Energy Engineering

Tech Science Press

Doi:10.32604/ee.2025.073445

ARTICLE

A Power System Preventive Control Method Based on Generative Adversarial Proximal Policy Optimization

Yun Yu[1], Li Lin[2,*], Ximing Zhang[1], Yang Yu[3], Wei Zhang[2] and Kai Cheng[3]

[1]China Southern Power Grid Co., Ltd., Guangzhou, 510663, China
[2]Yunnan Electric Power Co., Ltd., Kunming, 650041, China
[3]China Southern Power Grid Digital Grid Research Institute Co., Ltd., Guangzhou, 510555, China

*Corresponding Author: Li Lin. Email: linli_2025@126.com

Received: 18 September 2025; Accepted: 17 November 2025; Published: 18 June 2026

ABSTRACT: Traditional transient stability preventive control calculation methods suffer from low computational efficiency, struggling to meet the real-time decision demands of increasingly large-scale power systems. Meanwhile, reinforcement learning-based preventive control approaches, which adopt an "offline training, online application" framework, show greater promise in preventive control. However, they still face challenges such as low computational efficiency in electromechanical transient simulation and insufficient decision robustness. Therefore, this paper proposes a power system predictive control strategy based on Generative Adversarial Proximal Policy Optimization (GA-PPO). Firstly, considering multiple constraints in transient stability operation, a power system preventive control model is constructed with the objective of minimizing the total amount of adjustments, along with its Markov Decision Process (MDP) formulation. Then, the discriminator of Generative Adversarial Network (GAN) measures the gap between the expert demonstration distribution and the generated trajectory distribution, providing correction parameters for the advantage function of the Proximal Policy Optimization (PPO) algorithm, enhancing the agent's exploration efficiency. Finally, the discriminator's update mechanism is enhanced by Wasserstein distance, ensuring more stable training while enabling continuous adversarial interaction between discriminator and generator to explore higher convergent rewards. Case studies demonstrate that the proposed GA-PPO algorithm significantly reduces training time and achieves higher convergent rewards compared to PPO and Soft Actor-Critic (SAC) algorithms.

KEYWORDS: Generative adversarial; proximal policy optimization; transient stability; preventive control

1 Introduction

Currently, the transformation pathways and development paradigms of power systems have become a focal research area for scholars worldwide [1]. However, with the continuous expansion of power system scale and the high penetration of power electronics-based energy sources, the safe and stable operation of power systems faces unprecedented challenges [2]. In addressing voltage and frequency stability issues, Mejia-Ruiz et al. [3] proposed a novel hierarchical optimal control framework integrated with battery energy storage systems to support both frequency and voltage in multi-area transmission systems. Furthermore, literature [4] introduced a real-time co-simulation framework to experimentally validate the dynamic performance of network-level controllers in power systems. As one of the three core stability problems in power systems, transient stability [5] has been analyzed by researchers solving optimal power flow problems with transient stability constraints based on time-domain simulation methods or direct methods. Nevertheless, the time-consuming and labor-intensive time-domain simulation method, coupled with the

overly conservative direct method, struggles to meet the online decision-making requirements of today's large-scale power grids [6].

The rapid advancement of artificial intelligence in recent years and its applications in power system forecasting [7] and economic dispatch [8] have introduced novel solution paradigms for transient stability research, such as state identification [9] and early warning [10], transient stability assessment [11], preventive control [12], and emergency control [13]. Addressing preventive control challenges in power systems, Liu et al. [14] derived the Transient Stability Index (TSI) based on deep belief networks and integrated it with steady-state dispatch. By linearizing transient stability constraints and employing an interior-point method, they obtained an iterative control path for preventive control strategies. Ren et al. [15] combined model-driven interpretability with data-driven fitting capabilities to rapidly generate dynamic safety domains corresponding to system instability patterns, ensuring the solubility of TSC-OPF. Meanwhile, in the reference [16], a XGBoost model has been trained for dispatch optimization using time-domain simulation data and implemented transient stability preventive control via a differential evolution algorithm, offering enhanced interpretability. In the literature [17], the CBAM and CNN are integrated to explore key response characteristics related to voltage dynamic process, providing guidance and basis for the formulation of preventive control strategy.

Unlike deep learning and gradient boosting tree algorithms based on static data, Deep Reinforcement Learning (DRL) possesses unique advantages in dynamic decision-making, particularly suited for preventive control problems based on TSC-OPF. Wang and Tang [18] developed a transient stability evaluator using convolutional neural networks and transferred it to a dual-delay deep deterministic gradient algorithm through transfer learning, achieving transient stability preventive control. Furthermore, reference [19] embedded physical knowledge into the policy network, enhancing the model's interpretability. Jiang et al. [20] extended monitoring information from single-time-slice to multi-time-slice, realizing a multi-period online preventive control strategy for power grids.

However, constrained by the computational efficiency of electromechanical transient simulation in power systems, the DRL methods generally suffer from issues such as excessively long training times. To address this, Zeng et al. [21] employed distributed deep deterministic policy gradient to enhance the model's learning efficiency. Reference [22] transformed the distributed resource control problem into a multi-agent distributed cooperative optimization problem for solution, introducing the advantage function decomposition theorem to accelerate agent convergence. However, both approaches impose additional computational resource requirements and neglect the acceleration effect of expert data demonstration on training [23–25]. Thus, this paper proposes a GA-PPO-based prevention control architecture to address the above issues. The contribution of this manuscript is summarized as follows:

(1) The discriminator network parameterizes the gap between expert demonstrations and generated trajectories, combining the advantage function of the environment reward-modified PPO algorithm to enhance the agent's exploration efficiency.
(2) To alleviate GAN's vanishing gradients and mode collapse issues, the objective function of the discriminator is redefined using the Wasserstein distance. The discriminator's network architecture and update mechanism are also improved, ensuring more stable training.
(3) In the context of adversarial training between the discriminator and the generator, the progressive refinement of the advantage function facilitates increasingly accurate advantage estimations within the value network. This, in turn, provides more directed guidance for the exploration process of the generative policy, thereby mitigating the challenges associated with exploratory behavior.

The organization of this manuscript is as follows. Section 2 shows a preventive control model for power systems. Section 3 analyzes the preventive control architecture based on GA-PPO. In Section 4, the performance of the proposed method is compared. Finally, the research conclusions and future research are summarized in Section 5.

2 Preparations

Transient stability of power systems refers to the ability to maintain stability during the transition from a static state to another static state following a disturbance. The TSC-OPF model used in this paper is presented in Section 2.1, while Section 2.2 details the MDP modeling of the power system preventive control model.

2.1 TSC-OPF Dynamic Model

In practical engineering applications, power systems are typically described by a set of differential-algebraic equations, and time-domain simulation methods are employed to analyze their transient stability, as shown below:

$$\begin{cases} \dfrac{dx}{dt} = f(x, y) \\ 0 = g(x, y) \end{cases} \tag{1}$$

where x is the state variable describing the system's dynamic characteristics; y is the algebraic variable describing the system's operational state; t is the time variable.

The objective of prevention and control in this paper is to minimize the total adjustment amount while maintaining the transient stability of the system. Therefore, the comprehensive TSC-OPF model can be represented as a set of complexes, high-dimensional nonlinear differential-algebraic equations. In practical engineering applications, the active power adjustment range of a generator constitutes its thermal reserve capacity. To achieve rapid active power adjustments with minimal total adjustment volume while involving as many generators as possible, this paper sets the objective function to minimize the total change in generator output, as shown below.

$$\begin{aligned} \min \ & \sum_{i \in N_G} \left| P_{G,i} - P_{G,i}^{ori} \right| \\ \text{s.t.} \ & h(x) = 0 \\ & S(x) \le 0 \end{aligned} \tag{2}$$

where, $P_{G,i}^{ori}$ denotes the initial generator output, $P_{G,i}$ denotes the adjusted generator output, and N_G represents the generator node. $h(x)$ denotes equality constraints, and $S(x)$ denotes inequality constraints, as detailed below.

$$P_{Gi} - P_{Li} = V_{Bi} \sum_{j=1}^{N_B} V_{Bj} \left(G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij} \right) \tag{3}$$

$$Q_{Gi} - Q_{Li} = V_{Bi} \sum_{j=1}^{N_B} V_{Bj} \left(G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij} \right) \tag{4}$$

Eqs. (3) and (4) represent power flow balance constraints, indicating that any node in the network must satisfy power flow equilibrium. In these equations, N_B denotes the number of nodes in the system; U_i represent the voltage magnitudes at node i; Q_{Gi} denotes the reactive power output from the generator at node i; P_{Li} and Q_{Li} denote the active and reactive power loads at node i, respectively; θ_{ij} denotes the

phase angle difference between nodes i and j; G_{ij} is the real part of the node admittance matrix; B_{ij} is the imaginary part of the node admittance matrix.

