

Lightweight actor-critic generative adversarial networks for real-time smart generation control of microgrids

Kunlun Han ^a, Kai Yang ^{a,b,c}, Linfei Yin ^{a,b,c,*}

^a College of Electrical Engineering, Guangxi University, Nanning, Guangxi 530004, China

^b Institute of Artificial Intelligence, Guangxi University, Nanning, Guangxi 530004, China

^c Guangxi Key Laboratory of Intelligent Control and Maintenance of Power Equipment, Guangxi University, Nanning 530004, China

HIGHLIGHTS

- Real-time smart generation control of microgrids is solved.
- Lightweight actor-critic generative adversarial networks are proposed.
- The training speed and stability of generative adversarial networks are improved.
- Multi-path lightweight method is proposed to reduce computation time of networks.
- Better dynamic performance and lower economic costs are obtained simultaneously.

ARTICLE INFO

Keywords:

Multi-path lightweight method
Ensemble empirical mode decomposition
Evolutionary strategy
Generative adversarial networks
Smart generation control

ABSTRACT

Large-scale introduction of new energy could effectively alleviate energy shortage and environmental pollution. However, the uncertainty of wind and solar energy brings serious random disturbance problems to microgrids. This paper proposes lightweight actor-critic generative adversarial networks based on ensemble empirical mode decomposition and evolutionary strategy for increasing the robustness and adaptability of microgrids. Firstly, to improve the training speed and stability of generative adversarial networks, the complex power data is properly decomposed into more regular and simpler sub-data by the ensemble empirical mode decomposition; the generative adversarial networks are optimized by the evolutionary strategy with a set of different loss functions. Secondly, fully connecting the generative adversarial networks with the actor-critic framework, the lightweight actor-critic generative adversarial networks can realize dynamic learning in the random environment and store the sample for online training by the empirical replay mechanism. Thirdly, the multi-path lightweight method is proposed to reduce the consumption of time and storage resources of lightweight actor-critic generative adversarial networks. Eventually, the lightweight actor-critic generative adversarial networks are compared with comparison algorithms in two-area and real-life four-area systems. Case study results reveal that lightweight actor-critic generative adversarial networks have better dynamic performance, online learning capabilities, and higher control performance with lower economic costs.

1. Introduction

Since global energy shortage and environmental pollution are getting increasingly serious, new energy generation with efficiency and clean has been widely connected to microgrids [1]. However, new energy sources such as wind and solar have the characteristics of randomness and intermittency, which intensifies the pressure on frequency regulation and power balance of microgrids [2]. Meanwhile, the

operation of the current microgrid presents stronger decentralization, diversity, and unpredictability [3]. Therefore, traditional microgrids urgently need to transform into smart microgrids that accommodate multiple new energy [4]. Therefore, intelligent control strategies should be designed to improve the self-adaptation and stability of microgrids.

Automatic generation control (AGC) [5] is an important technical tool to regulate systemic frequency and active power. According to the different implementation methods, the current AGC strategies can be divided into conventional AGC and smart generation control (SGC) [6].

* Corresponding author at: College of Electrical Engineering, Guangxi University, Nanning, Guangxi 530004, China.

E-mail address: yinlinfei@163.com (L. Yin).

Nomenclature	
Variables	
a_t	History action of microgrids
α_a	Learning rate of actor
α_c	Learning rate of critic
b	Baseline of loss function
B	Frequency bias coefficient of primary frequency regulation
β	Weight threshold of actor and critic
c	Weight coefficients of e_{ACE}
C_{total}	Total generation cost
δ	Adjustable coefficient of Huber Loss
D	Number of parameters of generator
Δf	Frequency deviation
ΔP_w	The w -th unit output
$\Delta P_w^{\min}, \Delta P_w^{\max}$	The w -th unit output limitation
ΔP_{ex}	Tie-line exchange power deviation
η	Balance coefficient of fitness functions
e_{ACE}	ACE value
e_{ISE}	ISE value
e_{IAE}	IAE value
e_{ITAE}	ITAE value
$E(\bullet)$	Calculation expected value
γ	Discount factor
G	Number of sub-paths
H	Decomposition times of EEMD
$\ell(t)$	Gaussian white noise
$IMF_f(t)$	Final IMFs
$G_N(x^{(n)})$	Predicted value
J	Number of IMFs
κ	Scale coefficients of C_{total}
λ	Regular coefficient of the nonconvex group MCP regularization
L	Number of hidden layers of each sub-path
$P_{w,t}$	The w -th unit output at the t -th time
P_w^{up}	The w -th unit maximum output between two control periods
P_w^{down}	The w -th unit minimum output between two control periods
R	Reward function of LAC-GANs
$\Re(\bullet)$	Overall evaluation function
σ	TD error
s_t	History state of microgrids
m_k	First order moment estimation
m_k'	Deviation correction of m_k
μ	Adjustment coefficient of the nonconvex group MCP regularization
v_a, v_b, v_c	Economic coefficients of the w -th generator unit
N	Batch size of training
$\varepsilon(t)$	Final residue
$\vartheta(t)$	Residue in EEMD
v_k	Second order moment estimation
v_k'	Deviation correction of v_k
W	Number of generator units
x	Sample from the real data distribution ρ_{data}
$x(t)$	Power data time series
$x'(t)$	Noise-added signal
$y^{(n)}$	Actual value
z	Noise from the prior distribution ρ_{noise}
ζ_1	Attenuation coefficient of m_t
ζ_2	Attenuation coefficient of v_t
Abbreviations	
ACE	Area control error
AGC	Automatic generation control
BN	Batch normalization
DC	Difference of convex
DNNs	Deep neural networks
EEMD	Ensemble EMD
EMD	Empirical modal decomposition
GANs	Generative adversarial networks
IAE	Integral absolute error
IMFs	Intrinsic mode functions
ISE	Integral squared error
ITAE	Integral time multiple absolute error
LAC-GANs	Lightweight actor-critic generative adversarial networks
PID	Proportional-integral-derivative
RMSE	Root mean square error
SGANs	Sequence generative adversarial networks
SGC	Smart generation control
TD	Temporal difference
MAE	Mean absolute error
MAPE	Mean absolute percentage error
MCP	Minimax concave penalized
MSE	Mean square error

Represented by proportional-integral-derivative (PID) [7], the AGC strategy determines the total regulated power singularly based on frequency deviation or area control error (ACE) [8], then controls the power generation of generation units. However, the conventional AGC strategy cannot dynamically learn, analyze, and judge in the operation process [9]. Therefore, the SGC strategy has been proposed to replace the conventional AGC strategy [6]. Represented by reinforcement learning [10] based on the Markov process, the SGC strategy repeatedly trials during interacting with the environment and obtains the optimal control strategy by continuously learning information. As the typical reinforcement learning, Q learning [11] as the SGC strategy can effectively enhance the adaptability and robustness of microgrids. However, regular reinforcement learning has three deficiencies: (i) the size of the state space may increase rapidly in the complex system, which causes “the curse of dimensionality” [12]; (ii) the fluctuation of wind power and solar power further decrease the frequency regulation ability of microgrids [13]; (iii) the frequency regulation performance and the economic benefits are not considered simultaneously [14].

Various deep learning algorithms are applied to overcome the above deficiencies. Deep reinforcement learning, such as deep deterministic policy gradient and soft actor-critic, combines deep neural networks (DNNs) [15] and reinforcement learning can effectively mitigate “the curse of dimensionality” [16]. Transfer learning [17] can effectively utilize historical optimization information for fast dynamic power allocation, which can decrease the impact of wind and solar stochastic fluctuations on the stability of microgrids. Hybrid deep neural networks combine optimization algorithms with backpropagation neural networks [18], which can reduce the frequency deviation and generation cost simultaneously. To learn in more complex random environments, long-term historical system information and the immediate systemic state are considered simultaneously. Therefore, deep learning requires a large amount of long-term historical system information for pre-training. However, various deep learning is usually supervised learning [19], such as DNNs, transfer learning, and backpropagation neural networks. Since the complexity of microgrids, complex power data requires a lot of labor and time resources for labeling, which limits

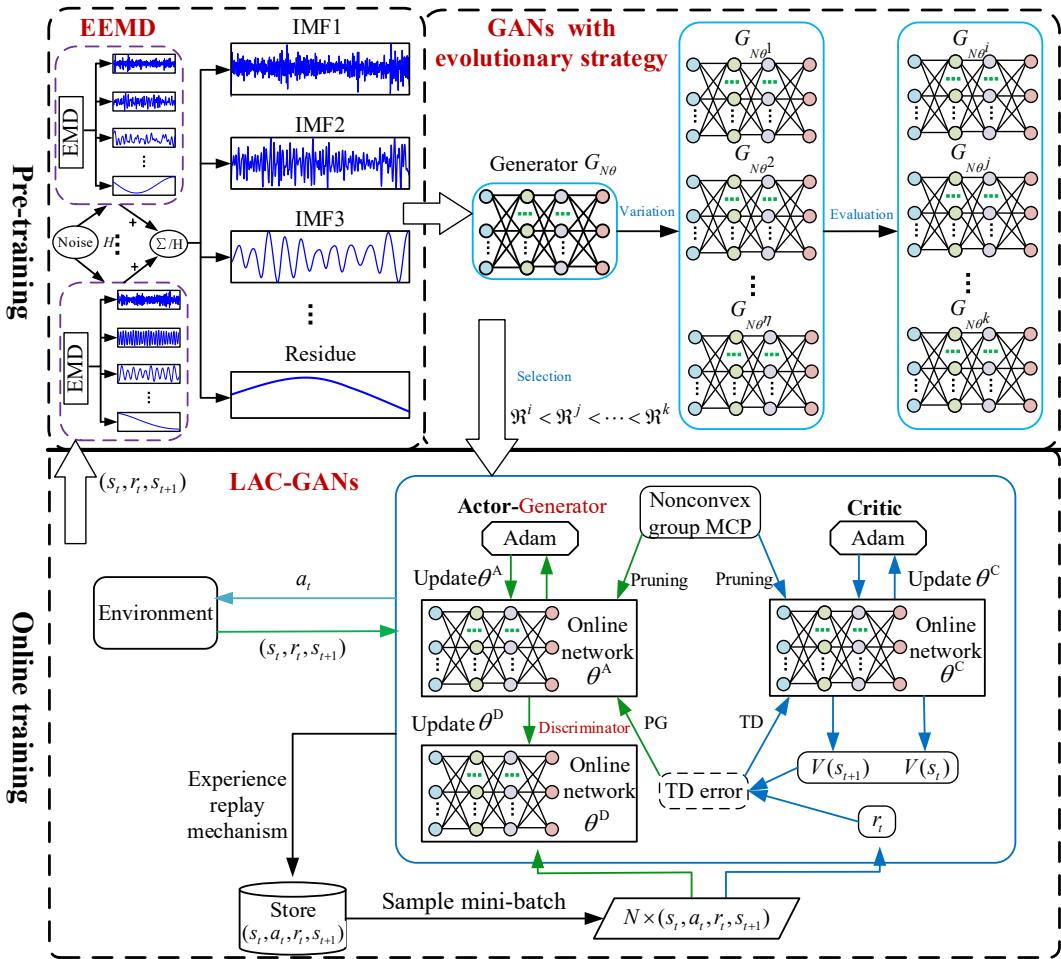


Fig. 1. Framework of LAC-GANs.

the practical application in SGC of deep learning.

To reduce the labor and time resources for labeling, unsupervised learning can be applied to the SGC. The generative adversarial networks (GANs) method [20] is one representative of unsupervised learning, which means that the training data does not require to be labeled. Besides, GANs can be utilized to predict the next action of microgrids, which allows the microgrid to make the optimal control decision in advance. However, the GANs often occur gradient disappearance and pattern collapse [21]. Numerous GANs variants can solve these problems, such as least squares GANs [22], Wasserstein GANs [23], and deep convolutional GANs [24]. However, the individual adversarial training target of all these GANs variants is unchanged; the training problem of GANs cannot be solved fundamentally [25]. Fortunately, the training efficiency of GANs can be improved effectively by reducing the complexity of the data [26]. Furthermore, the complex power data can be decomposed into more regular and simpler sub-data by the ensemble empirical modal decomposition (EEMD) [27], which can facilitate the model fitting of GANs. In addition, evolutionary strategies have been successfully applied to solve optimization problems for complex deep learning models [28]. The GANs can be optimized by the evolutionary strategy that considers the individual adversarial training process as an evolutionary process with different loss functions, which can optimize the parameters of GANs from multiple adversarial targets and improve the training stability of GANs [29].

