakers at multiple substations, causing widespread outages that affected more than 220,000 users (Nafees et al., 2023). In 2017, an attacker exploited the DNP3 protocol to infiltrate the U.S. power grid, using replay attacks or spoofed data packets to monitor and control the grid (Qassim et al., 2023).

The DRL-based controller, as a core component of the power system, is responsible for real-time monitoring and regulating the system's dynamic behavior. The stability and security of the power system are directly dependent on its decision-making process (Rasolomampionona et al., 2024; Zhang et al., 023a). If it is maliciously attacked, it could pose a serious threat to power system safety. Eexisting studies have recognized the threat of cyberattacks on DRL-based power system applications. Liu et al. (2023) employed the fast gradient sign method (FGSM) to attack the DRL controller in a single-area system, resulting in severe frequency fluctuations. Zheng et al. (2021) constructed the specified perturbations targeting DRL performance, leading to a significant decline in the DRL performance in power system topology optimization. However, most researchers focus on analyzing DRL vulnerabilities from the perspective of defenders with prior knowledge of the DRL model's parameters and structure, ignoring the perspective of attackers without such prior knowledge.

From the stance of a defender, this study examines the vulnerabilities of DRL controllers from the attacker's perspective to provide a more complete analysis. First, for an attacker, the parameters and structural information of the DRL-based adaptive controller are unknown, making it challenging to design adversarial attacks. Second, in power systems, it is difficult to obtain the necessary gradient information for the attack because the DRL controller cannot be directly queried. Third, designing a more robust controller can lead to overfitting, which reduces control performance. Therefore, training a robust DRL-based adaptive controller is complex because control performance and robustness must be managed. To address the model transparency issue,

Unedited

this paper introduces the zeroth-order optimization (ZOO) method. In the iterative process of generating adversarial samples, we use the ZOO method to approximate gradients, thereby avoiding reliance on traditional backpropagation. Additionally, by restoring the gradient information calculated using the ZOO method, the required gradient can be directly queried during the attack. Finally, by incorporating normal samples into the adversarial training dataset, we achieve improved robustness while maintaining control performance.

The contributions of this paper are as follows:

- A detailed analysis of the vulnerability of the DRL-based adaptive controller is conducted from the attacker's perspective.
- A black-box attack is designed by incorporating the ZOO method in adversarial attacks, enabling effective attacks on DRL-based adaptive controllers with unknown parameters.
- A new adversarial training paradigm is proposed that enhances the robustness of the DRL-based adaptive controller while maintaining control performance.

The remainder of this paper is organized as follows. Section 2 introduces the single-area LFC model and the DRL-based adaptive controller. Section 3 provides a detailed description of the framework and corresponding algorithms for constructing adversarial attacks using the ZOO method. Section 4 introduces adversarial training to defend against adversarial attacks. In Section 5, the vulnerability of the DRL controller and the robustness of adversarial training are verified using Python and Matlab's Simulink simulation platform. Section ?? concludes the paper.

2 System Model

2.1 LFC

The LFC mechanism for a single-area system, as depicted in Fig. 1, comprises essential components such as the generator, governor, and turbine. The control center monitors the frequency deviation Δf and sends the control signal ΔP_c through the communication network. The following equations represent the system dynamics for a single-area LFC model:

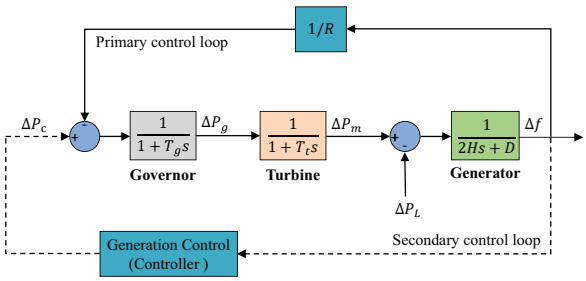

Fig. 1 Block diagram of the single-area LFC mode

$$\Delta \dot{f} = \frac{1}{2H}\left(\Delta P_m - \Delta P_L - D\Delta f\right) \tag{1}$$

$$\Delta \dot{P}_m = \frac{1}{T_t}(\Delta P_g - \Delta P_m) \tag{2}$$

$$\Delta \dot{P}_g = \frac{1}{T_g}(\Delta P_c - \Delta P_g - \frac{1}{R}\Delta f), \tag{3}$$

where ΔP_m and ΔP_g denote the mechanical power change and electrical power change, respectively. ΔP_L and ΔP_c denote load fluctuations and control signal T_g and T_t denote the governor time constant and turbine time constant, H, D and R denote the inertia constant, damping coefficient, and speed regulation ratio.

The LFC system typically operates based on feedback control principles. When a frequency deviation is detected, the system automatically adjusts the generator output to balance the load change. This process is primarily handled by primary and secondary control (Albeladi and Barati, 2023). Primary control independently performs automatic regulation through physical devices such as governors. Secondary control is the core component of a typical LFC system and relies heavily on communication networks and remote control commands. The regional control center receives frequency information Δf through communication channels and then generates a control signal Δp_c to regulate frequency within the area. During the process of receiving frequency information Δf at the control center, the attacks can inject disturbed signals through the communication network, misleading the control center into incorrect regulation.

2.2 DRL-based adaptive Controller

The PID controller is widely used in LFC systems to achieve enhanced control performance through parameter calibration. Recently, because

Unedited

DRL has demonstrated remarkable performance in control decision-making, many researchers have applied it to LFC systems. Because LFC is a continuous control problem, researchers have mainly focused on using classical DRL algorithms such as deep deterministic policy gradient and twin delayed deep deterministic policy gradient (TD3) (Fujimoto et al., 2018).

TD3 is a model-free and policy-based DRL algorithm specifically designed for control tasks with continuous action spaces. It consists of six neural networks: two actor and four critic networks. At each time step t, the actor network η generates the control command $a_t = \eta(s_t)$ based on the state s_t of the corresponding area. The action a_t interacts with the environment to obtain an immediate reward r_t which is preset according to the target objective. Typically, the objective of LFC is to minimize frequency deviations. Hence, the reward function r can be defined as follows:

$$r = -\varepsilon * |\Delta f|, \tag{4}$$

where ε is the scaling factor and $|\cdot|$ is the absolute value of a variable. To prevent DRL from focusing on short-term optimality, the cumulative reward $\sum_{t=1}^{T} \gamma^{t-1} r_t$ over time T is typically used for optimization. The parameter γ measures the importance of the cumulative reward relative to the immediate reward. The action-value function Q can be defined as $Q(s,a) = \sum_{t=1}^{T}[\gamma^{t-1} r_t | s = s_t, a = \eta(s_t)]$. In this study, the critic network is updated using temporal difference techniques (Maei, 2011), as shown by Eq (5):

$$\begin{aligned} y &= r + \gamma \min_{j=1,2} Q_{\theta'_j}(s', \tilde{a}) \\ \theta_j &\leftarrow \mathrm{argmin}_{\theta_j} \frac{1}{N} \sum_{i=1}^{N} (y - Q_{\theta_j}(s_i, a_i))^2, \end{aligned} \tag{5}$$

where s' and a_i are the next state and the action generated by the actor network η with s_i, $\tilde{a}$ is the action taken in the current state, y is the target value, θ_j denotes the parameters of the critic network, j is 1 or 2 and θ'_j denotes the parameters of the target critic network. Generally, using the experience replay technique, a fixed batch of data is randomly selected and the gradient ascent techniques are used to update the actor network. The gradient is updated based on the chain rule as shown in Eqs (6) and (7):