The inequality constraints are as follows:

$$P_{Gi}^{\min} \le P_{Gi} \le P_{Gi}^{\max}, Q_{Gi}^{\min} \le Q_{Gi} \le Q_{Gi}^{\max}, i \in N_G \tag{5}$$

$$V_{Bi}^{\min} \le V_{Bi} \le V_{Bi}^{\max}, P_{ij}^{\min} \le P_{ij} \le P_{ij}^{\max}, i, j \in N_B \tag{6}$$

Eq. (5) represents the generator output constraints, where $P_{Gi}^{\max}$ and $P_{Gi}^{\min}$ denote the upper and lower limits of the active power output allowed for the i-th generator, respectively; $Q_{Gi}^{\max}$ and $Q_{Gi}^{\min}$ denote the upper and lower limits of the reactive power output allowed for the i-th generator, respectively. Eq. (6) represents the node voltage and line power flow constraints, where $V_{Bi}^{\max}$ and $V_{Bi}^{\min}$ denote the upper and lower limits of the voltage magnitude allowed at node i, respectively; $P_{ij}^{\max}$, $P_{ij}^{\min}$ denote the upper and lower limits of the power flow on line from node i to j.

2.2 MDP Modeling

MDP is a sequential decision-making mathematical model used to simulate the stochastic policies of an agent's actions. Transiently stable preventive control is a real-time feedback-based rescheduling decision process, which MDP can describe in detail. Within MDP, state, action, and reward are the primary factors. The corresponding elements for preventive control are described below.

1. State Space

The current state should accurately reflect the operational status of the power system. When the power system operates normally, its topology and line parameters remain constant. Consequently, the magnitude and phase angle of node voltages contain extensive operational information about the system. For instance, the magnitude and phase angle of voltages at both ends of a line can be used to calculate line power flow. Conversely, during short-circuit faults, changes in the power angle between generators and the terminal voltages of generators partially indicate the system's stability state. For instance, excessive power angle differences signify generator desynchronization, leading to system disconnection. Therefore, this paper defines observed variables including node voltage magnitudes and phases, along with generator speed and power angle reflecting generator operational status. The state space is expressed as follows:

$$S = \left\{U_{Bi}, \delta_{Bi}, w_{Gj}, \delta_{Gj}\right\}, i \in N_B, j \in N_G \tag{7}$$

where, δ_{Bi} denotes the phase angle of node i, while ω_{Gj} and δ_{Gj} represent the rotational speed and power angle of generator j, respectively.

2. Action Space

The action space defines all possible control measures an agent can employ. In transient preventive control of power systems, adjusting generator output is one of the most effective control methods. Therefore, to enhance the generality of the results, this paper selects generator output adjustment as the control method. Consequently, the control target for the agent is the active power of generators within its assigned region, continuously adjusted within a specified range (70% to 130%). The maximum active power value for a single adjustment is set to 2% of the difference between the upper and lower limits. The action space is expressed as:

$$A_G = \left\{P_{Gj}\right\}, j \in N_G \tag{8}$$

where P_{Gj} is adjusted active power output for the generator j.

3. Reward

Transient stability prevention and control must ensure predefined fault sets achieve transient power angle stability. Therefore, after the agent executes, the strategy is input into the transient stability simulator. It sequentially traverses all fault conditions within the fault set. Whenever transient power angle stability is satisfied, no penalty is incurred; otherwise, a penalty of −1 is applied. The power angle stability criterion is defined by the following equation.

$$\max_{t\in[T_0,T_f]} |\delta_{i,t} - \delta_{j,t}| < \pi, \forall i, j \in N_G \tag{9}$$

Additionally, the objective function for active power flow adjustment in transient stability prevention and control is given by Eq. (2), and the reward function is expressed as follows:

$$R = \begin{cases} -K, & \text{if power flow diverged} \\ \sum\limits_{s\in F} -1 + \sum\limits_{t\in F} 0 - \sigma \sum\limits_{i\in S_G} |P_{G,i} - P_{G,i}^{ori}|, & \text{if power flow feasible} \end{cases} \tag{10}$$

in the above equation, K represents the penalty for system power flow non-convergence after operation; F denotes the fault set, where s and t represent fault samples in the set that lead to transient stability and transient instability, respectively, after system operation; σ is the weighting factor for active power change, selected as 0.02 in this study.

3 Proposed Framework Based on GA-PPO

DRL is the process by which an agent learns strategies through interaction with its environment to maximize rewards or achieve specific objectives. Through engagement with the environment, the agent continuously explores its surroundings and refines its strategies $\pi(a_t \mid s_t)$. During training, the agent seeks optimal policies π^* to maximize cumulative rewards r, as illustrated below (Fig. 1).

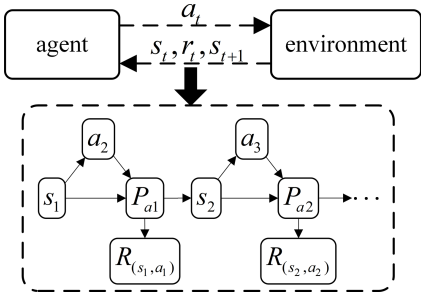

Figure 1: MDP model

To alleviate challenges such as agent exploration difficulties during the initial training phase, this paper improves the PPO algorithm by drawing inspiration from imitation learning. It proposes a preventive control architecture based on adversarial training for PPO. Section 3.1 details the Wasserstein-distance-enhanced GAIL algorithm. Section 3.2 primarily introduces the specific architecture and training framework of the adversarial training-based PPO algorithm.

3.1 Improved Discriminator Based on Wasserstein Distance

GAIL learns how to interact with the environment based on expert demonstration data, ultimately enabling the discriminator to fail to distinguish between the trajectory distributions generated by the agent and those generated by the expert. Its objective function is defined as follows.

$$\min E_{s_t \sim \pi_E} D'\left(\pi_\theta\left(\cdot|s_t\right), \pi_E\left(\cdot|s_t\right)\right) \tag{11}$$

where π is the policy parameter, π_θ denotes the policy network, and π_E represents the expert policy. $D'(\cdot)$ indicates the distance between the generated policy distribution and the expert policy distribution, typically measured by the Jones-Sutton divergence. The objective of imitation learning is to minimize the difference function $D'(\cdot)$.

Therefore, the optimization objectives for the discriminator and generator are as follows:

$$\max_D \left\{E_{\pi_\theta}\left[\log D\left(s,a\right)\right] + E_{\pi_E}\left[\log\left(1 - D\left(s,a\right)\right)\right]\right\} \tag{12}$$

$$\min_G \left\{-E_{\pi_\theta}\left[\log D\left(s,a\right)\right]\right\} \tag{13}$$

in the above equations, $D(\cdot)$ denotes the discriminator, and G denotes the generator. $E_{\pi_\theta}, E_{\pi_E}$ denotes the expected value of the discriminator's judgment results for the agent policy and expert policy, respectively, while $\log(\cdot)$ denotes the logarithm operation. Eq. (12) represents the discriminator's objective function, aiming to maximize the expert policy's judgment proximity to 1 while minimizing the generated policy's proximity to 0. Eq. (13) defines the generator's objective function, which seeks to maximize the discriminator's misclassification rate, thereby approximating the expert's optimal policy.