GANs can be applied in the reinforcement learning field [30]. For example, the sequence GANs (SGANs) [31] combine the GANs and policy gradient (PG) [10], which has the abilities of dynamic learning in real-time systems. However, the PG in SGANs suffers from low learning

efficiency and instability [32]. Compared with the PG, the actor-critic strategy is more stable and efficient [33]. Besides, the GANs can be connected to the actor-critic framework [34]. In adversarial advantage actor-critic [35], the discriminator of GANs can be applied as another critic, which can accelerate policy exploration efficiently. In adversarial guided actor-critic [36], the discriminator is utilized as an adversary to predict the action of the actor, which successfully adds the behavior diversity of the actor. The GANs are applied to the actor-critic framework that is a promising solution to achieve the intelligent control strategy for SGC and improve microgrid adaptation. Nevertheless, the above study still exists two unresolved deficiencies: (i) only the discriminator is introduced the actor-critic framework, which cannot be fully connected to the GANs; (ii) the GANs can be combined with the actor-critic framework to obtain excellent performance, which inevitably results in high demands on storage and calculation resources [37].

To address the above deficiencies, this paper proposes lightweight actor-critic generative adversarial networks (LAC-GANs) based on EEMD and evolutionary strategy for the SGC. To fully connect the GANs with the actor-critic framework, the generator is utilized as the actor to predict the selection action; the discriminator is applied as another critic to guide the generator. Besides, to break the correlation of online training data, the empirical replay mechanism is introduced into LAC-GANs. Meanwhile, the multi-path lightweight method based on the nonconvex group minimax concave penalized (MCP) regularization [38] is proposed to prune the LAC-GANs, which reduce time and storage resources. Compared with the common lightweight methods, such as with group Lasso regularization [39], the nonconvex group MCP regularization has stronger sparsity and unbiasedness with better lightweight

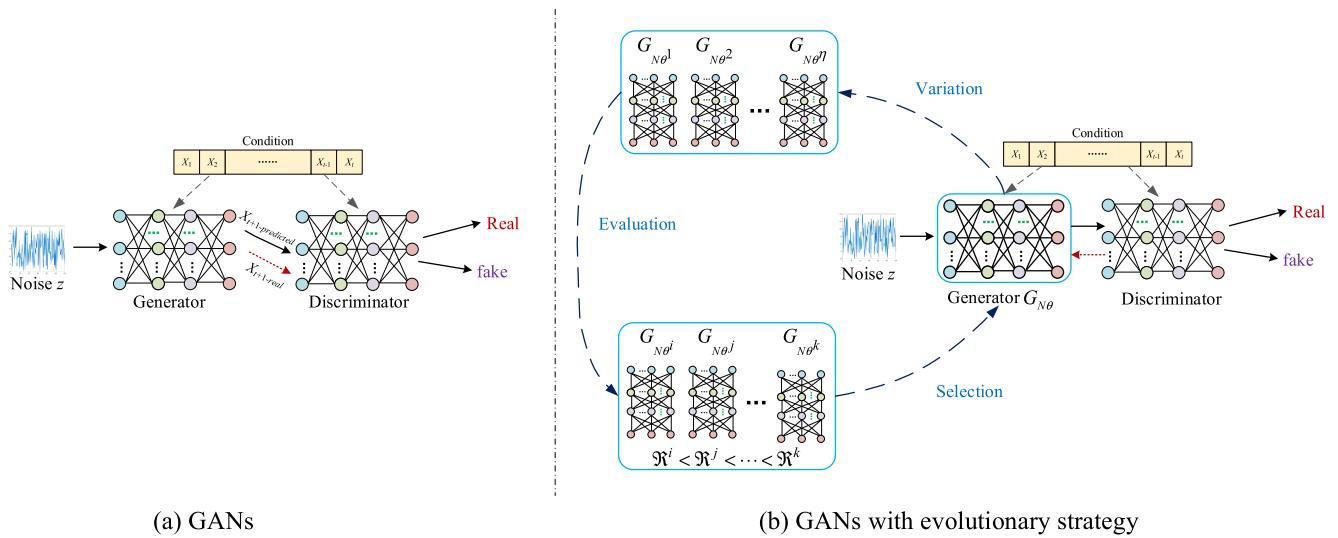


Fig. 2. Structures of two GANs: (a) GANs; (b) GANs with evolutionary strategy.

results [40]. Besides, the difference with convex (DC) decomposition [41] is applied to convert the nonconvex function into a convex function for optimizing the nonconvex group MCP regularization. After pre-training and online training, the LAC-GANs can obtain the optimal control strategy with less computing time and storage resources in a random environment. The major contributions of this paper are summed up as follows.

- (1) The EEMD is utilized to improve the training speed of GANs by splitting complex power data into simpler sub-data. Meanwhile, to mitigate the training problems of GANs, the evolutionary strategy is utilized to optimization the GANs.
- (2) The LAC-GANs fully connect the GANs and the actor-critic framework. In addition, the LAC-GANs can interact with the environment in real-time by the actor-critic strategy and store the sample for online training by the empirical replay mechanism.
- (3) The multi-path lightweight method is proposed to prune the LAC-GANs and reduce the consumption time. Furthermore, to optimize the nonconvex group MCP regularization, this paper applies the DC decomposition to convert the nonconvex function into the difference with two convex functions.
- (4) The SGC system based on LAC-GANs considers control performance and economic efficiency simultaneously and accommodates multiple distributed energies, such as wind, solar, and energy storage.

The remainder of the paper is separated into the following parts. **Section 2** covers LAC-GANs. **Section 3** deals with SGC systems. **Section 4** displays case studies. Finally, section 5 briefly summarizes this work.

2. Framework of lightweight actor-critic generative adversarial networks

The framework of the LAC-GANs is illustrated in Fig. 1. Firstly, history power data is decomposed into more regular and simpler sub-data by EEMD. Secondly, the evolutionary strategy pre-trains the GANs; the evaluation function selects the optimal generator. Thirdly, the optimal generator is reused as the actor and trained online with the critic by actor-critic strategy. Meanwhile, the multi-path lightweight method prunes the actor and critic. Eventually, a strategy that can satisfy the optimal performance with less computing time and storage resources in a random environment is achieved.

2.1. Ensemble empirical mode decomposition

The empirical mode decomposition (EMD) adaptively decomposes complex data into finite intrinsic mode functions (IMFs) and residuals [42]. The IMFs are the individual frequency components of the original data. Therefore, the EMD is widely applied to process the complex training data of deep learning models. However, if the signal polarization points are not uniformly distributed, the EMD could occur the signal mode overlap in the extracting process of data signal characteristics. To avoid the signal mode overlap, the EEMD is presented. First, a finite Gaussian white noise is added to the original data. Then, the EMD multiply decomposes the noise-added data; the average multiply decomposed noise-added data is obtained as the final IMFs. The EEMD adds white noise to the original data and utilizes the uniform distribution of the white noise spectrum to enable the original data to be automatically distributed to a suitable reference scale. Because of the characteristic of zero-mean noise, the noise cancels each other after several averages calculations. The result of the integrated mean is the final result.

The complementary EEMD (CEEMD) [43] adds Gaussian white noise in pairs to the original data, which effectively reduces the reconstruction error of the signal. However, the completeness of CEEMD cannot be proved. The complete EEMD with adaptive noise (CEEMDAN) [44], as an improvement of CEEMD, has almost zero reconstruction error. Nevertheless, the CEEMDAN remains unable to eliminate the residual Gaussian white noise. Compared to the CEEMD and CEEMDAN, the EEMD is the simplest and most effective way to decompose complex data.

The operation random, non-smooth and intermittent cause the power data of microgrids to be complex, which reduces the training speed of deep learning models. Therefore, this paper applies the EEMD to decompose data for increasing the training speed of GANs. The specific realization of the EEMD is shown below.

- (1) Append noise $\ell(t)$ to original data $x(t)$ to acquire noise-added data $x'(t)$, as [27]

$$x'(t) = x(t) + \ell(t) \quad (1)$$

- (2) Decompose $x'(t)$ to J finite IMFs and a residue $\vartheta(t)$ [27]

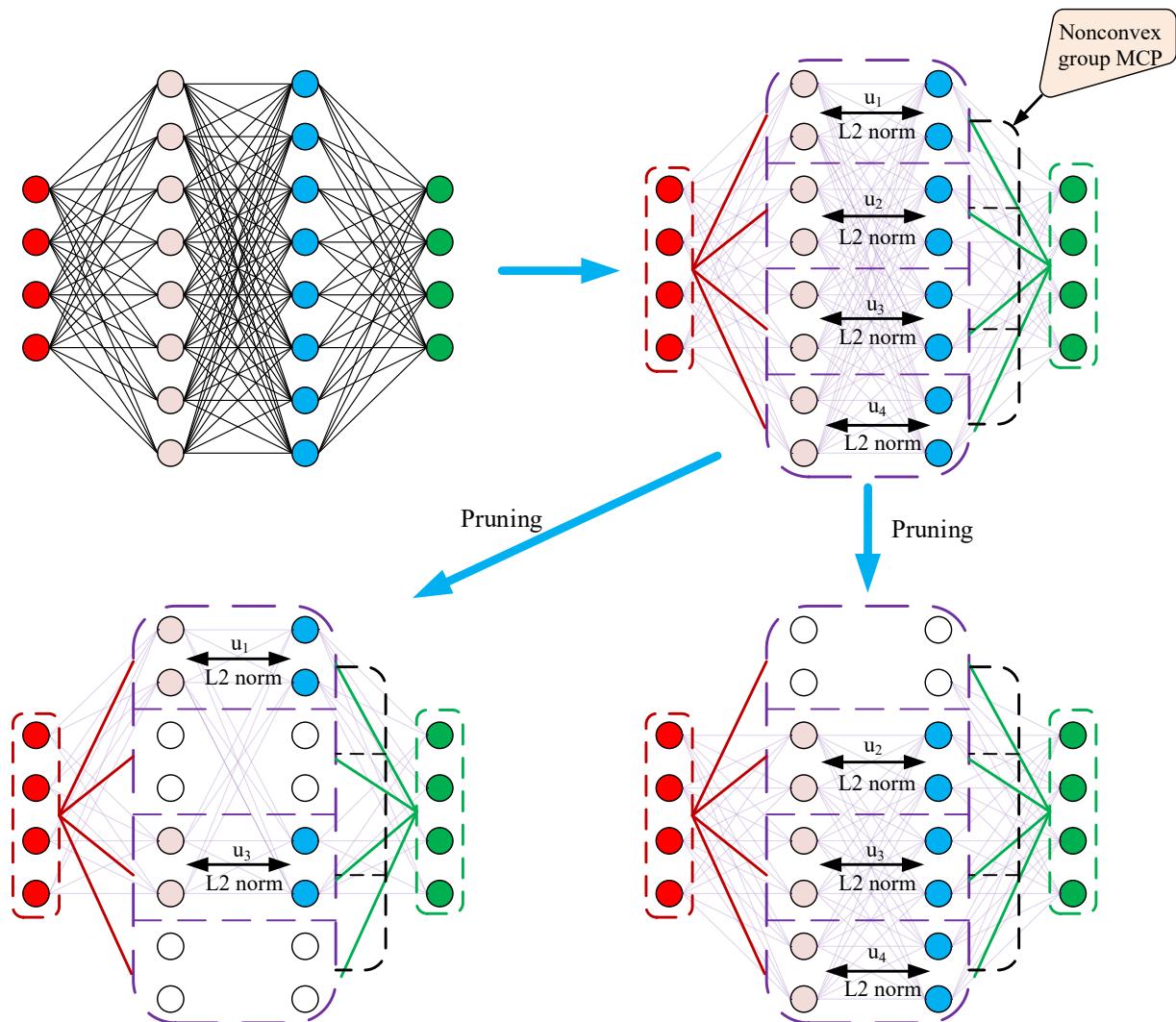


Fig. 3. Multi-path structure.

$$x^*(t) = \sum_{j=1}^J IMF_j(t) + \vartheta(t)$$

(3) Multiply decompose $x^*(t)$ with H times as [27]

$$IMF_f(t) = \frac{1}{H} \sum_{h=1}^H IMF_{jh}(t) \quad (3)$$

where H is the decomposition times of noise-added data $x^*(t)$ by EMD.