$$\nabla_{\theta^\eta} J \approx \frac{1}{N} \sum_{i=1}^{N} \left[\nabla_{a_i} Q(s_i, a_i)|_{a_i = \eta(s_i)} \nabla_{\theta^\eta} \eta(s_i) \right] \tag{6}$$

$$\theta^\eta \leftarrow \theta^\eta + \beta \nabla_{\theta^\eta} J, \tag{7}$$

where N, β, and s_i are the batch size, learning rate, and the state of the ith sample, respectively. $\nabla_{a_i} Q(s_i, a_i)$ is the gradient of the action-value function Q concerning the action a_i at state s_i. $\nabla_{\theta^\eta} \eta(s_i)$ is the gradient of the actor network. The objective function J, as given by Eq. (6), is used to update the parameters. Finally, the target network is updated using the soft update technique with a soft update factor τ:

$$\begin{aligned} \theta_i^{Q'} &\leftarrow \tau \theta_i^Q + (1-\tau)\theta_i^{Q'} \\ \theta^{\eta'} &\leftarrow \tau \theta^\eta + (1-\tau)\theta^{\eta'}, \end{aligned} \tag{8}$$

where Q', θ_i^Q, and $\theta_i^{Q'}$ are the parameters of the target critic network, the ith critic network, and the target critic network, respectively. θ^η and $\theta^{\eta'}$ are the parameters of the actor network and the target actor network, respectively.

3 Adversarial Attacks on the DRL-based controller

3.1 Vulnerability analysis

In modern power systems, a large amount of data is transmitted through communication networks, making it vulnerable to cyberattacks. For example, the Stuxnet virus can target supervisory control and data acquisition systems, allowing attackers to intercept and manipulate data during transmission from the sensing section to the control center (Chen, 2010). In 2010, an Iranian nuclear power plant was attacked by Stuxnet, resulting in 60% of its main systems failing (Yan et al., 2021). As shown in Fig. 2, in LFC, an attacker can use the Stuxnet virus to compromise the sensor and alter the frequency signal sent to the control center. Additionally, the attacker can employ protocol analysis tools to examine the network traffic between sensors and the control center. By understanding the data frame structure, the attacker can fabricate frequency signals and inject them into the communication network, causing the control center to receive false frequency information and make erroneous decisions (Mishchenko et al., 2024).

Unedited

DRL has demonstrated significant performance in solving frequency control problems, primarily due to the powerful approximation capabilities of DNNs. However, this also makes DRL inherently vulnerable to adversarial attacks. When a DRL-based adaptive controller is applied in an LFC system, a carefully crafted attack targeting the frequency data received by the control center could lead the power system to a dangerous state. Formally, this is represented as shown in Eq (9):

$$\begin{aligned} s_t^{adv} &= s_t + \sigma \\ a_t^{adv} &= \eta(s_t^{adv}|\theta^{\eta}), \end{aligned} \tag{9}$$

where σ is a specific perturbation, s_t^{adv} denotes the state resulting from the attack, and a_t^{adv} is the action generated from the corresponding state. When controlling the frequency of a power system using DRL agent, for a given state sample s, actor network η, and critic network Q, adversarial perturbation σ can be generated by solving the optimization problem presented in Eq (10):

$$\begin{aligned} &\min_{\sigma} Q(s, \eta(s+\sigma)) \\ &\text{s.t.,} \quad \sigma \in \mathcal{G} \end{aligned} \tag{10}$$

where σ and $\mathcal{G}$ denote the perturbation and the set of perturbations. Perturbations include both natural perturbations and malicious attacks. Because malicious attacks typically have a greater impact, we use them to represent all disturbances. It is worth noting that in DRL, Q and η are typically DNNs, making it difficult to obtain an analytical solution for the optimization problem. Fortunately, Q and η are differentiable. The perturbation σ can be iteratively solved using numerical optimization methods, such as gradient descent or projected gradient descent.

Because bad data detection can eliminate abnormal data caused by major disturbances, it is unnecessary to analyze cyberattacks that inject significant errors (Chaojun et al., 2015; Zhang et al., 2021). Therefore, we solve the optimization problem in Eq. (10) and make small but precise adjustments to the attack data.

3.2 Construction of the Adversarial Attack

We aimed to influence the decisions of the DRL-based adaptive controller by modifying the inputted state variables, as shown in Fig. 2. In preparation for the attack, the attacker needs to hack into the system

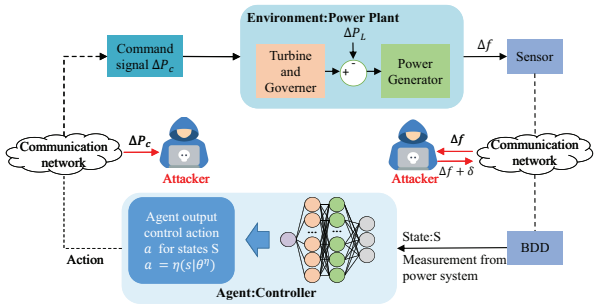

Fig. 2 DRL-based LFC for single-area system under the adversarial attacks. DRL, deep reinforcement learning; LFC, load frequency control.

and monitor the communication traffic between the data source and the control center through a man-in-the-middle proxy or by implanting malware. They also must be able to parse and manipulate packets. Because this is not the focus of our research, we assume the attacker already has system access and the capability to carry out the attack.

Once inside the system, the first challenge is to identify the manipulable state variable Δf from the controller's multiple input variables and designate it as the attack target. However, to modify Δf, we need to know the gradient. It is also challenging to obtain the gradient when the parameters and structure of the DRL controller are unknown. To address this, we introduced the ZOO method to estimate the model's gradients (Chen et al., 2017), but we further found that this method cannot be directly applied to dynamically changing power systems. Moreover, in a dynamic environment, attacks and gradient acquisition cannot be performed simultaneously. Thus, we divided the attack into two steps.

The first step was attack preparation. In a steady-state environment, the attacker monitors communications between the data source and the control center and collects Δf and ΔP_c data to calculate gradients. The attacker maintains a state-to-gradient mapping gradient list, storing the gradient information corresponding to different values of Δf. The second step was attack execution. During the attack, the attacker queries the gradient list to obtain the gradient value for Δf, generates new adversarial state variables, and manipulates the value of Δf. Finally, we demonstrated the effectiveness of the adversarial attack and analyzed its impact on system frequency stability. Our approach is more specifically described as follows.

Unedited

1) Identify variables to modify

Typically, attackers generate a new attack state s^{adv} by using adversarial attacks to tamper with the state s, thereby compromising the grid frequency. However, to improve the DRL model's understanding of LFC in power systems, the input states s have been empirically adjusted, such as by incorporating the integral and derivative of the frequency. These variables should not be directly modified. Therefore, the logical target for adversarial attacks should be the monitored variables Δf, not the empirically added integral of frequency $\int \Delta f$ or derivative of frequency $d\Delta f$, where $\int \cdot$ and $d\cdot$ are the integral and differential operators, respectively. As shown in Fig. 2, attackers impair the performance of the controller at the control center by tampering with the frequency deviation signal transmitted from the sensor to the control center. The mathematical form of the adversarial attack is given in Eq. (11):

$$\begin{aligned} &x_{n+1}^{adv} = \Pi_{x_0+\delta}(x_n^{adv} - \alpha sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv})))), \\ &x_n^{adv} = x_0,\ if\ n = 0, \end{aligned} \tag{11}$$

where the state s_0 is defined based on the observed variable x_0. $\Pi_{x_0+\delta}$ denotes a projection operator that limits the values to a permissible range around a base point s_0 with a perturbation δ. The $sign(\cdot)$ denotes the sign function, which is used to determine the direction of the gradient in each dimension. The variables n, α, and δ denote the number of iterations, maximum perturbation, and increment of perturbation in each iteration, respectively. $\nabla_{x_n^{adv}} Q(\cdot)$ denotes the gradient of the value function Q with respect to the states under adversarial attack.