In GAIL, the reward function for the agent policy network originates from the discriminator's evaluation of the current state-action pair. The reward function for the state-action pair at time step t is defined as follows:

$$r_{GAIL}\left(s_t, a_t\right) = \log D\left(s_t, a_t\right) \tag{14}$$

Although imitation learning lacks an explicit reward function, it still possesses all four elements of a MDP model. Consequently, agent policy updates can be performed by referencing various DRL algorithms. However, traditional GAIL algorithms exhibit poor training stability due to issues such as gradient vanishing and pattern collapse.

Therefore, this paper proposes an improved GAIL training framework based on Wasserstein distance, with the following specific enhancements:

Unlike GAIL, which uses KL divergence to distinguish between two distributions, Wasserstein GAN employs Earth-Mover distance to measure the difference between two distributions, which addresses issues such as gradient vanishing during GAIL training. The specific objective function is defined as follows:

$$\max_D \left\{E_{\pi_\theta}\left[D\left(s,a\right)\right] - E_{\pi_E}\left[D\left(s,a\right)\right]\right\} \tag{15}$$

$$\min_G \left\{-E_{\pi_\theta}\left[D\left(s,a\right)\right]\right\} \tag{16}$$

Furthermore, the discriminator must remove the activation function $Sigmoid(\cdot)$ from the final layer, resulting in the following changes to its loss function:

$$J_D = -\left(E_{\pi_E}\left(D\left(s,a\right)\right) - E_{\pi_\theta}\left(D\left(s,a\right)\right)\right) + \mu^* gp \tag{17}$$

in the above equation, μ denotes the gradient penalty system, and gp represents the gradient penalty, with the specific expression given by:

$$gp = E_{\pi'}\left[\| \nabla_{(s,a)} D(s,a) \|_2 - 1\right] \tag{18}$$

where $\| \cdot \|$ denotes the L_2 norm of the discriminator gradient; ∇ represents the gradient calculation.

Furthermore, since the discriminator's output is not a probability of being classified as the expert policy but rather a score—where higher scores indicate greater similarity to the expert policy—the reward function in Eq. (14) is corrected as follows:

$$r_{GAIL}(s_t, a_t) = D(s_t, a_t) \tag{19}$$

The specific structure is shown in Fig. 2.

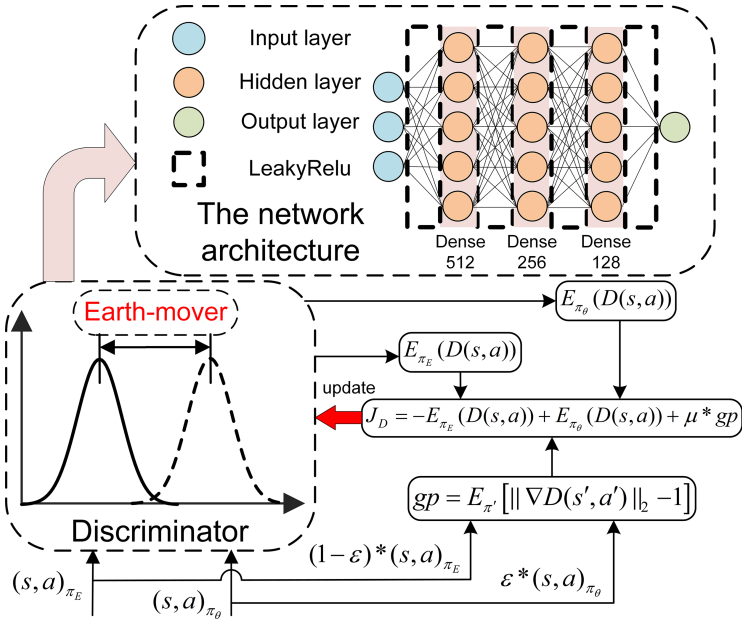

Figure 2: The improved discriminator

3.2 The Proposed GA-PPO Algorithm

The generator in this paper is the PPO-Clip algorithm. All subsequent references to PPO in this paper refer to the PPO-Clip algorithm. Its objective function is defined as follows:

$$J^{CLIP}(\pi) = E\left[\min\left(r(\pi) A, \text{clip}\left(r(\pi), 1-\epsilon, 1+\epsilon\right) A\right)\right] \tag{20}$$

in the above equation, $r(\pi)$ denotes the probability ratio between the new and old policies, ε is the pruning hyperparameter, A is the advantage function, and represents the difference between the reward and the state value.

With GAIL significantly enhances the learning efficiency of intelligent agents through expert demonstration, in the preventive control algorithm proposed in this paper, the objective function is as follows:

$$J(\pi) = (1-\alpha) E\left[\min\left(r(\pi) A, \text{clip}\left(r(\pi), 1-\epsilon, 1+\epsilon\right) A\right)\right] + \alpha \min E_{s_t \sim \pi_E} D_{KL}\left(\pi_\theta(\cdot|s_t), \pi_E(\cdot|s_t)\right) \tag{21}$$

where α is the parameter for balancing strategy improvement and cloning. During the early training phase, the agent aims to enhance learning efficiency through expert demonstration, gradually increasing its exploration capability thereafter. Consequently, the parameter α is gradually decayed as training progresses, with the specific expression given by:

$$\alpha_{t+1} = \begin{cases} \alpha_t/1.01, & R > R_{mean} \\ \alpha_t, & \text{otherwise} \end{cases} \tag{22}$$

among these, R denotes the reward for the current round, while R_{mean} represents the average reward over the preceding 10 rounds.

The PPO algorithm evaluates the quality of action distributions using an advantage function, defined as follows:

$$A_t = Q_{\pi_{\text{odd}}}(s_t, a_t) - V_{\pi_{\text{odd}}}(s_t) \tag{23}$$

where $V_{\pi_{old}}(\cdot)$ is the state value function, representing the average value of state s_t; $Q_{\pi_{old}}(\cdot)$ is the action value function, indicating the value of taking action a_t in state s_t, defined as follows:

$$Q_{\pi_{old}}(s_t, a_t) = r_t + \gamma V_{\pi_{old}}(s_{t+1}) \tag{24}$$

Unlike the traditional PPO algorithm, in the GA-PPO algorithm proposed in this paper, the action value function of the PPO algorithm serving as the generator network not only incorporates the manually defined reward function but also includes the simulated state reward generated by the discriminator network, as expressed in Eq. (19), specifically as follows:

$$\hat{Q}_{\pi_{old}}(s_t, a_t) = (1-\alpha) r_t + \alpha r_{GAIL} + \gamma V_{\pi_{old}}(s_{t+1}) \tag{25}$$

where r_t represents the immediate reward, and γ is the discount factor. In summary, the improved generalized advantage estimate is given by:

$$\hat{A}_t = \sum_{l=0}^{T-t} (\gamma\lambda)^l \left[(1-\alpha) r_t + \alpha r_{GAIL} + \gamma V(s_{t+1}) - V(s_t)\right] \tag{26}$$

in the equation above, λ is a parameter in the advantage calculation. Therefore, the loss function of the improved PPO algorithm is as follows:

$$L^{CLIP}(\pi) = E\left[\min\left(r(\pi)\hat{A}, \text{clip}(r(\pi), 1-\epsilon, 1+\epsilon)\hat{A}\right)\right] \tag{27}$$

The specific model architecture is shown in the Fig. 3 below.

3.3 Algorithm Training Process

In the application of existing DRL algorithms to preventive control problems, computational efficiency is low due to the extensive interaction required between the training process and the power system electromechanical transient simulator. The GA-PPO algorithm proposed in this paper enables the agent to interact with the environment under expert data guidance, sampling to obtain authentic state-action trajectories. Its specific training process is illustrated in the Algorithm 1 below.