(4) The final data $X(t)$ is obtained as [27]

$$X(t) = \sum_{f=1}^J IMF_f(t) + \varepsilon(t) \quad (4)$$

where $\varepsilon(t)$ is the final residue; $IMF_f(t)$ is the final IMFs.

2.2. Generative adversarial networks with evolutionary strategy

The GANs are deep learning models based on the generator and discriminator. The generator generates the most realistic data. In contrast, the discriminator distinguishes real and fake data as correctly as possible. The generator input is the random noise z . The discriminator

inputs are real and fake data generated by the generator. The GAN updates the generator and discriminator by the alternated iteration processes to solve the min-max game problem of GANs.

The loss function of the generator is minimum optimization, as [20]

$$\min V_G = E_{z \sim \rho_{\text{noise}}} [\log(1 - D(G(z)))] \quad (5)$$

The discriminator loss function is maximum optimization, as [20]

$$\max V_D = E_{x \sim \rho_{\text{data}}} [\log D(x)] + E_{z \sim \rho_{\text{noise}}} [\log(1 - D(G(z)))] \quad (6)$$

The whole loss function of GANs is [20]

$$\min_{G} \max_{D} V(D, G) = E_{x \sim \rho_{\text{data}}} [\log D(x)] + E_{z \sim \rho_{\text{noise}}} [\log(1 - D(G(z)))] \quad (7)$$

where ρ_{data} means real data distribution; ρ_{noise} means noise distribution; $E(\bullet)$ is the expected value.

The GANs can adjust the model on some conditional information (Fig. 2 (a)), as condition GANs. Therefore, the GANs can be applied to probabilistic predictions [45]. Besides, to make optimal control decisions one step ahead, the GANs can be utilized to precisely predict the next systemic action. When the sample set $X(t) = \{X_1, \dots, X_t\}$ is obtained by EEMD, the GANs are applied to model the probability distribution of predictive value X_{t+1} , i.e., $\rho(X_{t+1}|X(t))$. The sample set $X(t)$ is connected as the condition of the generator and discriminator. The mean value of the Gaussian noise is 0, and the standard deviation is 1. The Gaussian noise is utilized as the input of the generator. The discriminator obtains

the X_{t+1} forecasted by generator and checks whether X_{t+1} is the valid value that follows $X(t)$.

For the GANs with evolutionary strategy, the generator is regarded as a competing population; the discriminator is viewed as a competitive environment. The generator population can gradually adapt to the competitive environment by different variations during evolution. Different variations aim to generate offspring with different optimization paths to adapt to the competitive environment, which allows the GANs to approach the global optimum with different directions during the evolutionary process. In the evaluation process, the optimal offspring is selected based on a fitness function and participates in the next evolutionary process. Finally, the evolved offspring (i.e., the optimal generator) can generate increasingly accurate data. Therefore, this paper utilizes the GANs to predict the next action of microgrids to find the optimal control strategy. Meanwhile, the GANs are optimized to improve the training stability and accuracy by evolutionary strategy. Eventually, the optimal generator is selected for further training.

As shown in Fig. 2 (b), the evolutionary process consists of three steps.

- Variation: the generator $G_{N\theta}$ produces multiple offspring $\{G_{N\theta^1}, \dots, G_{N\theta^U}\}$ by different loss functions.
- Evaluation: the performance of each offspring is assessed by a fitness function $\mathfrak{R}(\bullet)$ based on the discriminator.
- Selection: the underperforming generators are eliminated based on the fitness value; the remaining generators are utilized for the next evolutionary process.

2.3. Lightweight actor-critic generative adversarial networks

The LAC-GANs can be seen as a deep reinforcement learning model with an actor-critic framework. The experience replay mechanism is applied to store the real-time sample, which is utilized for online updating LAC-GANs. The generator is utilized as the actor to predict the selection action; the discriminator is applied as another critic to guide the generator. The LAC-GANs fully connect the GANs with the actor-critic framework, which enables the GANs can continuously online learn until satisfying the optimal control performance in random environments. The LAC-GANs contain an actor (i.e., the optimal generator), a discriminator, and a critic. The inputs of the LAC-GANs are the samples (s_t, a_t, r_t, s_{t+1}) gathered by the experience replay mechanism [46] in real-time; the predicted action a_{t+1} is output by the current actor. Within a time-step, the LAC-GANs act a_t in the current state s_t and obtain the reward r_t .

Since the actor-critic strategy is limited by parameters redundancy and hard to deploy to devices with finite computing resources [47], the multi-path lightweight method is proposed to alleviate the problem of parameters redundancy and to reduce the consumption of time and storage resources. First, the actor and critic are multi-path structured. Second, the temporal difference (TD) [48] and the PG are utilized to the loss function of the actor and critic, respectively. Third, the nonconvex group MCP regularization is applied to the intergroup sparsity constraint. The DC decomposition can convert a nonconvex function into the difference with two convex functions [41]. Since the nonconvex function is hard to optimize, the nonconvex function of the nonconvex group MCP regularization can be expressed as two convex functions by the DC decomposition; the Adam optimizer [49] is utilized to update the actor and critic. Finally, the sub-paths with smaller parameter values are pruned; thus, the multi-path lightweight is realized. The details are as follows.

(a) Multi-path structure

According to the neuron number of the hidden layer, the actor and critic are divided into u parallel sub-paths; each sub-path is defined as a

group. The neuron number in the input and output layers is the same as the original networks. For example, the network structure 4-8-8-4 are divided into four sub-paths with the structure 4-2-2-4 (Fig. 3). The input of each sub-path is the same as the original networks; the output of the sub-paths is aggregated in the output layer. If the sub-path parameter expectation is smaller than the weight threshold β , the sub-path is pruned (i.e., the sub-path is removed). The multi-path pruning differs from dropout. In the pruning phase, parameters are discarded forever after pruning. Besides, the parameters that are smaller than the weight threshold β are pruned. Thus, multi-path pruning does not suffer from the bias issue.

(b) Loss function

The critic updates the parameter by minimizing TD error. Meanwhile, the L2 norm is imposed on the parameters within the group.

All parameters in LAC-GANs are represented by the θ . TD error is [11]

$$\sigma = r_t + \gamma V(s'_{t+1}, \theta) - V(s_t, \theta) \quad (8)$$

where σ is the TD error; r_t is the reward; γ means discount coefficient; $V(s_t, \theta)$ and $V(s'_{t+1}, \theta)$ are reward expectations in current state s and next state s' , respectively

The expression of the nonconvex group MCP regularization is [38]

$$\varphi(||\theta_l^u||_2) = \begin{cases} \lambda ||\theta_l^u||_2 - \frac{||\theta_l^u||_2^2}{2\mu}, & 0 \leq ||\theta_l^u||_2 \leq \lambda \mu \\ \frac{\mu \lambda^2}{2}, & ||\theta_l^u||_2 > \lambda \mu \end{cases} \quad (9)$$

where λ is the regular coefficient; μ is the adjustment coefficient; $||\bullet||_2$ is the L2 norm; θ_l^u is the weight matrix of the l -th layer of the u -th sub-path.

The loss function of the critic is

$$V_{\text{Critic}}(\theta^C) = E(\sigma^2) + \sum_{l=1}^L \sum_{u=1}^U \varphi(||\theta_l^u||_2) \quad (10)$$

where L is the number of hidden layers of each sub-path; U is the number of sub-paths; θ^C means all the critic parameters.

The actor updates the parameter by maximizing the expected value of the cumulated reward. The control policy of the actor can be considered as a K -step prediction process; the predicted trajectory τ of the actor in the environment is

$$\tau = \{s_1, a_1, r_1, s_2, a_2, r_2, s_k, a_k, r_k, \dots, s_K, a_K, r_K\} \quad (11)$$

The cumulated reward under the predicted trajectory is

$$r(\tau) = \sum_{k=1}^K r_k \quad (12)$$

Similar to the critic, the loss function of the actor is

$$V_{\text{Actor}}(\theta^A) = E(r(\tau)p(\tau|\theta)) + \sum_{l=1}^L \sum_{u=1}^U \varphi(||\theta_l^u||_2) \quad (13)$$

where $p(\tau|\theta)$ is the occurrence probability of the predicted trajectory τ in the actor parameter θ^A .

(c) Network updating and lightweight process

Since the nonconvex function of the group MCP regularization is hard to optimize, the DC decomposition technique decomposes the objective function into convex functions. For the critic, the objective function is decomposed into the two convex functions $g_1(\theta)$ and $g_2(\theta)$, as

$$V_{\text{Critic}}(\theta^C) = g_1(\theta) - g_2(\theta) \quad (14)$$

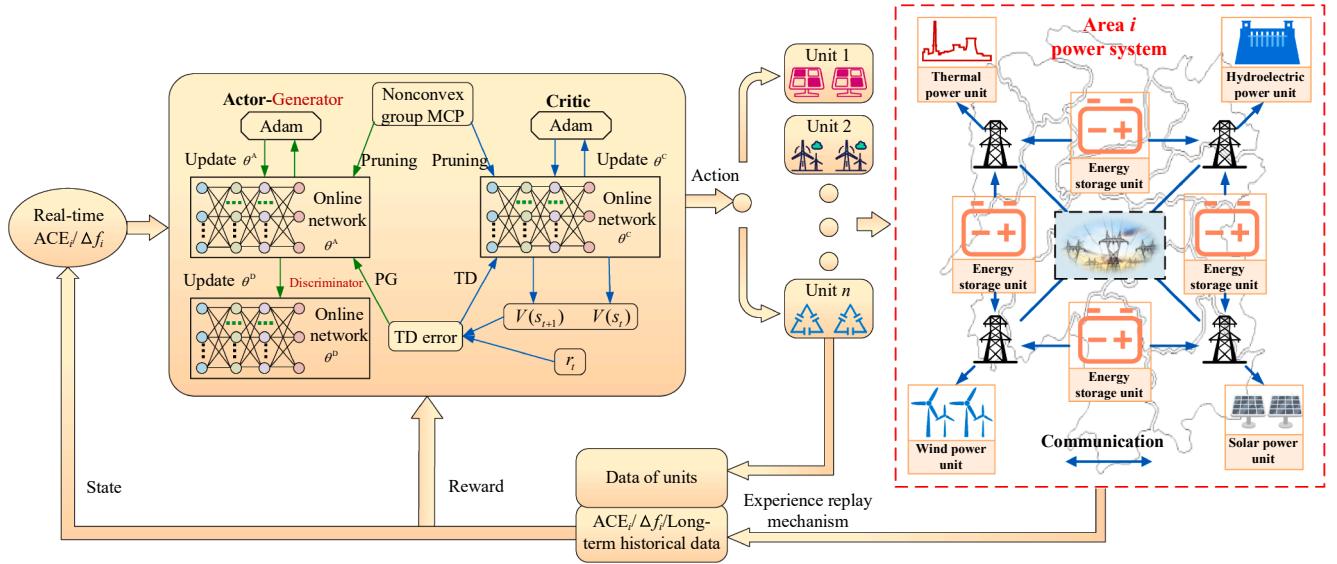


Fig. 4. SGC based on LAC-GANs.

Table 1
Structure parameters of GANs.

Module	Layer	Hidden units	Active function	BN
Generator-Actor	1	10	ReLU	32
Generator-Actor	2	100	ReLU	32
Generator-Actor	3	100	ReLU	64
Generator-Actor	4	4	Tanh	–
Discriminator	1	10	Leaky ReLU	64
Discriminator	2	100	ReLU	64
Discriminator	3	100	ReLU	128
Discriminator	4	2	Sigmoid	–
Critic	1	10	ReLU	32
Critic	2	100	ReLU	32
Critic	3	100	ReLU	64
Critic	4	4	Linear	–

Note: “–” means BN is not employed in the output layer.

Table 2
Parameters of LAC-GANs.