2) Black-box Adversarial Attack Based on the ZOO Method

The adversarial attack requires attackers to have a deep understanding of the target model, including its structure and parameters. In the LFC system, such information is often not readily accessible due to concerns about sensitive infrastructure security. Therefore, the attacker should design the adversarial attack in a black-box form. In existing research, black-box adversarial attack methods mainly include transfer attacks, query-based attacks, and hybrid attacks (Zhang et al., 2022; Tian et al., 2022; Takiddin et al., 2023). Transfer attacks rely on substitute models to approximate the target model, but when there is significant model bias, the attack effect may significantly decrease. Additionally, substitute models may generate abnormal perturbations due to training data bias, which can be detected by detection algorithms. Hybrid attacks typically require a substitute model and partial gradient information, which can be seen as an optimization of transfer attacks. However, the closed nature of power systems makes it difficult to obtain the information required for such attacks. In contrast, query-based ZOO black-box adversarial attack methods estimate gradients by simulating random perturbations that closely resemble real noise, without relying on the target model's structural information, making them more suitable for closed power systems.

The ZOO method is adopted to derive the adversarial attack. As shown in Fig. 2, in the LFC system, the attacker can monitor the frequency deviation signal Δf sent from the sensors to the control center and the control signal ΔP_c sent from the control center to the region. The attacker can apply small random perturbations to the original input and use the definition of derivatives to estimate the gradients, thus inferring the gradient's sign. The mathematical form of ZOO attacks is given in Eq. (12). The detailed attack generation process is shown in Algorithm 1, in which we have

$$\nabla_x Q(s, \eta(s)) \approx \mathbb{E}[\frac{Q(s, \eta(s + \epsilon u)) - Q(s, \eta(s))}{\epsilon} u], \tag{12}$$

where ϵ is a predefined very small positive value. u is a distribution with a mean of zero and a covariance of the identity matrix. In a single-area LFC system, the reward function for training the controller should be aligned to minimize frequency deviation. The primary function of the critic network Q is to evaluate the quality of the action taken in the current state. Therefore, a well-trained critic network should be able to assess the value of a specific action accurately, given a particular state. In other words, the attacker can train a critic network to replace the critic network Q in the DRL-based adaptive controller. Consequently, it becomes possible to compute the gradient without knowing the specific parameters of the model.

Although an attacker can monitor the state input to the DRL-based adaptive controller and the actions generated by the controller, the dynamic nature of the LFC system, with continuously fluctuat-

Unedited

Algorithm 1 Adversarial attack on the DRL-based controller

Input: Initial δ, number of steps n, monitored variable x_0, the trained actor network η and the critic network Q.

Set initial state based on monitored variables x_0: $s_0 \leftarrow (\Delta f,\ \int \Delta f,\ d\Delta f)$
Set step size: $\alpha \leftarrow \delta/n$
for $k = 0$ to n **do**
  Set current state based on monitored variables x_k: $s_k \leftarrow (\Delta f_k,\ (\int \Delta f)_k,\ (d\Delta f)_k)$
  Calculating the gradient using the ZOO method: $\nabla_{x_k} Q(s_0, \eta(s_k)) \approx \mathbb{E}[\frac{Q(s_0,\eta(s_k+\epsilon u))-Q(s_0,\eta(s_k))}{\epsilon} u]$
  Update x_{k+1} based on the gradient: $x_{k+1} \leftarrow x_k - \alpha\,\mathrm{sign}(\nabla_{x_k} Q(s_0, \eta(s_k))$
  Project onto l^∞ -ball: $x_{k+1} \leftarrow \mathrm{clamp}(x_{k+1}, x_0 - \delta, x_0 + \delta)$
end for
Output: X_n

ing frequency, makes it challenging to use the ZOO method. The ZOO method requires two feedbacks from identical states: first, to obtain the control action in the initial state, and second, to get a new control action after adding a small perturbation to this state, thus allowing gradient calculation. In a dynamic environment, it is nearly impossible to obtain consecutive identical states or perform gradient computation and attack simultaneously. However, there exists a steady state in the power grid where frequency fluctuations are minimal, making two consecutive states nearly identical and suitable for gradient computation. Therefore, we divided the attack process into two steps.

First, the attacker monitors the data transmitted between the sensors and the control center in a steady state, collecting a large amount of system state Δf and control action ΔP_c data. The corresponding gradient information is calculated using the ZOO method. Because the parameters of the trained DRL controller are fixed, the attacker can precompute and store the gradient information corresponding to each state and create a Gradient list. Second, during the attack, the message Δf transmitted from the sensor to the control center is monitored and used as the injection point for the attack. The corresponding gradient for Δf is retrieved from the gradient list. The attack program intercepts the message carrying Δf, modifies its value based on the adversarial attack method, and delivers it to the controller, thereby inducing erroneous decisions.

3) Validity of the adversarial attack

To demonstrate the effectiveness of our adversarial attack on DRL-based adaptive controllers, we present the following theoretical proof. Adversarial attacks can iteratively adjust the attack vector through multiple iterations. Each step is based on the gradient information of the current perturbed input. Through multiple iterations, the obtained adversarial perturbation is more precise to mislead the model. For each iteration's perturbation $-\alpha \cdot sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))$, it always minimizes the Q value. The specific proof is shown in Eqs (13) and (14).

To ensure that $x_n^{adv} - \alpha \cdot sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))$ can reduce the Q value, we perform a Taylor expansion of Q at x_{n+1}^{adv} near x_n^{adv}:

$$\begin{aligned} Q(s_0, \eta(s_{n+1}^{adv})) \approx\ & Q(s_0, \eta(s_n^{adv})) + B \\ & \cdot (-\alpha \cdot sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))) \\ \approx\ & Q(s_0, \eta(s_n^{adv})) - B \cdot \alpha \\ & \cdot sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))), \end{aligned} \tag{13}$$

where B is $\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))^T$. Because the adversarial perturbations are small, $(x_{n+1}^{adv} - x_n^{adv})$ can be approximated as consistent with $(s_{n+1}^{adv} - s_n^{adv})$. Therefore, it can be determined that $(x_{n+1}^{adv} - x_n^{adv})$ is $-\alpha \cdot sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))$ in the Taylor expansion. T represents the transpose of the gradient vector, which converts it from a column vector to a row vector to facilitate dot products or matrix operations with subsequent vectors or scalars. If $\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))^T \cdot \alpha \cdot sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv})))$ is a non-negative value, then it can be demonstrated that the adversarial perturbation can reduce the Q value (i.e., increase the frequency fluctuation).

$$\begin{aligned} & \nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))^T \cdot \alpha \cdot sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))) \\ =& \alpha \cdot ||\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))||^2 / ||\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))|| \\ =& \alpha \cdot ||\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))|| \geq 0 \\ & where,\ sign(\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))) \\ =& \nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv})) / ||\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv}))||, \end{aligned} \tag{14}$$

where $||\cdot||$ denotes the L2 norm of a vector. Therefore, after injecting adversarial attacks into the observation variable x, the resulting x^{adv} can always

Unedited

reduce the following long-term reward Q:

$$Q = \sum_{t=1}^{T} \mathbb{E}_{a\sim\eta(s_t^{adv})} \gamma^{t-1} r_t, \tag{15}$$

the r_t values are the negative absolute frequency deviations at time steps t, and γ is a constant of 0.99. In other words, a decrease in Q implies an increase in frequency deviation, which means increased frequency fluctuations.