1. The agent outputs actions a_t through the action network based on the input state s_t at the current time step. It then interacts with the power system transient simulator to obtain the reward r_t for the

current time step and the state s_{t+1} for the next time step. This process repeats cyclically until the cycle termination flag $done$ is set to true.

2. During the update phase, the discriminator is first updated using the loss between expert experience and actual action trajectories. Subsequently, the corresponding simulated state reward r_{GAIL} derived from the discriminator is embedded into the action value function. Then the modified advantage function $\hat{A}_t$ is computed to update the generator, thereby updating the PPO agent.
3. Finally, the agent interacts with the environment, generating new real-world action trajectories to complete a new round of updates. This iterative process continues until both the generator network and discriminator network converge to an equilibrium state.

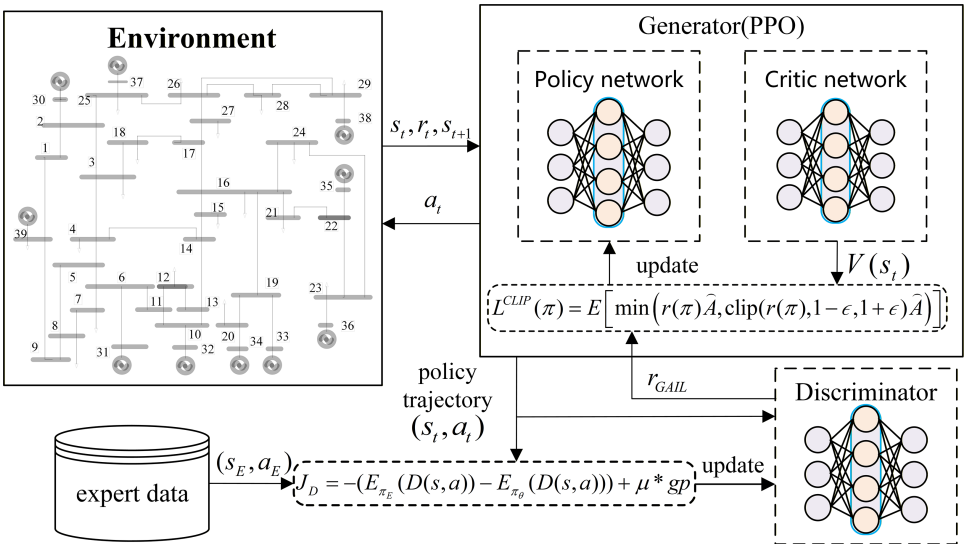

Figure 3: The GA-PPO architecture

Algorithm 1: GA-PPO

Input: Expert trajectories π_E, states
Output: Learned policy π_θ

Initialize policy π_θ (PPO parameters)
Initialize discriminator D_ω
for *iteration* $k = 1, 2, ..., K$ **do**
    Initialize policy buffer $\mathcal{B} \leftarrow \varnothing$
    //1. Sample trajectories
    while not terminal state **do**
        Sample trajectory $\tau \sim \pi_\theta$ and store in buffer $\mathcal{B}$
    end
    //2. Update discriminator
    Sample policy batch $\{(s_i, a_i)\} \sim \mathcal{B}$
    Sample expert batch $\{(s_j^E, a_j^E)\} \sim \pi_E$
    Update D_ω to minimize discriminator loss:
    $\mathcal{L}_D(\omega) = \mathbb{E}_{\pi_\theta}[D_\omega(s,a)] - \mathbb{E}_{\pi_E}[D_\omega(s,a)] + \mu \cdot \text{GP}$
    //3. Update PPO policy

(Continued)

Algorithm 1 (continued)

$r(s,a) = D_\omega(s,a) \cdot \alpha + r_t \cdot (1-\alpha)$
Compute advantages $\hat{A}_t$ using GAE
for *epoch* = 1, 2, ..., 10 **do**
Update π_θ by $\mathcal{L}_{\mathrm{PPO}}(\pi) =$
$\mathbb{E}_t\left[\min\left(r_t(\theta)\hat{A}_t, \mathrm{clip}\left(r_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_t\right)\right]$
end
buffer $\mathcal{B} \leftarrow \varnothing$
end
Return policy π_θ

4 Case Analysis

This paper employs an IEEE 39-node system [26] and an improved IEEE 68-node system to validate the effectiveness of the proposed GA-PPO algorithm. The simulation platform consists of a personal computer configured with a 2nd Gen Intel® Core (TM) i7-12700 CPU operating at 2.10 GHz and 32 GB of memory. The primary code is written in Python, with machine learning algorithms implemented using the PyTorch framework. Steady-state and transient simulations are performed using the PSAT toolbox in MATLAB. The power flow and time-domain simulations were solved using a fixed integration step size by default, and the numerical convergence tolerance was set to 1×10^{-5}.

4.1 Result in IEEE 39-Node System and Improved IEEE 68-Node System

The IEEE 39-node system comprises 39 nodes, 21 loads, 10 generators, 11 transformer branches, and 35 transmission lines.

During training, the load is set to vary between 90% and 110%. The agent outputs the active power of controllable generators based on input states. Power flow calculations and time-domain simulations traverse predefined faults in the fault set, capturing the power system's state after each predefined fault occurs. Eq. (9) is then used to determine whether the system is transiently stable. The update process employs stochastic gradient descent. The entire training cycle consists of 1000 iterations, with each iteration grouped into sets of 50 steps to monitor reward fluctuations. Training terminates if the reward change remains below 1×10^{-5} for 50 consecutive iterations. This study examines the transient stability of the system when three-phase short-circuit faults occur at Bus 8, Bus 28, and Bus 29 and persist for 0.1 s, with a simulation time set to 6 s. And the expert demonstrations utilized in this work were generated by the PPO agent that had been trained to convergence. The reward trajectory of the PPO agent stabilized after approximately 400 episodes, indicating it had reached a mature performance plateau (its learning curve is shown in Section 4.2). Data collected from a subsequent 800 episodes were used to form the expert dataset, ensuring the policy's behavior was stable and representative of a proficient, near-optimal control strategy. It is noteworthy that the expert policy is not required to be perfectly optimal. The core objective of leveraging this expert data is to provide a high-quality initialization for the GA-PPO agent, thereby guiding its initial exploration and significantly accelerating the training process. The subsequent adversarial training phase enables the agent to refine and potentially surpass the expert's performance. The hyperparameter settings for the GA-PPO algorithm are shown in Table 1.

The loss curve and reward function curve during training of the proposed GA-PPO algorithm are shown in Fig. 4.

The agent's reward function gradually increases during the first 50+ exploration rounds, indicating that the network is learning to avoid actions that cause the power angle to become unstable. Around 100 rounds,

the reward function gradually approaches −20, while the loss also converges toward 0. Subsequently, the agent continues to explore more optimal strategies based on this foundation.

Table 1: Model parameters

Parameter	Value
Policy network learning rate	5×10^{-5}
Value network learning rate	5×10^{-5}
Discriminator learning rate	2×10^{-4}
γ	0.98
λ	0.95
Hidden layer	[256, 256, 256]
Batch sizes	50