Symbol	Parameter	Value
α_a	Learning rate of actor	0.001
α_c	Learning rate of critic	0.001
γ	Discount factor	0.94
λ	Regular coefficient of the nonconvex group MCP regularization	0.8
μ	Adjustment coefficient of the nonconvex group MCP regularization	0.5
β	Weight threshold of actor and critic	40%

$$g_1(\theta) = E(\sigma^2) + \lambda \sum_{l=1}^L \sum_{u=1}^U \|\theta_l^u\|_2 \quad (15)$$

$$\begin{aligned} g_2(\theta) &= \lambda \sum_{l=1}^L \sum_{u=1}^U \|\theta_l^u\|_2 - \sum_{l=1}^L \sum_{u=1}^U \varphi(\|\theta_l^u\|_2) \\ &= \sum_{l=1}^L \sum_{u=1}^U \left\{ \begin{array}{ll} \frac{\|\theta_l^u\|_2^2}{2\mu}, & 0 \leq \|\theta_l^u\|_2 \leq \lambda\mu \\ \lambda\|\theta_l^u\|_2 - \frac{\mu\lambda^2}{2}, & \|\theta_l^u\|_2 > \lambda\mu \end{array} \right. \end{aligned} \quad (16)$$

Then, the derivative ω_c of $g_2(\theta)$ and the θ are calculated from the following objective function.

$$\theta = \operatorname{argmin}(g_1(\theta) - \langle \theta, \omega_c \rangle) \quad (17)$$

Substitute the expression of $g_1(\theta)$ to Eq. (17), as

$$\theta = \operatorname{argmin} J_c(\theta) \quad (18)$$

$$J_c(\theta) = E(\sigma^2) + \lambda \sum_{l=1}^L \sum_{u=1}^U \|\theta_l^u\|_2 - \langle \theta, \omega_c \rangle \quad (19)$$

where $\langle \theta, \omega_c \rangle$ is the inner product of θ and ω_c .

The parameter θ is updated by the Adam optimizer. At the k -th iteration, the gradient $g_k(\theta)$ of $J_c(\theta)$, first-order moment estimation m_k , and second-order moment estimation v_k are calculated [41].

$$m_k = \zeta_1 m_{k-1} + (1 - \zeta_1) g_k(\theta) \quad (20)$$

$$v_k = \zeta_2 v_{k-1} + (1 - \zeta_2) g_k^2(\theta) \quad (21)$$

where ζ_1 and ζ_2 are the attenuation coefficients of m_k and v_k , respectively; both ζ_1 and ζ_2 are set to be 0.5 in this paper.

The critic parameter θ^C is updated as

$$\theta^C = \theta_{k-1} + \alpha_c m_k / \sqrt{v_k}, \quad (22)$$

where α_c means learning rate; m_k and v_k are the deviation corrections of m_k and v_k , respectively.

Finally, when the expectation of the u -th sub-path satisfies $E(\theta_u) < \beta$, the sub-path is removed, and the critic is updated.

For the actor, the objective function is decomposed into two convex functions $g_3(\theta)$ and $g_4(\theta)$, as

$$V_{\text{Actor}}(\theta^A) = g_3(\theta) - g_4(\theta) \quad (23)$$

$$g_3(\theta) = E(r(\tau)p(\tau|\theta)) + \lambda \sum_{l=1}^L \sum_{u=1}^U \|\theta_l^u\|_2 \quad (24)$$

$$\begin{aligned} g_4(\theta) &= \lambda \sum_{i=1}^L \sum_{u=1}^U \|\theta_i^u\|_2 - \sum_{i=1}^L \sum_{u=1}^U \varphi(\|\theta_i^u\|_2) \\ &= \sum_{l=1}^L \sum_{u=1}^U \left\{ \begin{array}{ll} \frac{\|\theta_l^u\|_2^2}{2\mu}, & 0 \leq \|\theta_l^u\|_2 \leq \lambda\mu \\ \lambda\|\theta_l^u\|_2 - \frac{\mu\lambda^2}{2}, & \|\theta_l^u\|_2 > \lambda\mu \end{array} \right. \end{aligned} \quad (25)$$

The actor parameters are calculated from the following objective function.

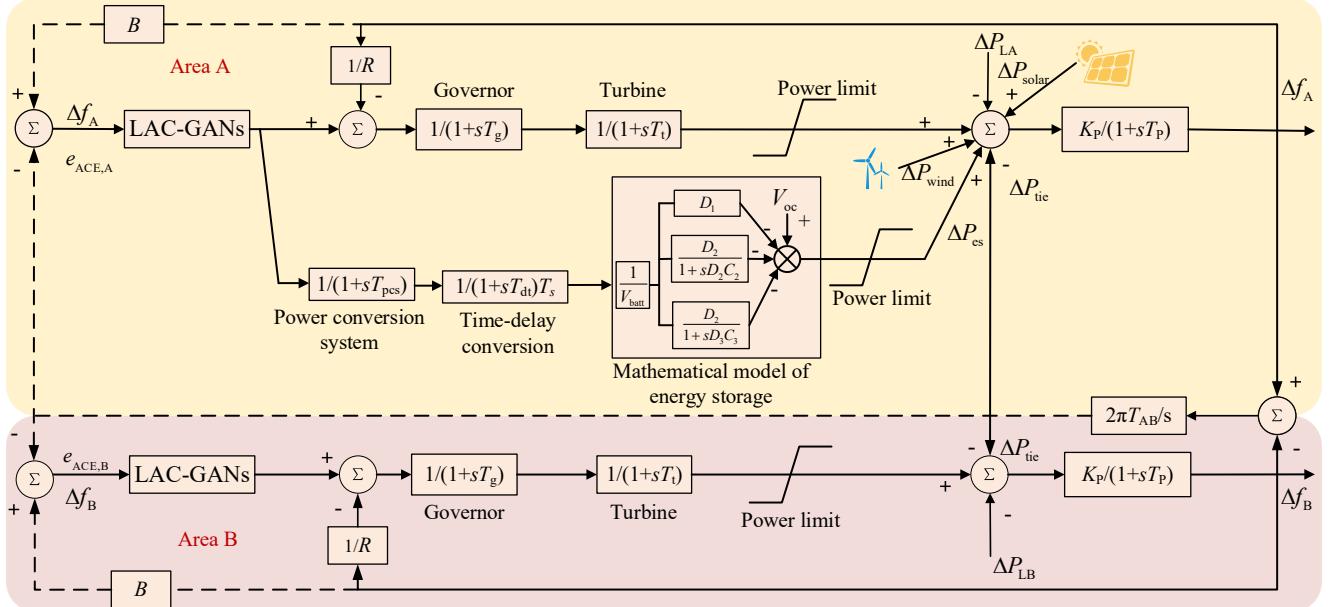


Fig. 5. IEEE two-area system.

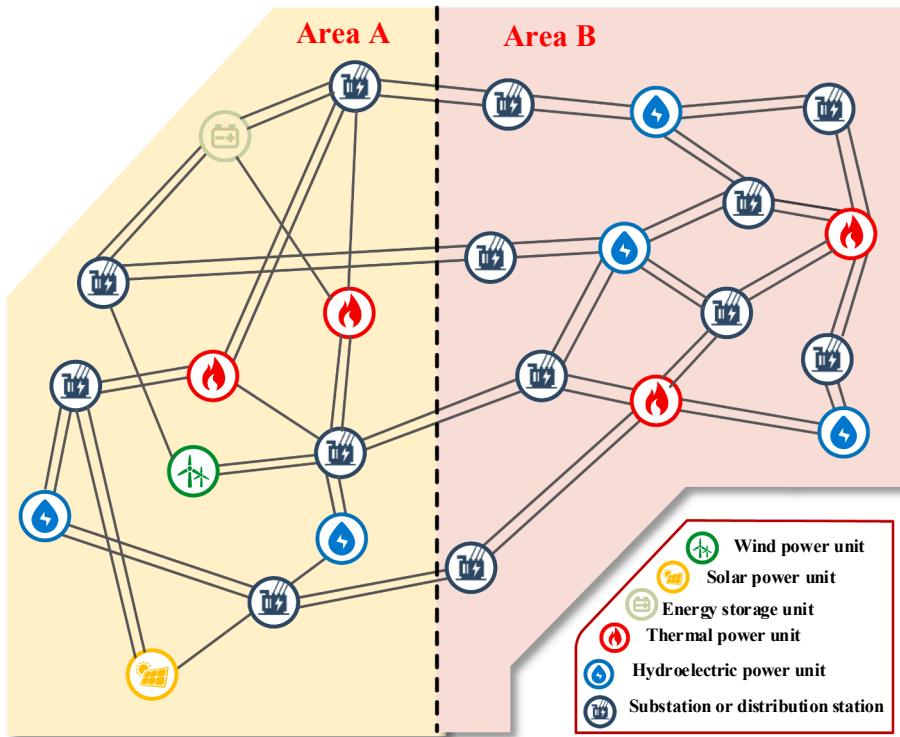


Fig. 6. The power grid sketch of Case I.

$$\theta = \text{argmin} J_a(\theta)$$

(26)

$$J_a(\theta) = g_3 - \langle \theta, \omega_a \rangle = E(r(\tau)p(\tau|\theta)) + \lambda \sum_{l=1}^L \sum_{u=1}^U \|\theta_l^u\|_2 - \langle \theta, \omega_a \rangle$$

(27)

where the ω_a is the derivative of $g_4(\theta)$; $\langle \theta, \omega_a \rangle$ is the inner product of θ and ω_a .

Then, the parameters of the actor are updated by the Adam optimizer. Meanwhile, the gradient $g_k(\theta)$ of $J_a(\theta)$ is calculated; baseline b is

introduced to reduce variance.

$$g_k(\theta) = \nabla_\theta \left[\log(r(\tau) - b) + \lambda \sum_{l=1}^L \sum_{u=1}^U \|\theta_l^u\|_2 \right]$$

The TD error updates the actor. The $r(\tau) - b$ is replaced by the σ .

$$g'_k(\theta) = \nabla_\theta \left[\log\sigma + \lambda \sum_{l=1}^L \sum_{u=1}^U \|\theta_l^u\|_2 \right]$$

The m_k and v_k are calculated by the $g'_k(\theta)$ of $J_a(\theta)$, respectively. The

Table 3
System parameters of Case I.

Symbol	Parameter	Value
T_g , T_t , T_p	Governor, turbine, and frequency response time constants	0.08, 0.3, 20 s
T_s	Conversion coefficient of time-delay conversion	1.2
T_{pcs}	Time constant of conversion system	0.01 s
T_{dt}	Time constant of time-delay conversion	0.0001 s
T_{AB}	Tie-line time constant	3.42 s
K_p	Frequency response coefficient	270 Hz/p.u.
V_{batt}	Terminal voltage of energy storage unit	768 V
V_{oc}	Open circuit voltage of energy storage unit	800 V
B	Frequency bias coefficient of primary frequency regulation	4166 MW/Hz
R	Frequency deviation coefficient of secondary frequency regulation	2.4 Hz/p.u.
C_2	Capacitance of second energy storage branch	63.3F
C_3	Capacitance of third energy storage branch	641F
D_1	Resistance of first energy storage branch	0.134 Ω
D_2	Resistance of second energy storage branch	0.33 Ω
D_3	Resistance of third energy storage branch	0.98 Ω

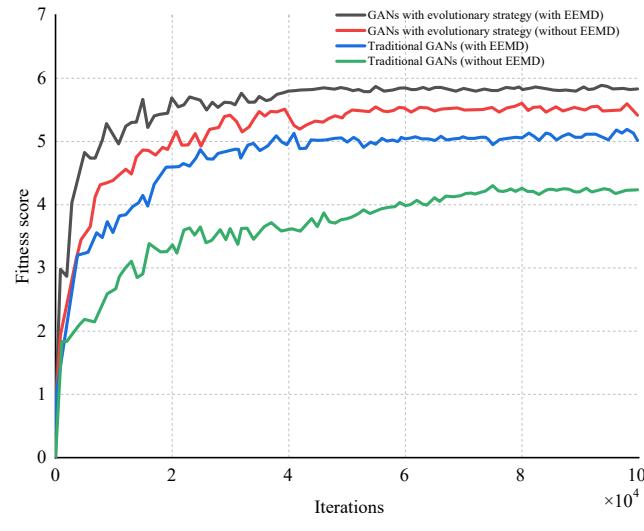
Table 4
Parameters of comparison algorithm in Case I.