Remark 1. *x^{adv} is obtained by iteratively applying small perturbations. At each iteration, the perturbation is calculated by multiplying a step size α with the sign (+1 or-1) of the gradient $\nabla_{x_n^{adv}} Q(s_0, \eta(s_n^{adv})$, followed by a projection step to ensure the perturbation stays within the δ-ball around the original input. Here, the attack generation process can be seen as a gradient descent method that minimizes the state value Q(i.e., maximizes the frequency deviation), thereby degrading the controller's performance and significantly impacting the effectiveness of the DRL-based adaptive controller.*

4 The Mitigation Strategy against the Adversarial Attack

The DRL-based adaptive controller exhibits vulnerability when faced with adversarial attacks, highlighting the need for effective mitigation strategies to ensure system stability and security. To address this, we introduce adversarial training to enhance the controller's robustness and performance under adversarial attack (Tan et al., 2020).

For the DRL-based frequency control problem, the agent's goal is to select the actions from all possible options that maximize the long-term reward function Q. Typically, we assume a stable environment, meaning the LFC system parameters remain unchanged. However, when the DRL-based adaptive controller faces attacks and disturbances, it can be seen as a change in the LFC environment. This change may cause the DRL controller trained in a stable environment to fail in the new, attacked environment, leading to poor frequency control. Therefore, it is necessary to retrain the DRL controller to ensure it maintains good performance even in the attacked and disturbed environment. To solve this, we employ adversarial training to enhance the robustness of the DRL controller (Pattanaik et al., 2018; Madry et al., 2018; Jia et al., 2022). We use extreme perturbation adversarial samples as the training set, enabling the controller to develop stronger robustness and adaptability during operation. This not only defends against adversarial attacks but also mitigates natural disturbances. The detailed adversarial training process is outlined in algorithm 2.

Algorithm 2 Adversarial training against the adversarial attack.

Data: Trained TD3 agent with networks η, η', Q_1, Q_1', Q_2, Q_2', number of steps n, batch size N, perturbation magnitude δ, replay buffer R^{adv} and demonstration buffer R with trajectories from the normal data.

for $episode = 1$ to m **do**
  Reset the environment.
  for $t = 1$ to T **do**
    Set initial monitored variables x_t.
    Perturb the monitored variables:
    $x_t^{adv} \leftarrow$ adversarial attacks(x_t,η,Q,δ,n)
    Set $s_t = (\Delta f, \int \Delta f, d\Delta f)$
    Obtain action a_t=$\eta(x_t^{adv})$
    Using a_t to obtain r_t and s_{t+1}
    Store transition (s_t^{adv},a_t,r_t,s_{t+1}) in R^{adv}
    Sample a minibatch of $0.7N$ transitions (s_i^{adv},a_i,r_i,s_{i+1}) from R^{adv} and $0.3N$ transitions (s_i,a_i,r_i,s_{i+1}) from R.
    Update critic networks Q_1 and Q_2
    if every policy delay step **then**
      Update actor network η
      Update target networks η', Q_1', and Q_2':
      $\theta_1^{Q'} \leftarrow \tau\theta_1^{Q} + (1-\tau)\theta_1^{Q'}$
      $\theta_2^{Q'} \leftarrow \tau\theta_2^{Q} + (1-\tau)\theta_2^{Q'}$
      $\theta^{\eta'} \leftarrow \tau\theta^{\eta} + (1-\tau)\theta^{\eta'}$
    end if
  end for
end for

In the adversarial training process, we find that training solely on adversarial samples enhances robustness against adversarial attacks but significantly reduces control performance in the absence of attacks. This may be due to the DRL model overfitting to adversarial samples, resulting in poorer control of normal data. To address this issue, we incorporate a portion of normal data into adversarial training to prevent the model from overlearning adversarial patterns. As shown in algorithm 2, we use two replay buffers—one for adversarial samples and the other

Unedited

for normal samples. During sample selection, we use 70% adversarial samples and 30% normal data, thus managing improved robustness with stable control performance in non-adversarial conditions and preventing overfitting. Ultimately, the agent demonstrates strong robustness when facing adversarial attacks and maintains excellent control performance in the absence of attacks.

5 Simulation Results

5.1 Training DRL-based adaptive controller

We conduct simulation tests on a single-area power system using the deep learning framework PyTorch 2.2 and the Simulink toolbox of MATLAB 2024. The parameters of the single-area LFC model are provided in (Pandey et al., 2020). First, we train a DRL-based adaptive controller. For the training of the controller, the size of the memory replay buffer, the discount factor γ, the learning rate α, and the soft update factor τ are set to 10000, 0.99, 0.01, and 0.005, respectively. We set the initial noise to 0.3 and apply a high decay rate of 0.1. This allows sufficient exploration of actions in the early stage of training while not affecting the convergence of the reward in the later stages.

Unedited

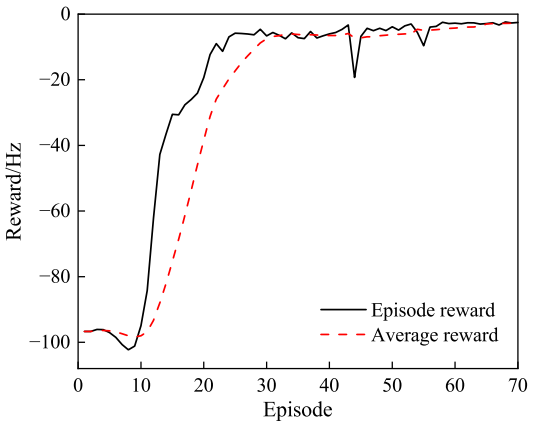

Fig. 3 During training of the DRL

The DRL training process is shown in Fig. 3. We take the cumulative reward of all time steps within an episode as the episode reward and compute the moving average over intervals of 10 episodes as the average reward. As the number of training episodes increases, the DRL-based controller performance gradually improves. After approximately 30 training episodes, the reward value converges and stabilizes around −2.6, indicating that the model has completed learning and achieved good control performance.

To compare the performance of the DRL-based controller, we separately train a PID controller and a DNN controller. The parameters of the PID controller are tuned using MATLAB's Control Toolbox. The DNN is trained on data collected under normal operating conditions, which include frequency deviation information and the corresponding control signals. After several rounds of iterative training, the DNN is able to adjust the control signal based on the input frequency deviation, thereby achieving stable frequency control of the system. Then, we analyze the performance of the DRL-based controller by comparing it with PID and DNN controllers. Over the period from 0 s to 60 s, perturbations of −0.01 p.u., 0.01 p.u., 0 p.u., −0.01 p.u., 0 p.u., and −0.02 p.u. are applied to the system every 10 s. The control effects on frequency deviation for both controllers under these disturbance conditions are presented in Fig. 4. As illustrated in Fig. 4, all three controllers are able to effectively regulate system frequency in response to load fluctuations, and their control performance remains relatively stable. However, the DRL-based controller demonstrates superior performance in terms of frequency regulation speed and accuracy, with faster settling time and smaller overshoot.

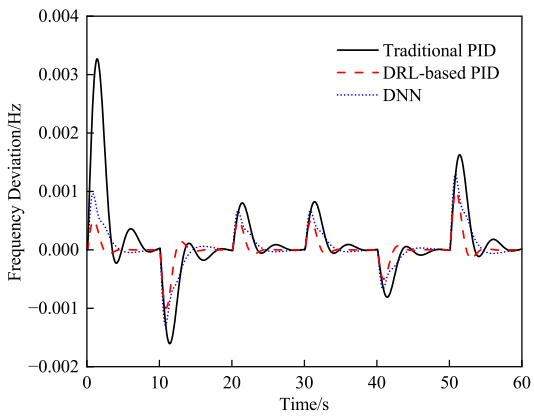

Fig. 4 Dynamic frequency deviation under step disturbance conditions. DNN, deep neural network; DRL, deep reinforcement learning; PID, proportional integral derivative.