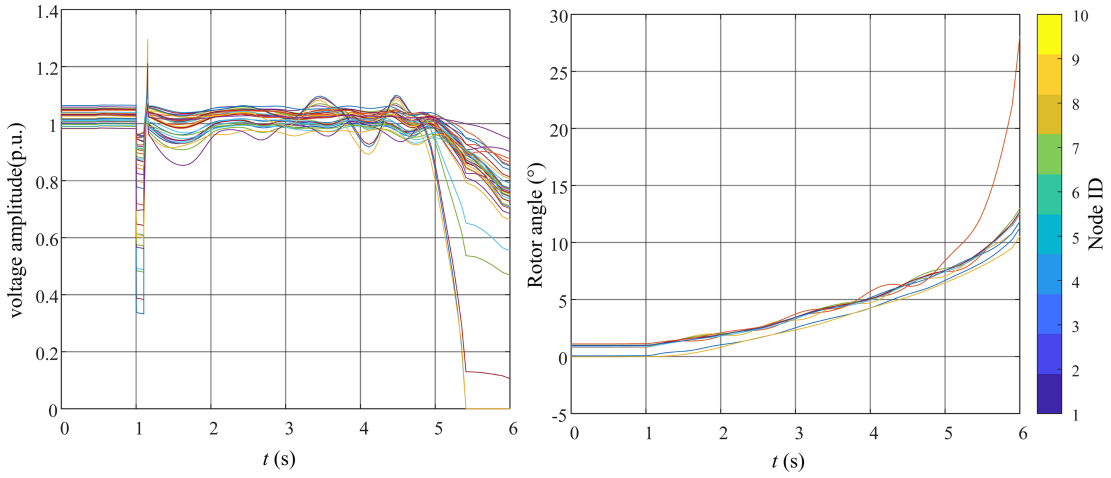

Figure 4: The simulated fault on Bus 8 during system operation without preventive control measures

Based on the training results of the GA-PPO algorithm, the agent's preventive control strategy for the predefined fault set is shown in Table 2, with adjustments made for each controllable generator node. Nodes 38 and 39 exhibit poor transient stability and are prone to triggering transient instability during faults, resulting in the largest power output adjustments. Conversely, Node 32 demonstrates better stability, leading to the smallest power output adjustment.

Table 2: Prevention and control strategy based on GA-PPO

Generator node ID	Output before adjustment	Adjusted output	Adjustment amount
30 (p.u.)	2.5	2.4435	0.0565
32 (p.u.)	6.5	6.4954	0.0046
33 (p.u.)	6.32	6.2655	0.0545
34 (p.u.)	5.08	5.2293	−0.1493
35 (p.u.)	6.5	6.5329	−0.0329
36 (p.u.)	5.6	5.6459	−0.0459
37 (p.u.)	5.4	5.4137	−0.0137
38 (p.u.)	8.8	8.1405	0.6595
39 (p.u.)	10	9.686	0.314

Further observation of the power angle and voltage curves in the preventive control scenario following a three-phase short circuit on bus 8 is shown in Figs. 5 and 6.

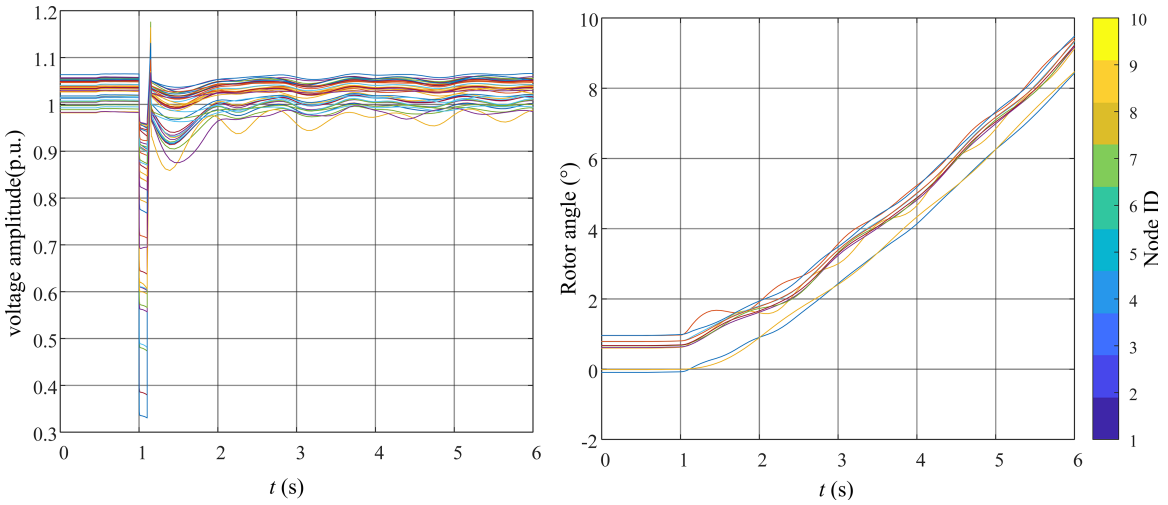

Figure 5: The simulated fault on Bus 8 during system operation with GA-PPO output strategy

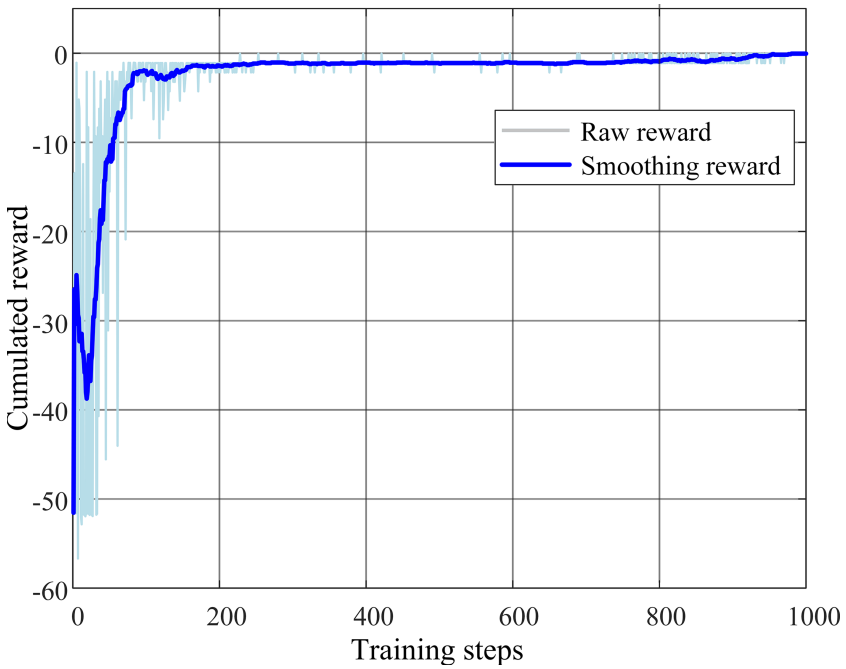

Figure 6: The reward curve for the improved IEEE 68-node system

Fig. 5 demonstrates that without preventive control measures, the simulated fault on Bus 8 during system operation would cause the generator's power angle to lose synchronization. This results in severe node voltage oscillations, with excessively high peak voltages at the instant of fault clearance, rendering the system inoperable. However, after implementing output adjustments using the GA-PPO output strategy, the generator phase angle deviation curve and node voltage curve during a severe fault are shown in Fig. 6. It is evident that the generator phase angle deviation does not exhibit a continuously widening trend. Although

voltage oscillations occur to some extent after the fault, they gradually stabilize as the system operates. No generator desynchronization or sustained severe node voltage oscillations occur, and the peak node voltage at the instant of fault clearance remains below $V_{Bi}^{\max}$.

This paper further verifies the effectiveness of the proposed method on the modified IEEE 68-node system, which comprises 68 buses, 16 generators, and 33 loads. A wind farm consisting of 114 doubly fed induction generators (DFIGs) with 570 mw is integrated at Bus 5. The wind speed model for the wind farm is simulated using a Weibull distribution. All synchronous generators in the system are represented by sixth-order models, while the DFIGs are modeled using third-order representations. To simulate transient instability, a typical fault scenario is implemented as follows: a three-phase short-circuit fault is applied at Bus 15, Bus 29, and Bus 47 at t = 1 s, with a fault duration of 0.1 s. Load levels are randomly selected between 90% and 110% to enhance sample diversity. The agent is trained based on the GA-PPO algorithm, and the resulting reward curve is shown in the figure below.

As observed, the training curve of the GA-PPO algorithm exhibits a highly stable convergence trend. The reward rises to around −1 after approximately 200 episodes and eventually converges near −0.05 after about 900 episodes.