Algorithm	Parameter	Value
PID	Proportional	-1400
	Integral	-10
	Derivative	0.00001
Q learning	Learning rate	0.1
	Probability coefficient	0.5
	Reward coefficient	0.9
DNNs	Actions set	{-50, -40, -25, -10, -2.5, 0, 2.5, 10, 25, 40, 50}
	Network structure	[10, 100, 100, 4]
	Learning rate	0.001
GANs	Generator	[10, 100, 100, 4]
	Discriminator	[10, 100, 100, 2]
	Learning rate	0.01

actor is updated as

$$\theta^A = \theta_{k-1} + \alpha_a m_k' / \sqrt{v_k}, \quad (30)$$

When the weight expectation of the u -th sub-path satisfies $E(\theta_u) < \beta$;



(a)

the sub-path is eliminated, and the actor is updated.

2.4. Training process of lightweight actor-critic generative adversarial networks

The training process of the LAC-GANs includes pre-training and online training. The evolutionary strategy updates the GANs in pre-training. After pre-trained, the generator network is reused as the actor in LAC-GANs. The actor and the critic are updated by the multipath lightweight method. Meanwhile, the actor is trained with the discriminator; the discriminator is updated by Eq. (20) with mini-

Table 5
Errors obtained GANs with two updating strategies.

Model	RMSE (kW)	MAPE (%)
Traditional GANs with EEMD	26.1258	2.2155
Traditional GANs without EEMD	33.2546	2.7568
GANs with evolutionary strategy (with EEMD)	18.2724	1.3681
GANs with evolutionary strategy (without EEMD)	22.5679	1.7238

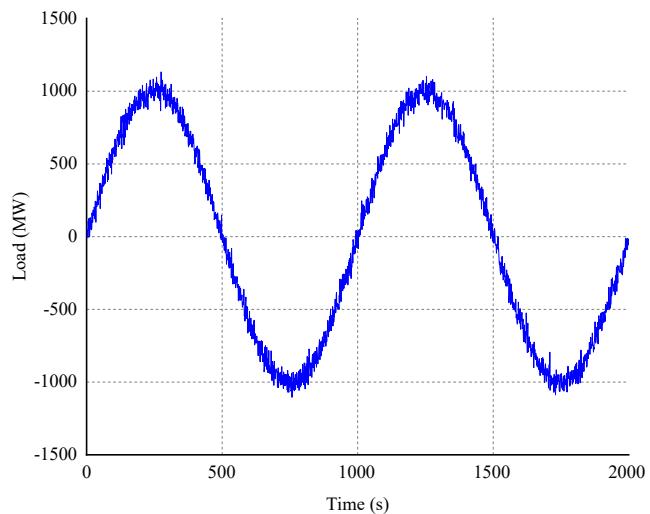
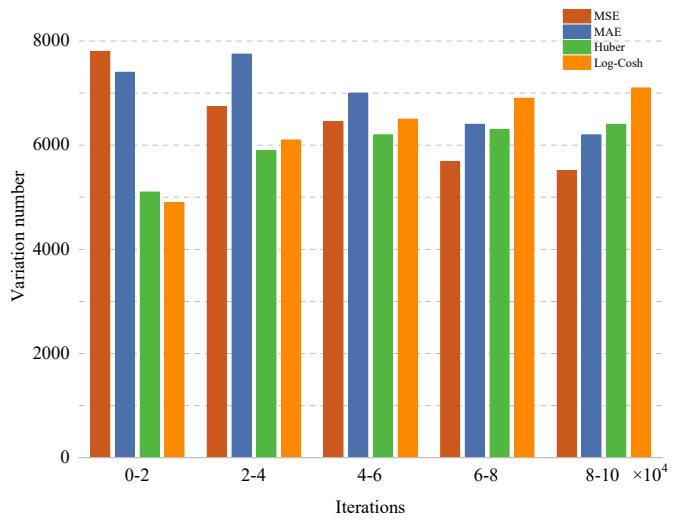


Fig. 8. Sinusoidal load with Gaussian white noise.



(b)

Fig. 7. Training process of GANs with different updating strategies: (a) fitness score of GANs with different updating strategies; (b) variation number of GANs with evolutionary strategy.

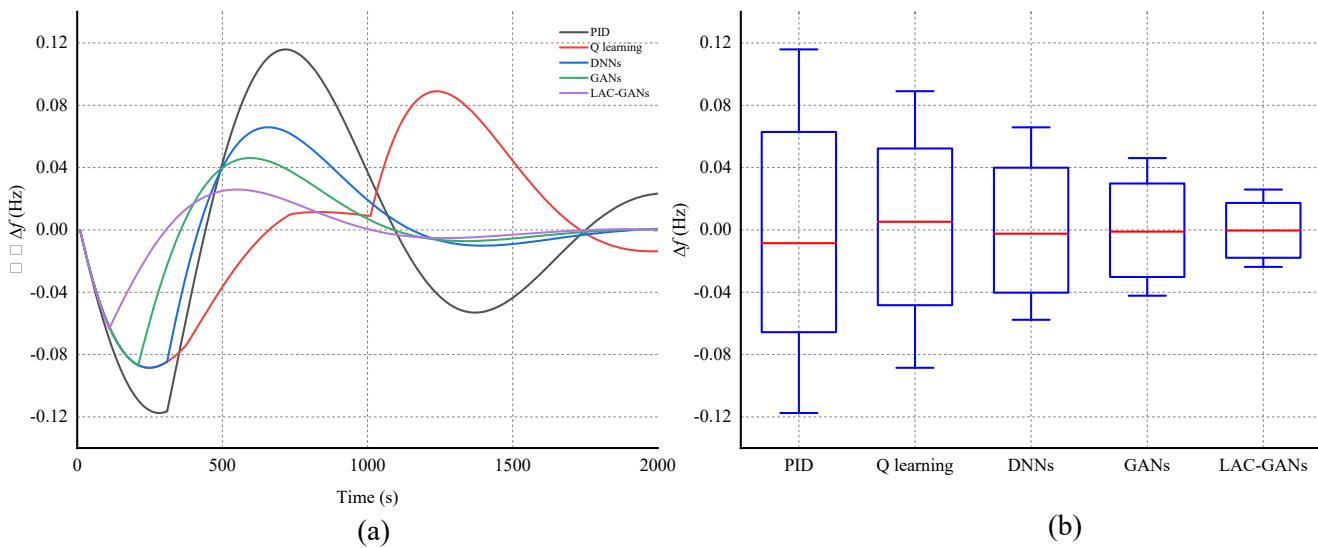


Fig. 9. Frequency deviations obtained by comparison algorithms: (a) frequency deviation curves; (b) box diagram of frequency deviations.

Table 6
Online training results of comparison algorithms.

Algorithm	Training time (s)	Computation time (s)	Computation memory (kB)
PID	–	0.01756	–
Q learning	130.9358	0.6824	75.2578
DNNs	206.6201	0.5634	56.2146
GANs	230.2141	0.7895	95.6473
LAC-GANs	190.3365	0.4656	43.3247

Note: “–” means PID does not need training and networks.

batches.

During the pre-training, the training data of the GANs are the history state s_t and history action a_t obtained from the controlled microgrids. First, the training data is decomposed into relatively simple and regular components by the EEMD. Then, the GANs with evolutionary strategy are trained to precisely predict the next systemic action. During the online training, the training data of the LAC-GANs are the real-time state of microgrids. Under different system states, the LAC-GANs can obtain higher frequency regulation performance with lower generation costs through constant trial and error.

To reduce the prediction error of GANs, four loss functions are introduced in the variation process of the evolutionary strategy.

(1) Mean square error (MSE): The gradient of MSE-based loss function is decreased with the decreasing of loss value; thus, the predicted values of GANs based on the MSE-based loss function is more accurate than that of mean absolute error (MAE) at the end of training. However, the GAN based on the MSE-based loss function easily falls into the local optimum.

$$\mathcal{L}^{\text{MSE}} = \frac{1}{N} \sum_{n=1}^N (y^{(n)} - G_N(x^{(n)}))^2 \quad (31)$$

where N represents training batch size; $y^{(n)}$ and $G_N(x^{(n)})$ are actual and predicted values, respectively.

(2) MAE: The GANs based on the MAE-based loss function will not fall into the local optimum. However, the gradient of the MAE-based loss function remains constant in the training process and maybe a large value at the end of training even if the loss value is small.

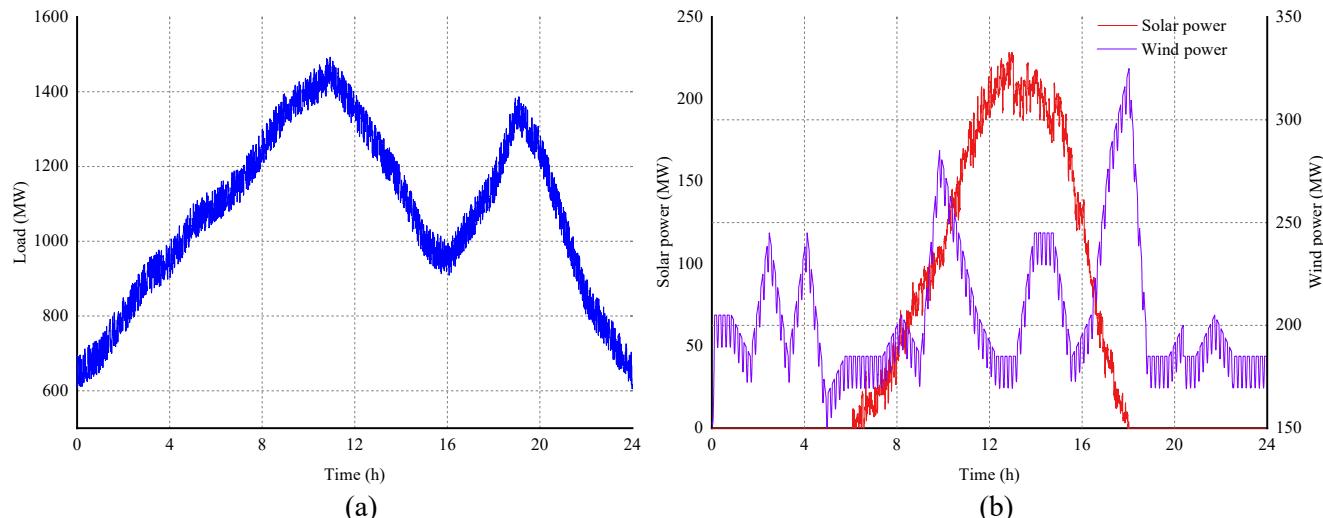


Fig. 10. Curves of loads: (a) real-life resident load; (b) wind and solar power.