5.2 Performance of Adversarial Attacks for the DRL-based adaptive controller

Typically, attacks are more covert during periods of significant disturbance. Therefore, we use Gaussian noise with a variance of 0.02 to simulate a random environment, and attack experiments are conducted under these conditions. The gradient information required for the attack is obtained using the ZOO method. First, we test the accuracy of the ZOO method in obtaining gradient information in this environment. Because the attack method used in this study only requires gradient signs, we focus solely on the accuracy of the ZOO method in calculating the gradient signs of the states. By comparing the gradient signs of states calculated by the ZOO method and the backpropagation method every 0.1 s over 60 s, we find that the ZOO method performs well. As shown in Fig. 5, the ZOO method achieves high accuracy in calculating the gradient signs of state variables Δf, $\int \Delta f$, and $d\Delta f$. The accuracy of gradients for all state variables is over 92%, indicating that the ZOO method can be effectively integrated into adversarial attacks to achieve black-box attacks.

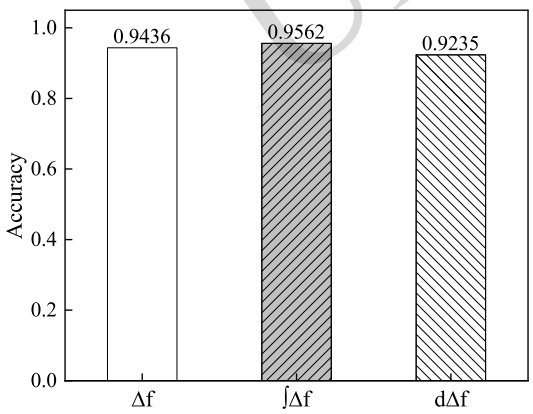

Fig. 5 The accuracy of the ZOO method in computing the gradient signs of the three state variables. ZOO, zeroth-order optimization.

Second, we evaluate the impact of the Q network structure on the effectiveness of the attack. We design and train three Q networks with different complexities: Q1 (3 layers, 16 neurons per layer), Q2 (4 layers, 32 neurons per layer), and Q3 (5 layers, 128 neurons per layer). Using the Q2 network to generate FGSM adversarial examples, we perform adversarial attacks on controllers based on the Q1, Q2, and Q3

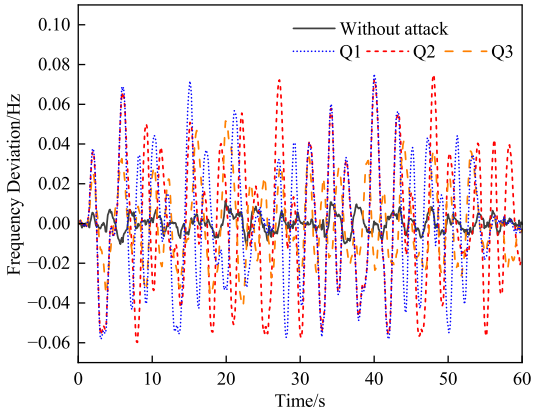

Fig. 6 The impact of different critic Q network structure on adversarial attacks.

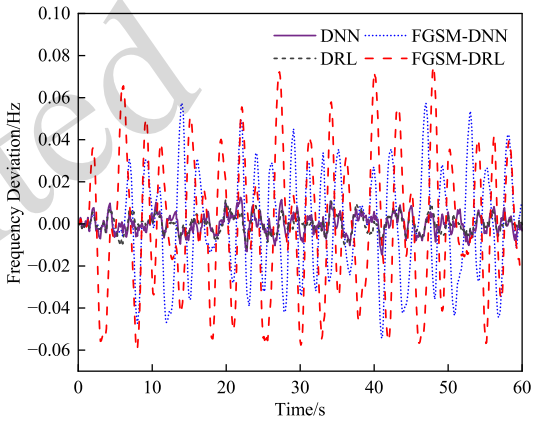

Fig. 7 The control performance of the DNN and DRL controllers under FGSM attacks. DNN, deep neural network; DRL, deep reinforcement learning; FGSM, fast gradient sign method.

networks, respectively. As shown in Fig. 6, because Q2 generates adversarial examples to attack its own controller, the attack effect is the most significant. For the relatively simple Q1 network, the attack still shows a strong effect. In contrast, the more complex Q3 network shows a weaker attack effect, but still demonstrates some vulnerability. This indicates that the network structure does not significantly affect the attack's effectiveness; as long as the network can effectively estimate the state-action values, the attack can still have a certain level of impact. Third, we compare the performance of the DNN and DRL controllers when facing FGSM adversarial attacks. As shown in Fig. 7, although the DRL controller has excellent performance under normal conditions, its performance deteriorates when subjected to adversarial attacks, showing vulnerabilities similar to those of the DNN controller. This highlights the potential vulnerabilities of DRL in real-world appli-

Unedited

Table 1 Comparison of frequency deviation responses under FGSM and PGD

σ	FGSM		PGD		Without attack	
	MAFD (%)	LVFD (p.u.)	MAFD (%)	LVFD (p.u.)	MAFD (%)	LVFD (p.u.)
0.01	0.42	0.0136	0.38	0.0136	0.29	0.0091
0.03	0.56	0.0168	0.51	0.0168	0.29	0.0091
0.06	1.03	0.0363	1.41	0.0361	0.29	0.0091
0.09	1.37	0.0443	1.69	0.0521	0.29	0.0091
0.12	1.48	0.04914	2.00	0.0601	0.29	0.0091

FGSM, fast gradient sign method; MAFD: mean absolute of frequency deviation; LVFD: largest variation of frequency deviation; PGD, projected gradient descent.

cations, especially when facing adversarial attacks.

Finally, we employ two classic adversarial attack methods, FGSM and PGD, to conduct attack experiments on the DRL-based adaptive controller (Madry et al., 2018). As shown in Fig. 8, under random disturbances alone, the controller exhibits good performance. Due to the continuous presence of random disturbances, the controller cannot fully stabilize the frequency deviation at zero, but the fluctuation range remains small, generally within ±0.005. However, when adversarial attacks are injected, the controller's performance significantly deteriorates.

In Fig. 8a, both FGSM and PGD cause an increase in the frequency deviation fluctuation, leading to a decline in control performance. Interestingly, in this figure, the performance of FGSM and PGD are similar. This may be attributed to the fact that the experiment controls a single-area LFC system, where the neural network used is relatively simple, and the attack magnitude is small. As the perturbation increased, the performance of the DRL-based controller noticeably worsened. In Figs 8b–8c, when the perturbations are 0.06 and 0.12, respectively, the controller's performance degrades significantly, and the frequency deviation fluctuations become more severe. Particularly at the threshold of 0.12, the controller shows considerable vulnerability to adversarial attacks. Table 1 further quantifies the impact of adversarial attacks on DRL-based controller performance using numerical data. In scenarios without adversarial attacks, the DRL-based adaptive controller achieves a mean absolute frequency deviation of 0.29% and a maximum deviation of 0.0091 p.u. However, the performance of the controller deteriorates drastically under adversarial attacks. At a perturbation of 0.03, FGSM and PGD attacks cause the mean absolute deviation and peak deviation to rise to 0.56%, 0.0168 p.u. and 0.51%, 0.0168 p.u., respectively. Compared to the scenario without attacks, the mean absolute deviation increased by 93.10% and 75.86%, while the peak deviation increased by 84.61% for both attacks. As the perturbations increase, both deviations continue to rise, and control performance deteriorates further. At a perturbation of 0.12, the PGD attack results in the mean absolute deviation and peak deviation increasing to 2.00% and 0.0601 p.u., representing increases of 589.66% and 560.43%, respectively, compared to the no-attack scenario. These results indicate that the control performance of the DRL-based controller is highly vulnerable to adversarial attacks and can be significantly compromised.