Upon completion of training, a fault scenario is selected from the training set to validate the effectiveness of the control strategy learned by the agent: a three-phase short-circuit fault occurs at Bus 47 with a duration of 0.1 s and a load level of 99%. Without control intervention, the system voltage response in this scenario is shown in Fig. 7a. The system fails to maintain transient stability after being subjected to this significant disturbance. When the trained agent is applied to implement preventive control in the same scenario, the resulting system dynamic response is shown in Fig. 7b.

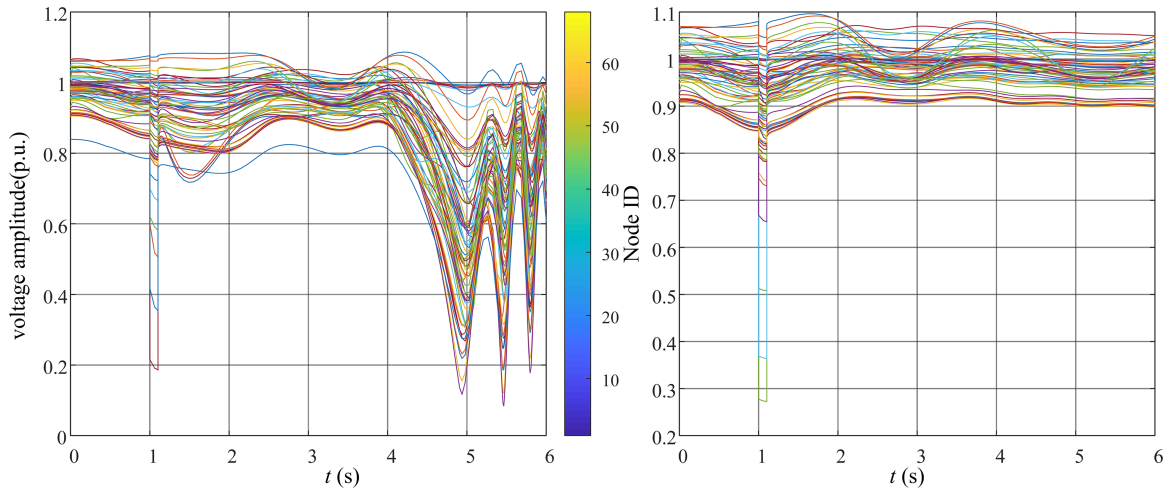

Figure 7: The system voltage response curve. (**a**) Without preventive control. (**b**) With GA-PPO output strategy

The corresponding control strategy of the agent is presented in Table A1 in the Appendix A. The results demonstrate that after the agent implements preventive control actions, the system voltage can be promptly restored to within the permissible range and remain stable, indicating that both rotor angle stability and voltage stability of the system are maintained.

4.2 Comparison of Different Methods

To further validate the effectiveness of the proposed method, we selected two commonly used continuous action space DRL methods from existing studies [22]—PPO and SAC—as comparison algorithms. The final results are shown in Fig. 8.

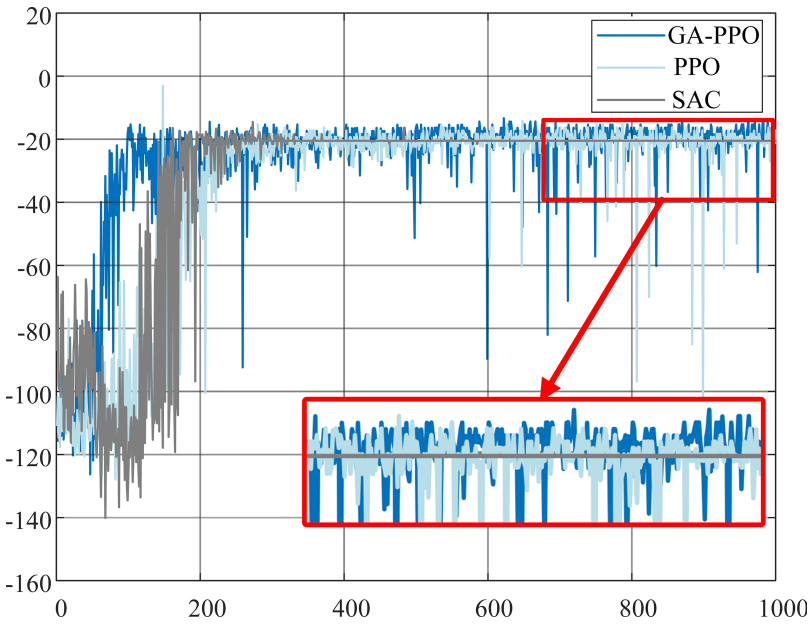

Figure 8: The result of different algorithms

The first 50 training rounds of the SAC algorithm constitute the experience accumulation phase. After 50 rounds, the reward curve first declines and then rapidly rises, indicating that the agent learns effective information from exploration. It eventually converges around −20, exhibiting overfitting. In contrast, the reward curve of the PPO algorithm continues to rise throughout the entire training process. As shown in Fig. 8, during the early training phase, our method learns from expert demonstrations to explore higher-reward regions more rapidly. The enlarged region further demonstrates that GA-PPO achieves superior reward convergence compared to both SAC and PPO algorithms.

The convergence performance of different algorithms is presented in Table 3. It can be observed that GA-PPO requires the shortest computational time during the training phase, amounting to only 19.62 h. This represents a 16.79% improvement over the PPO algorithm and a 30.96% improvement over SAC. Furthermore, in terms of the convergent reward, GA-PPO achieves a value of −18.3, which is notably higher than those of PPO (−19.1) and SAC (−22.7), representing an improvement of approximately 4.19% over PPO and 19.38% over SAC. Ignoring the electromechanical transient simulation time, all three agents take only about 0.002 s from receiving the state to providing preventive control decisions, with negligible time differences. However, there are slight differences in the objective function, as shown in Eq. (2). The GA-PPO has the lowest total adjustment (1.3619 p.u.).

Table 3: Comparison of convergence performance

Method	Training time	Training improvement	Convergent reward	Reward improvement	Total adjustment
SAC	28.42 h	(baseline)	−22.7	(baseline)	2.0521 (p.u.)
PPO	23.58 h	20.53%	−19.5	16.41%	1.5489 (p.u.)
GA-PPO	19.62 h	30.96%	−18.3	19.38%	1.3619 (p.u.)

4.3 Sensitivity Analysis of Parameter α

To further investigate the effectiveness of the proposed method, an analysis is conducted on the parameter α in Eq. (22). Training is performed with α set to 0.3, 0.6, and 0.9, respectively. The performance of the GA-PPO reward curves for each parameter setting is as follows (Fig. 9):

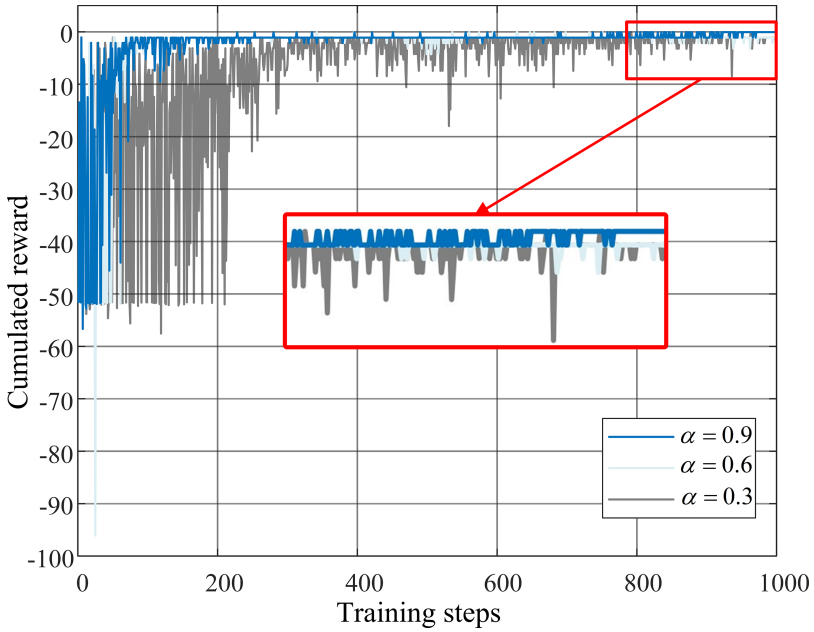

Figure 9: The performance of GA-PPO reward curves for each parameter

It can be observed that the configuration with α = 0.9 demonstrates the best performance, requiring only 12.3 h to converge and achieving the highest final reward (−0.05). While the configuration with α = 0.6 eventually attains the reward (−1.06), it necessitates a longer training duration of 13.5 h. In contrast, the α = 0.3 configuration requires 14.8 h of training and only achieves a same reward of −1.06, failing to reach the optimal performance level. This comparative analysis confirms the superior performance of the α = 0.9 parameter setting.