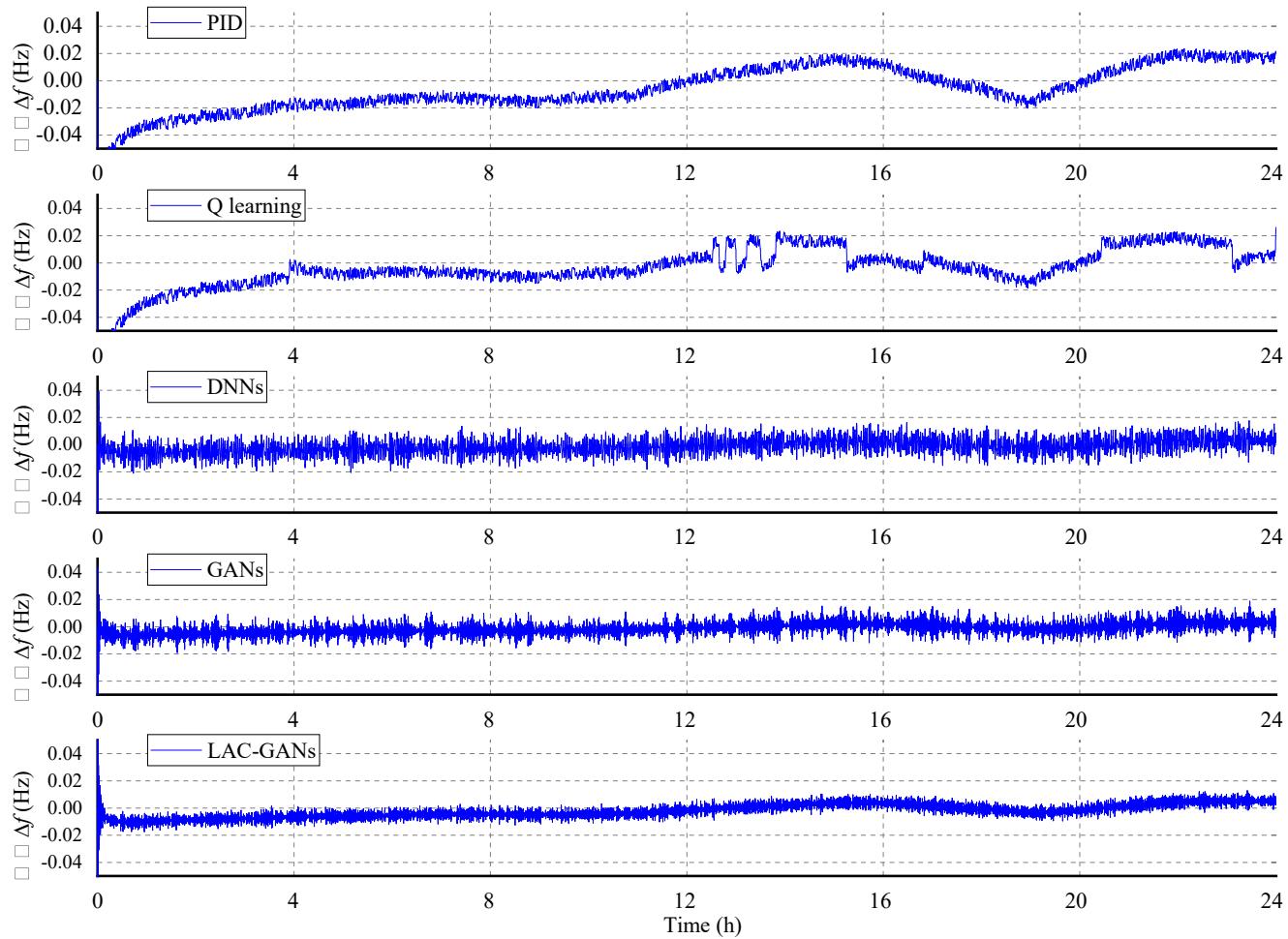


Fig. 11. Frequency deviations achieved by different algorithms (Case I).

$$\mathcal{A}^{MAE} = \frac{1}{N} \sum_{n=1}^N |y^{(n)} - G_{N\theta}(x^{(n)})| \quad (32)$$

(3) Huber loss: Huber loss combines the advantages of MSE and MAE. However, the adjustable coefficient δ of Huber loss is difficult to be determined.

$$\mathcal{A}^{Huber} = \begin{cases} \frac{1}{N} \sum_{n=1}^N \frac{1}{2} (y^{(n)} - G_{N\theta}(x^{(n)}))^2, & \text{if } (y^{(n)} - G_{N\theta}(x^{(n)})) \leq \delta \\ \frac{1}{N} \sum_{n=1}^N \delta |y^{(n)} - G_{N\theta}(x^{(n)})| - \frac{1}{2} \delta^2, & \text{otherwise} \end{cases} \quad (32)$$

(4) Log-Cosh loss: Log-Cosh loss obtains accurate prediction and will not fall into the local optimum. However, the Log-Cosh loss is plagued by hessian problems.

$$\mathcal{A}^{log-cosh} = \frac{1}{N} \sum_{n=1}^N \log(\cosh(y^{(n)} - G_{N\theta}(x^{(n)}))) \quad (33)$$

Two main metrics are considered in the evaluation process of the evolutionary strategy.

(1) Prediction accuracy: the most significant metric for assessing the performance of evolved offspring is prediction accuracy, as

$$\mathfrak{R}_p = \mathcal{A}^{MSE} \quad (34)$$

(2) Generalization ability: to effectively mitigate overfitting, the generalization ability of evolved offspring is assessed by the loss function of the discriminator, as

$$\mathfrak{R}_g = E_{x \sim \rho_{data}} [\log D(x)] + E_{z \sim \rho_{noise}} [\log(1 - D(G(z))] \quad (35)$$

The total evaluation function of evolved offspring is calculated as

$$\mathfrak{R} = \mathfrak{R}_p + \eta \mathfrak{R}_g \quad (36)$$

where η is set 0.01 to balance the two metrics.

The better performing offspring are selected based on the relatively high fitness score calculated by Eq. (36) (i.e., selection) and participate in the next evolution. Finally, the final surviving offspring is selected to participate in online training (Algorithm 1).

Algorithm 1. Pseudo code of LAC-GANs.

- 1: Initialize parameters (standard deviation of Gaussian white noise N_{std} ; the number of noise addition k ; learning rate of the actor α_a ; learning rate of the critic α_c ; discount factor γ ; regular coefficient of the nonconvex group MCP regularization λ ; adjustment coefficient of the nonconvex group MCP regularization μ ; weight threshold of actor and critic β)
- 2: Initialize GANs
- 3: Acquire history power data (state s_t and history action a_t)
- 4: Decompose history power data by the Eqs. (1)-(4)
- 5: Take the decomposed data as the input of GANs
- 6: Pre-train GANs by the evolutionary strategy as Fig. 2
- 7: Optimize the GANs by the Eqs. (31)-(33)
- 8: Select the optimal generator as the actor of LAC-GANs by the Eq. (36)
- 9: Multi-path structure the actor and critic as Fig. 3

(continued on next page)

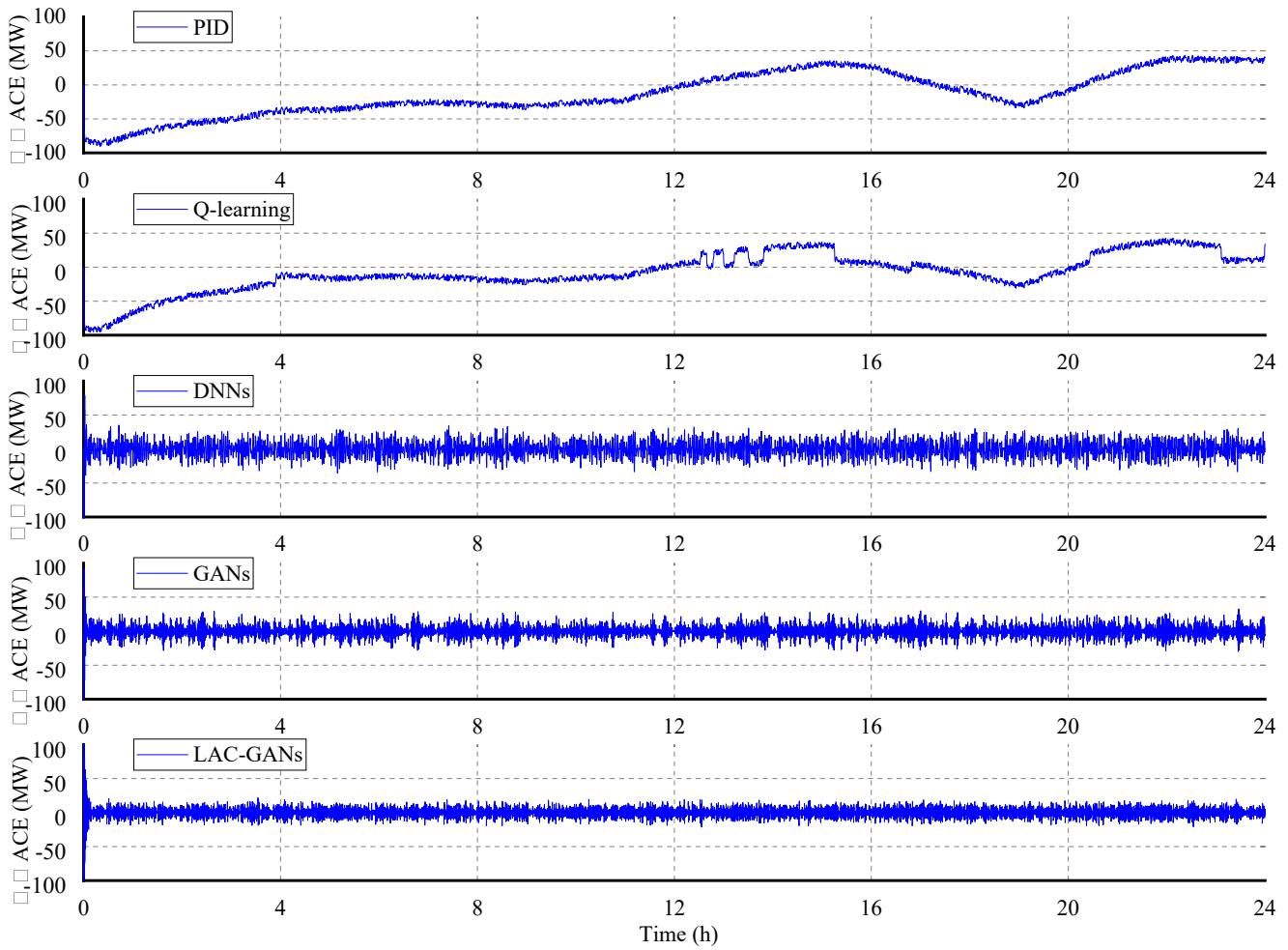


Fig. 12. ACEs achieved by different algorithms (Case I).

Table 7
Evaluation indices achieved by different algorithms in Case I.

Algorithm	$ \Delta f $ (Hz)	$ e_{ACE} $ (MW)	ISE	IAE	ITAE ($\times 10^7$)	C_{total} ($\times 10^6$ \$)
PID	0.0077	46.6083	6.3870	669.4009	5.7836	3.0908
Q learning	0.0061	34.7995	4.4102	527.1061	4.5542	2.7127
DNNs	0.0051	29.6653	3.5989	439.4683	3.7944	2.3738
GANs	0.0045	26.6373	3.7717	399.5905	3.3349	1.9004
LAC-GANs	0.0039	22.8337	2.9693	338.5054	2.7607	1.6304

(continued)

Algorithm 1. Pseudo code of LAC-GANs.

```

9: Online train of LAC-GANs by the multi-path lightweight method
10: for  $k = 1 \dots K$  do
11:   Acquire the real-time system state
12:   Predict the action  $a_t$  and act  $a_t$ 
13:   Acquire the next state  $s_{t+1}$  and immediate reward value  $r_t$ 
14:   Store real-time system state by the experience replay mechanism
15:   Prune the actor and critic by intergroup sparsity constraint with Eqs. (8)-(13)
16:   Update the actor by Eqs. (14)-(21)
17:   Update the critic by Eqs. (22)-(30)
18:   Update the discriminator by Eqs. (5)-(7)
19: End for

```

3. Smart generation control based on lightweight actor-critic generative adversarial networks

The LAC-GANs are applied to verify the feasibility of the LAC-GANs in real-life microgrids. Meanwhile, certain evaluation metrics are

presented to assess the efficacy of the LAC-GANs.

3.1. Smart generation control

The SGC based on the LAC-GANs is shown in Fig. 4. At each control period, the system stores the ACE, frequency deviation Δf , and long-term history data of each area by the experience replay mechanism in real-time. The reward value r_t is calculated; the current system state s_t (i.e., ACE and Δf) are utilized as the inputs of LAC-GANs. The LAC-GANs track real-time load disturbances and generate the optimal control command. Then, the control command is distributed to all generation units. The control commands are continuous prediction actions that vary with system states. The generator units are utilized to control the large-scale load disturbances, such as thermal and hydroelectric power units. Meanwhile, wind and solar generation units are integrated into the SGC system. The energy storage units can assist the generator units as an auxiliary frequency regulation means.