5.3 Performance of adversarial training

Finally, we conducted adversarial training on the DRL-based adaptive controller. Although the control performance of the trained controller slightly decreased, with an average absolute frequency deviation of 0.33% and the largest deviation of 0.0114 p.u. (compared to 0.29% and 0.0091 p.u. for the controller without adversarial training), it demonstrated improved robustness against adversarial attacks (PGD and FGSM), as shown in Fig. 9. Particularly, when the attack intensity σ is 0.03, the adversarially trained controller demonstrates good defense performance. As depicted in Fig. 9a, despite PGD or FGSM attacks, the controller still maintains stable control over frequency deviation. Compared to the no-attack case, the frequency fluctuation slightly increases, but the peak deviation remains within the ±0.016 p.u. range, with the mean absolute deviation being less than 0.38%.

When the attack intensity σ increases to 0.06, as shown in Fig. 9b, the defense effect begins to weaken. At this point, PGD attacks start to increase the frequency deviation fluctuations, with the mean

Unedited

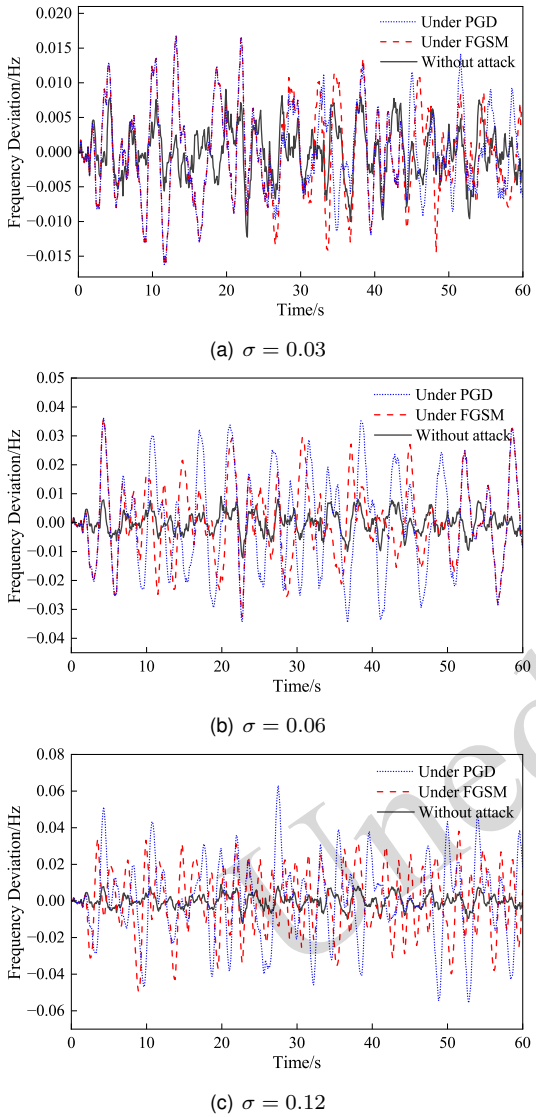

Fig. 8 Comparing the performance of the DRL-based controller under different attack types and varying perturbations. (a), (b), and (c) are the control performance of the controller under PGD and FGSM attacks with perturbations of 0.03, 0.06, and 0.12, respectively. DRL, deep reinforcement learning; FGSM, fast gradient sign method; PGD, projected gradient descent.

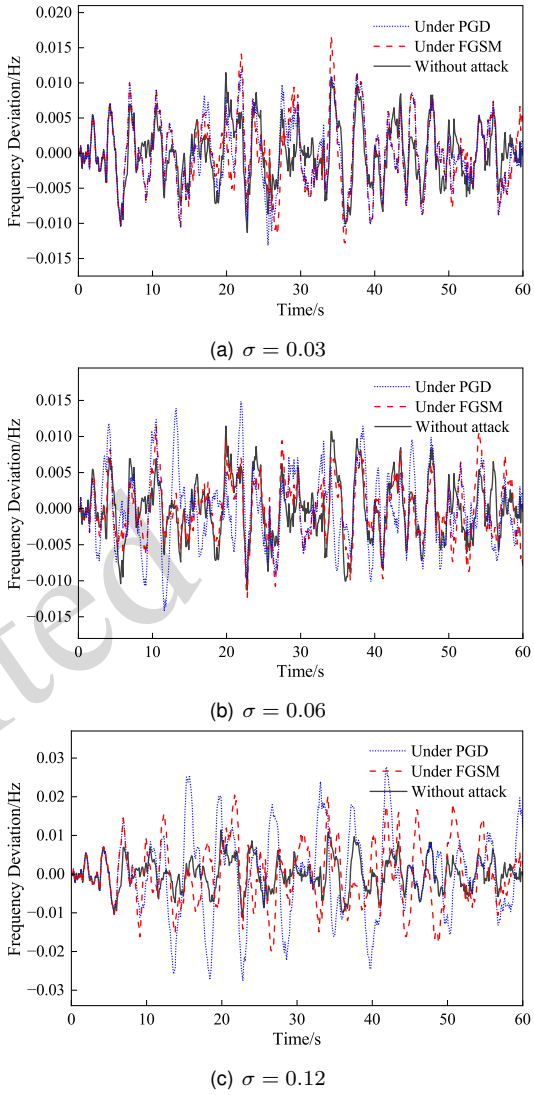

Fig. 9 The performance of the DRL-based controller after adversarial training under adversarial attacks. (a), (b), and (c) are the control performance of the controller under PGD and FGSM attacks with perturbations of 0.03, 0.06, and 0.12, respectively. DRL, deep reinforcement learning; FGSM, fast gradient sign method; PGD, projected gradient descent.

absolute deviation rising to 0.43%, but the peak deviation remains within the ±0.016 p.u. range. When the attack intensity reaches 0.12, as shown in Fig. 9c, the impact of adversarial attacks on the controller significantly increases, and the defense effectiveness becomes relatively weaker. At this stage, FGSM and PGD attacks increase the peak deviation and average absolute deviation to 0.021 p.u., 0.64% and 0.28 p.u., 0.84%, respectively. However, compared to the untrained controller (with peak deviation and average absolute deviation increases of 0.049 p.u., 1.48% and 0.06 p.u., 2%), the adversarially trained controller still exhibits strong robustness.

To compensate for the insufficient defense capability when the attack intensity σ exceeds 0.06, we introduce a lightweight attack detection mechanism based on changes in Q-values to monitor potential abnormal behavior in real time. The Q-value, a core indicator in reinforcement learning models, directly reflects the agent's evaluation of and preference for different actions in a given state. Meanwhile, the ad-

Unedited

versarial strategy proposed in this paper explicitly minimizes the Q-value to generate adversarial samples, thus misleading the agent into making incorrect decisions.