4.4 Generalization Performance Analysis

To analyze the generalization capability of the trained agent, a testing scenario involving a new fault not encountered during training is conducted: a three-phase short-circuit fault occurs at Bus 28 with a duration of 0.1 s and a load level of 99%. Fig. 10 presents the system voltage response curves without control and

with control actions applied by the agent in this scenario. The corresponding control strategy of the agent is provided in Table A2 in the Appendix A. The results indicate that the agent can still produce effective control decisions when confronted with unseen testing scenarios, thereby demonstrating its satisfactory generalization performance.

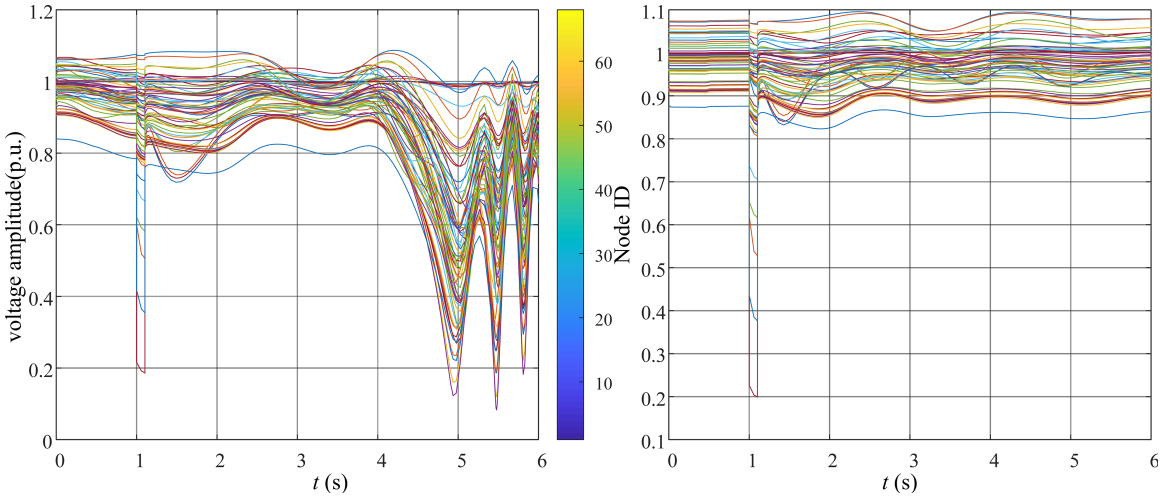

Figure 10: The system voltage response curve. (**a**) Without preventive control. (**b**) With GA-PPO output strategy

5 Conclusion

The computational complexity of transient stability preventive control methods for conventional power systems increases exponentially with grid scale expansion. Meanwhile, the DRL algorithm imposes significant training burdens due to extensive interactions with transient simulators during training. Recent approaches such as distributed architectures and multi-agent systems have partially improved learning efficiency but introduce new demands on experimental equipment. Therefore, this paper proposes a GA-PPO-based preventive control architecture with the following advantages:

1. The discriminator network parameterizes the gap between expert demonstrations and generated trajectories, integrating environmental rewards to refine the advantage function of the PPO algorithm, thereby enhancing the agent's exploration efficiency.
2. To address GAN's vanishing gradients and mode collapse issues, the objective function is restructured using Wasserstein distance. Improvements to the discriminator's network architecture and update mechanism ensure more stable training.
3. During adversarial training between discriminator and generator, continuous refinement of the advantage function enables the value network to obtain more precise advantage estimates. This further guides the exploration direction of the generative policy, reducing exploration difficulty.

Further research will be conducted on the impact of different hyperparameters on the convergence speed and effectiveness of reinforcement learning for intelligent agents; future investigations will also include a systematic analysis of the reward function's structure and its hyperparameters, so as to further enhance the performance and interpretability of the proposed control framework.

Acknowledgement: This paper was completed with the hard help of every author.

Funding Statement: This research was funded by Key technologies for stability analysis and coordinated control of a new power system based on data-mechanism integration of Technology Project of China Southern Power Grid Company Limited (ZBKJXM20232027).

Author Contributions: The authors confirm contribution to the paper as follows: Yun Yu proposed the research concept, designed the methodology, conducted simulations, managed the project. Li Lin and Ximing Zhang contributed to validation, reviewing and editing the manuscript. Yang Yu provided critical resources and assisted with formal analysis. Wei Zhang contributed to investigation and data collection. Kai Cheng participated in writing review. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data of IEEE 39 power system that support the findings of this study are openly available in [psat_python] at [https://github.com/Thueea-bcb/psat_python] (accessed on 16 November 2025).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

Abbreviations

GA-PPO	Generative adversarial proximal policy optimization
MDP	Markov decision process
PPO	Proximal policy optimization
GAN	Generative adversarial network
SAC	Soft actor-critic
TSC-OPF	Transient stability constrained-optimal power flow
TSI	Transient stability index
DRL	Deep reinforcement learning
$P_{G,i}^{ori}$, $P_{G,i}$	The initial and adjusted generator output
$h(x)$, $S(x)$	The equality constraints and inequality constraints
$P_{Gi}^{\max}$, $P_{Gi}^{\min}$	The upper and lower limits of the active power output
$Q_{Gi}^{\max}$, $Q_{Gi}^{\min}$	The upper and lower limits of the reactive power output
$V_{Bi}^{\max}$, $V_{Bi}^{\min}$	The upper and lower limits of the voltage magnitude
$P_{ij}^{\max}$, $P_{ij}^{\min}$	The upper and lower limits of the power flow on line from node i to j
δ_{Bi}	The phase angle of node i
K	The penalty for system power flow non-convergence after operation
π_θ,π_E	The agent policy and expert policy
α	The parameter for balancing strategy improvement and cloning

Appendix A

Table A1: Prevention and control strategy for IEEE 68-node system based on GA-PPO

Generator node ID	Output before adjustment	Adjusted output	Adjustment amount
1 (p.u.)	2.5	2.55	−0.05
2 (p.u.)	5.45	5.341	0.109
3 (p.u.)	6.5	6.3733	0.1267
4 (p.u.)	6.32	6.2257	0.0943
5 (p.u.)	5.052	4.9567	0.0953
6 (p.u.)	7	6.86	0.14

(Continued)

Table A1 (continued)

Generator node ID	Output before adjustment	Adjusted output	Adjustment amount
7 (p.u.)	5.6	5.6028	−0.0028
8 (p.u.)	5.4	5.2936	0.1064
9 (p.u.)	8	7.8449	0.1551
10 (p.u.)	5	5.0125	−0.0125
11 (p.u.)	10	10.1997	−0.1997
12 (p.u.)	13.5	13.77	−0.27
13 (p.u.)	35.91	36.628	−0.718
14 (p.u.)	17.85	17.8583	−0.0083
15 (p.u.)	10	10.1989	−0.1989
16 (p.u.)	2.5	2.55	−0.05