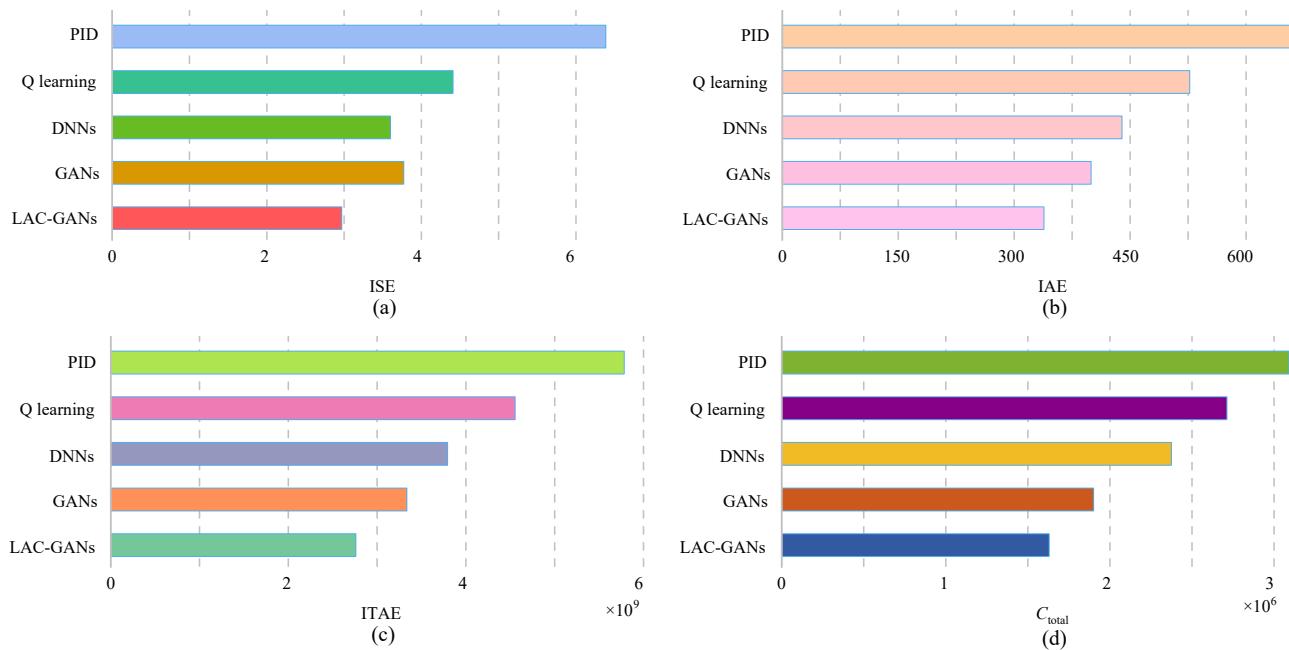


Fig. 13. Bar diagram obtained by evaluation indices in Area A (Case I): (a) ISE; (b) IAE; (c) ITAE; (d) C_{total} .

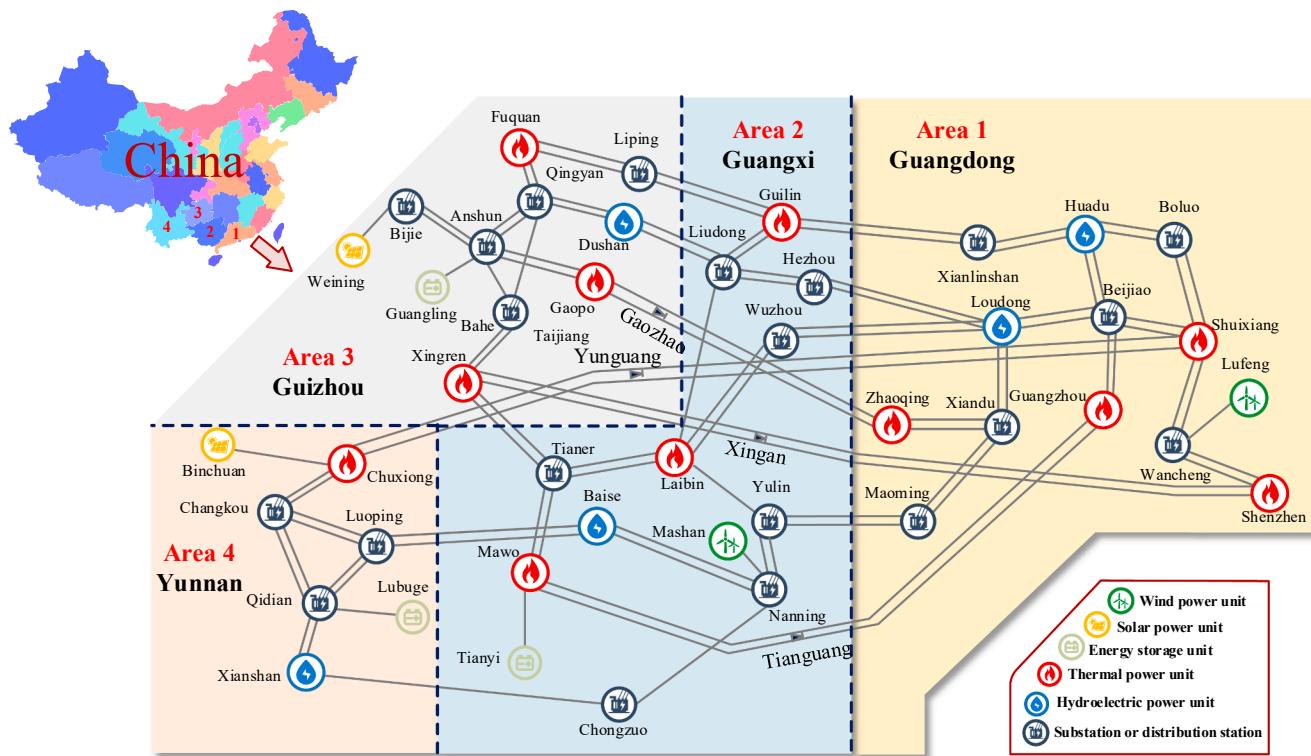


Fig. 14. China Southern Power Grid.

The main features of the SGC based on LAC-GANs are: (i) the LAC-GANs are applied to each area, which can coordinate control the generator units and energy storage units of multi-area; (ii) the control performance and generation cost are considered simultaneously, which can achieve the comprehensive optimization of dynamic performance and economic efficiency; (iii) various intelligent algorithms are suitable for the SGC and can reduce active power fluctuations.

3.2. Evaluation and reward functions

This paper adopts two evaluation indices to calibrate the model error of GANs: the root mean square error (RMSE) and mean absolute percentage error (MAPE). The LAC-GANs aim to minimize Δf at lower generation cost; thus, typical control performance index, ACE, and economic index are utilized as the evaluation criteria. Meanwhile, the reward function of the LAC-GANs is constructed with ACE and the economic costs.

Table 8

System parameters of Case II.

Area	Unit	ΔP^{\min} (MW)	ΔP^{\max} (MW)
Area 1-Guangzhou	Thermal power unit	2000	400
Area 1-Shenzhen	Thermal power unit	1500	100
Area 1-Zhangqing	Thermal power unit	800	200
Area 1-Shuixiang	Thermal power unit	735	200
Area 1-Loudong	Hydroelectric power unit	650	50
Area 1-Huadu	Hydroelectric power unit	650	50
Area 1-Lufeng	Wind power unit	300	20
Area 2-Guilin	Thermal power unit	1000	100
Area 2-Laibin	Thermal power unit	550	50
Area 2-Mawo	Thermal power unit	500	50
Area 2-Baise	Hydroelectric power unit	650	20
Area 2-Mashan	Wind power unit	300	20
Area 3-Fuquan	Thermal power unit	850	100
Area 3-Gaopo	Thermal power unit	550	50
Area 3-Xingren	Thermal power unit	500	50
Area 3-Dushan	Hydroelectric power unit	650	20
Area 3-Weining	Solar power unit	300	20
Area 3-Guangning	Energy storage unit	250	20
Area 4-Chuxiong	Thermal power unit	850	50
Area 4-Xishan	Hydroelectric power unit	500	20
Area 4-Binchuan	Solar power unit	300	20
Area 4-Lubuge	Energy storage unit	250	20

Note: ΔP^{\min} and ΔP^{\max} mean minimum and maximum power capacities, respectively.

Table 9

Parameters of comparison algorithm in Case II.

Algorithm	Parameter	Value
PID	Proportional	-2432
	Integral	-33
	Derivative	0.00001
Q learning	Learning rate	0.1
	Probability coefficient	0.5
	Reward coefficient	0.9
DNNs	Actions set	{-50, -40, -25, -10, -2.5, 0, 2.5, 10, 25, 40, 50}
	Network structure	[10, 100, 100, 4]
GANs	Learning rate	0.001
	Generator	[10, 100, 100, 4]
	Discriminator	[10, 100, 100, 4]
	Learning rate	0.01

(a) Performance indices of GANs

Both RMSE and MAPE are applied to reflect the prediction error of GANs, as

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{n=1}^N [y^{(n)} - G_{N\theta}(x^{(n)})]^2} \quad (37)$$

$$\text{MAPE} = \frac{100\%}{N} \sum_{n=1}^N \left| \frac{y^{(n)} - G_{N\theta}(x^{(n)})}{y^{(n)}} \right| \quad (38)$$

(b) Performance indices of LAC-GANs

(1) The typical control performance indices contain integral squared error (ISE), integral absolute error (IAE), and integral time multiple absolute error (ITAE). The three indices are applied to evaluate the cumulative error of Δf . The lower ISE, IAE, and ITAE mean the greater performance of the LAC-GANs.

$$e_{\text{ISE}} = \int_0^{\infty} \Delta f^2(t) dt \quad (39)$$

$$e_{\text{IAE}} = \int_0^{\infty} |\Delta f(t)| dt \quad (40)$$

$$e_{\text{ITAE}} = \int_0^{\infty} t |\Delta f(t)| dt \quad (41)$$

(2) The ACE is calculated as

$$e_{\text{ACE}} = B \Delta f + \Delta P_{\text{ex}} \quad (42)$$

where B is frequency deviation factor; ΔP_{ex} is tie-line exchange power deviation.

(3) The economic index is the total generation cost, which is the most popular index for reflecting economic efficiency, as

$$C_{\text{total}} = \sum_{w=1}^W (v_a \Delta P_w^2 + v_b \Delta P_w + v_c) \\ \text{s.t. } \begin{cases} \Delta P_w^{\min} \leq \Delta P_w \leq \Delta P_w^{\max} \\ P_{w,t} - P_{w,t-1} \leq P_w^{\text{up}} \\ P_{w,t-1} - P_{w,t} \leq P_w^{\text{down}} \end{cases} \quad (43)$$

where C_{total} is the total generation cost; W means the number of units; ΔP_w and $P_{w,t}$ are the w -th unit output power; ΔP_w^{\min} and ΔP_w^{\max} mean minimum and maximum power capacities of the w -th unit, respectively; v_a , v_b , and v_c are w -th unit economic coefficients; P_w^{up} and P_w^{down} mean power limitations of the w -th unit between two control periods.

(c) Reward function of LAC-GANs

The linear weighted square sum of the ACE and the C_{total} is utilized as the reward function. Besides, the C_{total} can directly reflect the economic efficiency of systems. However, the value of the C_{total} is usually much larger than the ACE. The values of the total generation cost are scaled to classify the ACE and the total generation cost in the same order of magnitude. The reward function R is expressed as

$$R = -c(e_{\text{ACE}})^2 - (1-c)(\kappa C_{\text{total}})^2 \quad (44)$$

where c and $(1-c)$ are the weights of e_{ACE} and C_{total} , respectively; c is set to be 0.5 in this paper; κ is set as 1/20000 to scale the value of C_{total} .

3.3. Parameter selection

The main parameters are set by theoretical analysis and manual trial and error. The key parameters are described as follows.

(a) Parameters of the EEMD.

The N_{std} and H are two main parameters of the EEMD. The N_{std} depends on the noise interference size in the original data and is generally set as the range [0.01, 0.1]. For the complex power data, the N_{std} is set to be 0.01 in this paper. The H is the repeat times of EMD and is usually chosen as [50, 100]. After numerous tests, the H is 50 in this paper.

(b) Hyperparameters of the GANs in pre-training.