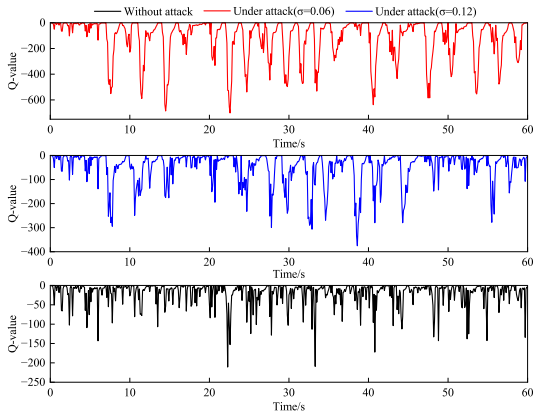

Fig. 10 The variation of Q-values under different attack intensities and no attack

When an adversarial attack occurs, Q-values exhibit significant abnormal fluctuations that differ markedly from the fluctuations observed under normal conditions. As shown in Fig. 10, the Q-value remains relatively stable and exhibits small fluctuations when the controller is not under attack. However, when the attack intensity σ exceeds 0.06, the Q-value shows pronounced fluctuation, with both fluctuation magnitude and instability increasing significantly. Based on this behavior, we apply a moving average to smooth the Q-value sequence and define a threshold for attack detection. If the deviation of the Q-value's moving average exceeds this threshold, an alert is triggered, indicating that the system may be under adversarial attack. Specifically, we use three times the standard deviation of the moving average at attack intensity σ 0.06 as the threshold for detecting adversarial attacks. This ensures sufficient sensitivity to Q-value anomalies while avoiding interference from normal environmental variations.

To evaluate the applicability of this detection mechanism under different attack intensities, we conduct detection experiments with attack intensities σ of 0.06, 0.09, and 0.12. As illustrated in Fig. 11, the proposed method effectively identifies abnormal fluctuations in Q-values across all tested attack intensities. Notably, when the attack intensity exceeds 0.09, the deviation in Q-values becomes more pronounced, and the anomaly features are more evident. In these

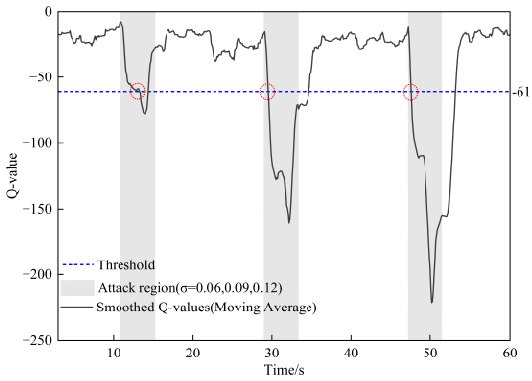

Fig. 11 Attack detection based on smoothed Q-value fluctuations under different attack intensities

cases, the detection algorithm can promptly identify the fluctuations and signal potential attack behavior. These results demonstrate that the proposed Q-value-based detection mechanism exhibits good sensitivity and practicality in high-intensity attack scenarios. It can effectively enhance the operational security of the system.

Unedited

6 Conclusions

Given that attackers typically face unknown model parameters and structures, we introduced the ZOO method to enable effective adversarial attacks in a black-box setting, thereby simulating real-world attack threats. First, we designed an adaptive controller based on DRL. Second, we conducted a vulnerability analysis of DRL-based adaptive control using adversarial attacks combined with the ZOO method. Finally, experimental results showed that adversarial attacks combined with the ZOO method significantly reduced the control effectiveness of the DRL-based adaptive controller. Based on an analysis of the attack effects, we adopted adversarial training as a defense strategy, enhancing the robustness and security of DRL controllers against potential adversarial attacks. In the future, we will analyze the vulnerabilities of multi-area LFC systems and propose more robust defense measures.

Contributors

Wei WANG and Zhenyong ZHANG designed the research, processed the data,and drafted the paper. Zhenyong ZHANG helped organize the paper. Zhenyong ZHANG, Xin WANG, and Xuguo JIAO revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Albeladi F, Barati M, 2023. Grid-supportive load frequency control using deep reinforcement learning. Proc IEEE Kansas Power and Energy Conference, p.1-5. https://doi.org/10.1109/KPEC58008.2023.10215451

Behzadan V, Munir A, 2017. Vulnerability of deep reinforcement learning to policy induction attacks. Proc 13th Int Conf on Machine Learning and Data Mining in Pattern Recognition, p.262-275.

Chaojun G, Jirutitijaroen P, Motani M, 2015. Detecting false data injection attacks in ac state estimation. *IEEE Trans Smart Grid*, 6(5):2476-2483. https://doi.org/10.1109/TSG.2015.2388545

Chen PY, Zhang H, Sharma Y, et al., 2017. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. Proc 10th ACM Workshop on Artificial Intelligence and Security, p.15-26. https://doi.org/10.1145/3128572.3140448

Chen ST, Liu GJ, Zhou ZY, et al., 2024. Robust multi-agent reinforcement learning method based on adversarial domain randomization for real-world dual-uav cooperation. *IEEE Trans Intell Veh*, 9(1):1615-1627. https://doi.org/10.1109/TIV.2023.3307134

Chen TM, 2010. Stuxnet, the real start of cyber warfare? [editor's note]. *IEEE Network*, 24(6):2-3. https://doi.org/10.1109/MNET.2010.5634434

Doan DV, Nguyen K, Thai QV, 2022. Load-frequency control of three-area interconnected power systems with renewable energy sources using novel pso pid-like fuzzy logic controllers. *Eng, Technol Appl Sci Res*, 12(3):8597-8604. https://doi.org/10.48084/etasr.4924

Dogru O, Velswamy K, Ibrahim F, et al., 2022. Reinforcement learning approach to autonomous pid tuning. *Comput Chem Eng*, 161:107760. https://doi.org/10.1016/j.compchemeng.2022.107760

Fujimoto S, Hoof H, Meger D, 2018. Addressing function approximation error in actor-critic methods. Proc 35th Int Conf on Machine Learning, p.1582-1591.

Gleave A, Dennis M, Wild C, et al., 2020. Adversarial policies: attacking deep reinforcement learning. *Proc 8th Int Conf on Learning Representations*, .

Guo WR, Liu GJ, Zhou ZY, et al., 2024. Enhancing the robustness of qmix against state-adversarial attacks. *Neurocomputing*, 572:127191. https://doi.org/10.1016/j.neucom.2023.127191

Hao JB, Tao Y, 2022. Adversarial attacks on deep learning models in smart grids. *Energy Rep*, 8:123-129. https://doi.org/10.1016/j.egyr.2021.11.026

Jia XJ, Zhang Y, Wu BY, et al., 2022. Las-at: adversarial training with learnable attack strategy. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.13388-13498. https://doi.org/10.1109/CVPR52688.2022.01304

Liu XH, Jiao QM, Yan ZM, 2023. Load frequency control with deep reinforcement learning under adversarial attacks. Proc 18th Conf on Industrial Electronics and Applications, p.257-262. https://doi.org/10.1109/ICIEA58696.2023.10241803

Madry A, Makelov A, Schmidt L, et al., 2018. Towards deep learning models resistant to adversarial attacks. *Proc 6th Int Conf on Learning Representations*, .

Maei HR, 2011. Gradient temporal-difference learning algorithms. *University of Alberta, Edmonton, Canada*, .