Table A2: Generalization performance analysis for IEEE 68-node system

Generator node ID	Output before adjustment	Adjusted output	Adjustment amount
1 (p.u.)	2.5	2.6236	−0.1236
2 (p.u.)	5.45	5.3403	0.1097
3 (p.u.)	6.5	6.11	0.39
4 (p.u.)	6.32	5.9408	0.3792
5 (p.u.)	5.052	4.9742	0.0778
6 (p.u.)	7	6.58	0.42
7 (p.u.)	5.6	5.4928	0.1072
8 (p.u.)	5.4	5.0763	0.3237
9 (p.u.)	8	8.1031	−0.1031
10 (p.u.)	5	5.1616	−0.1616
11 (p.u.)	10	10.6	−0.6
12 (p.u.)	13.5	13.2313	0.2687
13 (p.u.)	35.91	38.0646	−2.1546
14 (p.u.)	17.85	18.47	−0.62
15 (p.u.)	10	10.1719	−0.1719
16 (p.u.)	2.5	2.6236	−0.1236

References

1. Yan J, Li C, Liu Y, Yu D, Jia Z. Incremental model evolution for power system security early warning based on knowledge distillation and active learning. IEEE Trans Ind Inform. 2024;20(11):12958–68. doi:10.1109/TII.2024.3431034.
2. Bao T, Ma X, Li Z, Yang D, Wang P, Zhou C. Novel static security and stability control of power systems based on artificial emotional lazy Q-learning. Energy Eng. 2024;121(6):1713–37. doi:10.32604/ee.2023.046150.
3. Mejia-Ruiz GE, Paternina MRA, Segundo Sevilla FR, Korba P. Fast hierarchical coordinated controller for distributed battery energy storage systems to mitigate voltage and frequency deviations. Appl Energy. 2022;323(5):119622. doi:10.1016/j.apenergy.2022.119622.
4. Mejia-Ruiz GE, Arrieta Paternina MR, Ramirez-Gonzalez M, Sevilla FRS, Korba P. Real-time co-simulation of transmission and distribution networks integrated with distributed energy resources for frequency and voltage support. Appl Energy. 2023;347(7):121046. doi:10.1016/j.apenergy.2023.121046.
5. Hatziargyriou N, Milanovic J, Rahmann C, Ajjarapu V, Canizares C, Erlich I, et al. Definition and classification of power system stability-revisited & extended. IEEE Trans Power Syst. 2021;36(4):3271–81. doi:10.1109/tpwrs.2020.3041774.
6. Guan S, Zhang R, Xu R. Power system preventive transient stability control: a comprehensive review. In: Proceedings of the 2023 IEEE International Conference on Energy Technologies for Future Grids (ETFG); 2023 Dec 3–6; Wollongong, Australia. p. 1–6. doi:10.1109/ETFG55873.2023.10408487.

7. Zhu S, Ma H, Chen L, Wang B, Wang H, Li X. et al. Short-term load forecasting of an integrated energy system based on STL-CPLE with multitask learning. Prot Control Mod Power Syst. 2024;9(6):71–92. doi:10.23919/PCMP.2023.000101.
8. Xia T, Zhang N, Li W, Du E, Su Y, Fang C, et al. Efficient embedding of neural network-based stability constraints into power system dispatch. IEEE Trans Power Syst. 2024;39(3):5443–6. doi:10.1109/tpwrs.2024.3363948.
9. Zhang Y, Yi Y, Deng W, Liu S, Zhou L, Lin K, et al. Consumer-branch connectivity identification of low voltage distribution networks based on data-driven approach. Prot Control Mod Power Syst. 2024;9(4):69–82. doi:10.23919/PCMP.2023.000465.
10. Ma H, Ma L, Wang Z, Li Z, Zhu Y, Liu Y. Multi-lever early warning for wind and photovoltaic power ramp events based on neural network and fuzzy logic. Energy Eng. 2024;121(11):3133–60. doi:10.32604/ee.2024.055051.
11. Xia S, Zhang C, Li Y, Li G, Ma L, Zhou N, et al. GCN-LSTM based transient angle stability assessment method for future power systems considering spatial-temporal disturbance response characteristics. Prot Control Mod Power Syst. 2024;9(6):108–21. doi:10.23919/pcmp.2023.000116.
12. Su T, Zhao J, Chen X. Deep sigma point processes-assisted chance-constrained power system transient stability preventive control. IEEE Trans Power Syst. 2024;39(1):1965–78. doi:10.1109/tpwrs.2023.3270800.
13. Chen Y, Zhu J, Liu Y, Zhang L, Zhou J. Distributed hierarchical deep reinforcement learning for large-scale grid emergency control. IEEE Trans Power Syst. 2024;39(2):4446–58. doi:10.1109/tpwrs.2023.3298486.
14. Liu Y, Su T, Qiu G, Gao H, Liu J, Shui Y. Analytic deep learning and stepwise integrated gradients-based power system transient stability preventive control. IEEE Trans Power Syst. 2024;39(1):863–76. doi:10.1109/tpwrs.2023.3248293.
15. Ren J, Zeng Y, Qin C. Hybrid model-driven and data-driven surrogate-assisted method for transient stability preventive control. IEEE Trans Ind Applicat. 2025;61(2):2375–85. doi:10.1109/tia.2024.3462679.
16. Zhang S, Zhang D, Qiao J, Wang X, Zhang Z. Preventive control for power system transient security based on XGBoost and DCOPF with consideration of model interpretability. CSEE J Power Energy Syst. 2021;7(2):279–94. doi:10.17775/CSEEJPES.2020.04780.
17. Zhang Z, Qin B, Ding T, Gao X, Zhang Y. CBAM-CNN based transient overvoltage preventive control considering piecewise linear control sensitivity. IEEE Trans Power Syst. 2025;40(5):3645–56. doi:10.1109/tpwrs.2025.3546676.
18. Wang T, Tang Y. Transient stability preventive control based on graph convolution neural network and transfer deep reinforcement learning. CSEE J Power Energy Syst. 2025;11(1):136–49. doi:10.17775/CSEEJPES.2022.05030.
19. Liu Y, Gao S, Qiu G, Liu T, Ding L, Liu J. A physics-informed action network for transient stability preventive control. IEEE Trans Power Syst. 2023;38(2):1771–4. doi:10.1109/tpwrs.2022.3233763.
20. Jiang T, Qie Z, Li W, Li Z, Wu X. Multi-period online preventive control technology for high-proportion renewable energy power grid using reinforcement learning. Autom Electr Power Syst. 2025:1–12. (In Chinese). doi:10.7500/AEPS20241222002.
21. Zeng H, Zhou Y, Guo Q, Cai Z, Sun H. Distributed deep reinforcement learning-based approach for fast preventive control considering transient stability constraints. CSEE J Power Energy Syst. 2023;9(1):197–208. doi:10.17775/CSEEJPES.2020.04610.
22. Niu ZW, Ji Y, Li BY, Dang ZF, Wu YX, Han XQ. Preventive control method of power system transient stability based on multi-agent deep reinforcement learning with advantage decomposition. Power Syst Technol. 2025;49(6):2311–21. (In Chinese). doi:10.13335/j.1000-3673.pst.2024.0389.
23. Huang Y, Zhao X. Wind farm control via offline reinforcement learning with adversarial training. IEEE Trans Automat Sci Eng. 2025;22(18):12845–56. doi:10.1109/tase.2025.3548565.
24. Zhu Z, Chan KW, Bu S, Hu Z, Xia S. An imitation learning based algorithm enabling priori knowledge transfer in modern electricity markets for Bayesian Nash equilibrium estimation. IEEE Trans Power Syst. 2024;39(4):5465–78. doi:10.1109/tpwrs.2023.3341456.
25. Qian T, Liang Z, Shao C, Zhang H, Hu Q, Wu Z. Offline DRL for price-based demand response: learning from suboptimal data and beyond. IEEE Trans Smart Grid. 2024;15(5):4618–35. doi:10.1109/tsg.2024.3382293.
26. Thueea-bcb. Psat_Python [Dataset]. San Francisco, CA, USA: GitHub Inc.; 2023 [cited 2025 May 10]. Available from: https://github.com/Thueea-bcb/psat_python.