Michel A, Jha SK, Ewetz R, 2022. A survey on the vulnerability of deep neural networks against adversarial attacks. *Prog Artif Intel*, 11(2):131-141. https://doi.org/10.1007/s13748-021-00269-9

Mishchenko D, Oleinikova I, Erdődi L, et al., 2024. Multidomain cyber-physical testbed for power system vulnerability assessment. *IEEE Access*, 12:38135-38149. https://doi.org/10.1109/ACCESS.2024.3375401

Moldovan D, Ayyanar R, 2024. Dnp3 implementation in a high der penetration distribution system. Proc IEEE Kansas Power and Energy Conf, p.1-5. https://doi.org/10.1109/KPEC61529.2024.10676137

Muduli R, Jena D, Moger T, 2025. Application of reinforcement learning-based adaptive pid controller for automatic generation control of multi-area power system. *IEEE Trans Automat Sci Eng*, 22:1057-1068. https://doi.org/10.1109/TASE.2024.3359219

Muhammad MS, Alshra'a AS, German R, 2024. Survey of cybersecurity in smart grids protocols and datasets. *Procedia Comput Sci*, 241:365-372. https://doi.org/10.1016/j.procs.2024.08.049

Nafees MN, Saxena N, Cardenas A, et al., 2023. Smart grid cyber-physical situational awareness of complex operational technology attacks: A review. *ACM Comput Surv*, 55(10):215. https://doi.org/10.1145/3565570

Pandey SK, Gupta P, Dwivedi SS, 2020. Full order observer based load frequency control of single area power system. Proc 12th Int Conf on Computational Intelligence and Communication Networks, p.239-242. https://doi.org/10.1109/CICN49253.2020.9242561

Pattanaik A, Tang ZY, Liu SJ, et al., 2018. Robust deep reinforcement learning with adversarial attacks. *Proc 17th Int Conf on Autonomous Agents and MultiAgent Systems*, :2040-2042.

Qassim QS, Ali MAM, Tahir NM, 2023. Security analysis of dnp3 protocol in scada system. Proc 13th Int Conf on Control System, Computing and Engineering, p.314-319. https://doi.org/10.1109/ICCSCE58721.2023.10237142

Qiaoben Y, Ying CY, Zhou XN, et al., 2024. Understanding adversarial attacks on observations in deep reinforcement learning. *Sci China Inf Sci*, 67(5):152104. https://doi.org/10.1007/s11432-021-3688-y

Raju GV, Srikanth NV, 2024. Frequency control of an islanded microgrid with multi-stage pid control approach using moth-flame optimization algorithm. *Electr Eng*, . https://doi.org/10.1007/s00202-024-02518-1

Unedited

Rasolomampionona DD, Połecki M, Zagrajek K, et al., 2024. A comprehensive review of load frequency control technologies. *Energies*, 17(12):2915.
https://doi.org/10.3390/en17122915

Saxena S, Bhatia S, Gupta R, 2021. Cybersecurity analysis of load frequency control in power systems: a survey. *Designs*, 5(3):52.
https://doi.org/10.3390/designs5030052

Shabani H, Vahidi B, Ebrahimpour M, 2013. A robust pid controller based on imperialist competitive algorithm for load-frequency control of power systems. *ISA Trans*, 52(1):88-95.
https://doi.org/10.1016/j.isatra.2012.09.008

Shangguan XC, Zhang CK, He Y, et al., 2021. Robust load frequency control for power system considering transmission delay and sampling period. *IEEE Transactions on Industrial Informatics*, 17(8):5292-5303.
https://doi.org/10.1109/TII.2020.3026336

Shi Hr, Liu Gj, Zhang Kw, et al., 2023. Marl sim2real transfer: Merging physical reality with digital virtuality in metaverse. *IEEE Trans Syst, Man, Cybern: Syst*, 53(4):2107-2117.
https://doi.org/10.1109/TSMC.2022.3229213

Shuprajhaa T, Sujit SK, Srinivasan K, 2022. Reinforcement learning based adaptive pid controller design for control of linear/nonlinear unstable processes. *Appl Soft Comput*, 128:109450.
https://doi.org/10.1016/j.asoc.2022.109450

Takiddin A, Ismail M, Serpedin E, 2023. Robust data-driven detection of electricity theft adversarial evasion attacks in smart grids. *IEEE Trans Smart Grid*, 14(1):663-676.
https://doi.org/10.1109/TSG.2022.3193989

Tan KL, Esfandiari Y, Lee XY, et al., 2020. Robustifying reinforcement learning agents via action space adversarial training. Proc American Control Conf, p.3959-3964.
https://doi.org/10.23919/ACC45564.2020.9147846

Tian Jw, Wang Bh, Li J, et al., 2022. Adversarial attacks and defense for cnn based power quality recognition in smart grid. *IEEE Trans Network Sci Eng*, 9(2):807-819.
https://doi.org/10.1109/TNSE.2021.3135565

Xue J, Liu Zn, Liu Gj, et al., 2024. Robust wind-resistant hovering control of quadrotor uavs using deep reinforcement learning. *IEEE Trans Intell Veh*, .
https://doi.org/10.1109/TIV.2023.3324687

Yan S, Gu ZH, Park JH, 2021. Memory-event-triggered H_∞ load frequency control of multi-area power systems with cyber-attacks and communication delays. *IEEE Trans Network Sci Eng*, 8(2):1571-1583.
https://doi.org/10.1109/TNSE.2021.3064933

Yan ZM, Xu Y, 2019. Data-driven load frequency control for stochastic power systems: A deep reinforcement learning method with continuous action search. *IEEE Trans Power Syst*, 34(2):1653-1656.
https://doi.org/10.1109/TPWRS.2018.2881359

Zhang LH, Jiang CM, Pang AP, 2022. Black-box attacks and defense for dnn-based power quality classification in smart grid. *Energy Rep*, 8:12203-12214.
https://doi.org/10.1016/j.egyr.2022.09.032

Zhang ZY, Deng RL, Yau DK, et al., 2021. Zero-parameter-information data integrity attacks and countermeasures in iot-based smart grid. *IEEE Internet Things J*, 8(8):6608-6623.
https://doi.org/10.1109/JIOT.2021.3049818

Zhang ZY, Yang ZB, Yau DK, et al., 2023a. Data security of machine learning applied in low-carbon smart grid: A formal model for the physics-constrained robustness. *Appl Energy*, 347:121405.
https://doi.org/10.1016/j.apenergy.2023.121405

Zhang ZY, Deng Rl, Tian Yl, et al., 2023b. Spma: Stealthy physics-manipulated attack and countermeasures in cyber-physical smart grid. *IEEE Trans Inf Forensics Secur*, 18:581-596.
https://doi.org/10.1109/TIFS.2022.3226868

Zhang ZY, Yang KD, Tian YL, et al., 2024a. An anti-disguise authentication system using the first impression of avatar in metaverse. *IEEE Trans Inf Forensics Secur*, 19:6393-6408.
https://doi.org/10.1109/TIFS.2024.3410527

Zhang ZY, Deng RL, Yau DK, 2024c. Vulnerability of the load frequency control against the network parameter attack. *IEEE Trans Smart Grid*, 15(1):921-933.
https://doi.org/10.1109/TSG.2023.3275988

Zhang Z, Liu M, Sun M, et al., 2024. Vulnerability of machine learning approaches applied in iot-based smart grid: A review. *IEEE Internet of Things Journal*, 11(11):18951-18975.
https://doi.org/10.1109/JIOT.2024.3349381

Zheng Y, Yan ZM, Chen KJ, et al., 2021. Vulnerability assessment of deep reinforcement learning models for power system topology optimization. *IEEE Trans Smart Grid*, 12(4):3613-3623.
https://doi.org/10.1109/TSG.2021.3062700

Zhou ZY, Liu GJ, Guo WR, et al., 2024a. Adversarial attacks on multiagent deep reinforcement learning models in continuous action space. *IEEE Trans Syst, Man, Cybern: Syst*, 54(12):7633-7646.
https://doi.org/10.1109/TSMC.2024.3454118

Zhou ZY, Liu GJ, Zhou MC, 2024b. A robust mean-field actor-critic reinforcement learning against adversarial perturbations on agent states. *IEEE Trans Neural Networks Learn Syst*, 35(10):14370-14381.
https://doi.org/10.1109/TNNLS.2023.3278715

Unedited