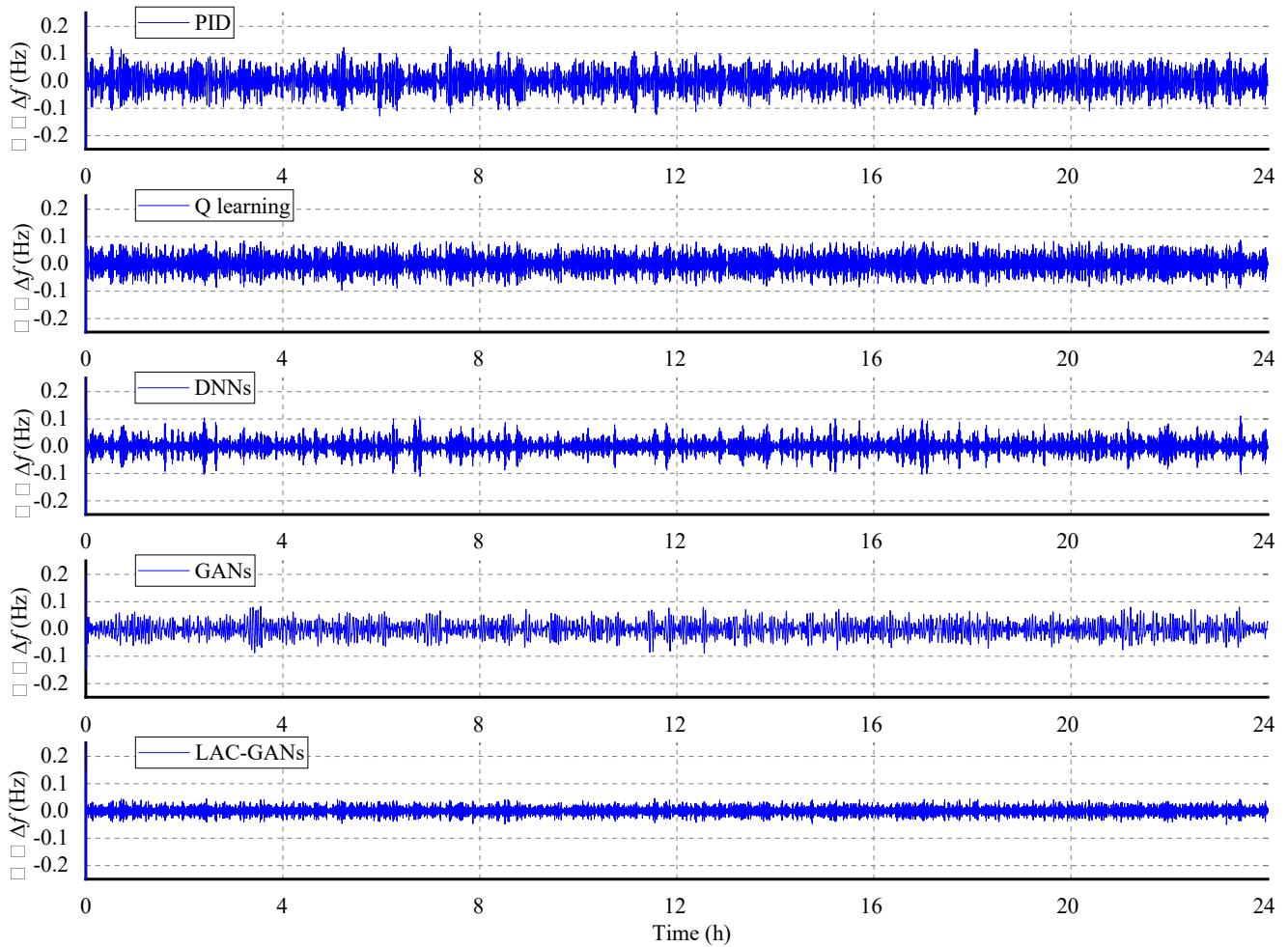


Fig. 15. Frequency deviations obtained by different algorithms in Area 1 (Case II).

The evolutionary strategy is applied to train the GANs for improving training stability. However, the inappropriate choice of hyper-parameters could cause the imbalance of the GANs training. The activation functions in the discriminator input and output layers are the Leaky ReLU and the Sigmoid functions, respectively. The structure parameters of the GANs are shown in Table 1.

The generator and discriminator are four-layer fully connected neural networks with dimensions of 10-100-100-4 and 10-100-100-2, respectively. In addition, the batch normalization (BN) layers are added to mitigate bad initialization impacts and keep the gradient propagation to each layer.

The learning rate of generator and discriminator are configured as 0.001 and 0.004 for the same updating rate, respectively.

(c) Parameters of the LAC-GANs in online training

After the pre-trained, the generator of GANs is reused as the actor. The parameters of LAC-GANs require reasonable settings in the SGC, as shown in Table 2.

(1) The structure of the critic

The critic is a fully connected neural network with dimensions of 10-100-100-4. The BN layer is added to each hidden layer. The critic output active function is linear. The active functions of the rest layers are ReLU. The learning rate is specified to 0.001 based on feedback from the training. Table 1 displays the structural parameters of the critic.

(2) The learning rate α_a of the actor

The α_a represents the updated rate of parameters, is the same as α_c . Therefore, the parameter α_a and α_c are the same and are chosen as 0.001.

(3) The discount factor γ

When γ tends to 1, the long-term reward can be obtained; when γ tends to 0, only the current reward is obtained. The γ of LAC-GANs is set as 0.94 to pursue long-term reward in this paper.

(4) The regular coefficient λ and the adjustment coefficient μ of the group MCP regularization

The λ and μ are utilized to adjust the penalty strength and control the penalty range, respectively. The λ and μ are set to be 0.8 and 0.5, respectively.

(5) The weight threshold β of the actor and critic

After removing the sub-path with smaller weights, the model size is reduced without impacting model performance. Through numerous trials and iterations, the weight threshold β is set to be 40% of the total weight.

4. Cases studies

The programming of case studies is emulated on MATLAB R2020b on

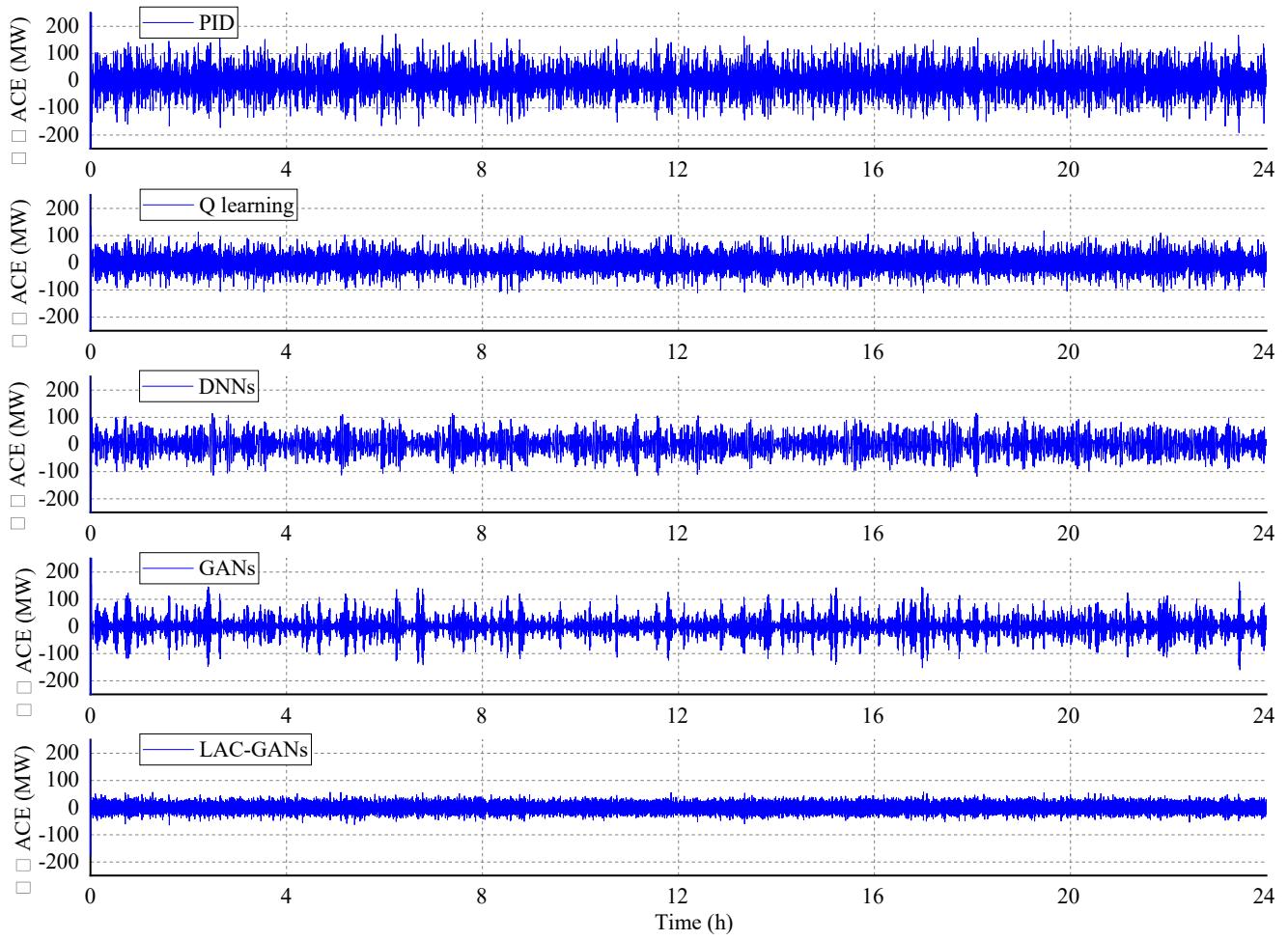


Fig. 16. ACEs obtained by different algorithms in Area 1 (Case II).

Table 10
Evaluation indices obtained by different algorithms in Case II.

Area	Algorithm	$ \Delta f $ (Hz)	$ e_{ACE} $ (MW)	ISE	IAE	ITAE ($\times 10^7$)	C_{total} ($\times 10^9$ \$)
Area 1	PID	0.0645	97.6247	209.14	3329.0762	14.4221	7.1236
	Q learning	0.0572	79.2546	102.55	2224.7201	9.4202	6.8246
	DNNs	0.0481	68.2454	113.19	2447.2015	10.4032	7.3217
	GANs	0.0422	59.2456	87.47	2048.9715	9.5763	6.3514
	LAC-GANs	0.0355	48.3599	55.36	1739.3321	7.5045	5.9425
Area 2	PID	0.0092	87.5421	8.1246	542.3214	4.6241	5.2471
	Q learning	0.0103	93.3452	7.9345	575.2145	4.1201	5.3578
	DNNs	0.0076	77.2456	7.5524	532.3571	3.2477	4.6987
	GANs	0.0082	69.1248	6.1237	489.1245	3.1147	4.9631
	LAC-GANs	0.0075	88.1247	6.0145	470.2546	3.2374	4.5512
Area 3	PID	0.0086	85.3147	7.3145	510.6541	4.3217	4.9524
	Q learning	0.0074	81.6316	6.2587	489.2484	4.1245	4.7836
	DNNs	0.0071	79.2534	6.0124	452.1277	3.9452	4.2356
	GANs	0.0069	73.4311	6.9247	432.4245	3.7652	4.0247
	LAC-GANs	0.0066	69.1275	6.1236	386.1782	3.4321	4.8654
Area 4	PID	0.0088	79.2473	6.4214	470.4451	3.9624	4.3244
	Q learning	0.0073	72.7354	6.1027	438.2756	3.7528	4.1245
	DNNs	0.0071	70.1453	5.9021	421.6589	3.5246	3.9756
	GANs	0.0058	63.3475	5.3245	401.2347	3.1245	2.7564
	LAC-GANs	0.0059	62.2145	5.8015	378.4376	2.9547	2.5786

a laptop with i5-10210U CPU and 32 GB RAM. The proposed algorithm needs to be tested on benchmark conditions and the real-life system [50]. Therefore, the proposed LAC-GANs are applied in the IEEE two-area system (Case I) [51] and China Southern Power Grid (Case II) [52]. Meanwhile, two classical control algorithms (i.e., PID and Q learning) and two intelligent control algorithms (i.e., DNNs and GANs)

are compared with the LAC-GANs under two cases. The parameters of four comparison algorithms are optimized by the genetic algorithm [53] with population size 50 and iteration number 100. In pre-training, the training data of all algorithms are the history state s_t and history action a_t obtained from Case I and Case II within 30 days. In online training, the training data of all algorithms are the real-time state of Case I and Case

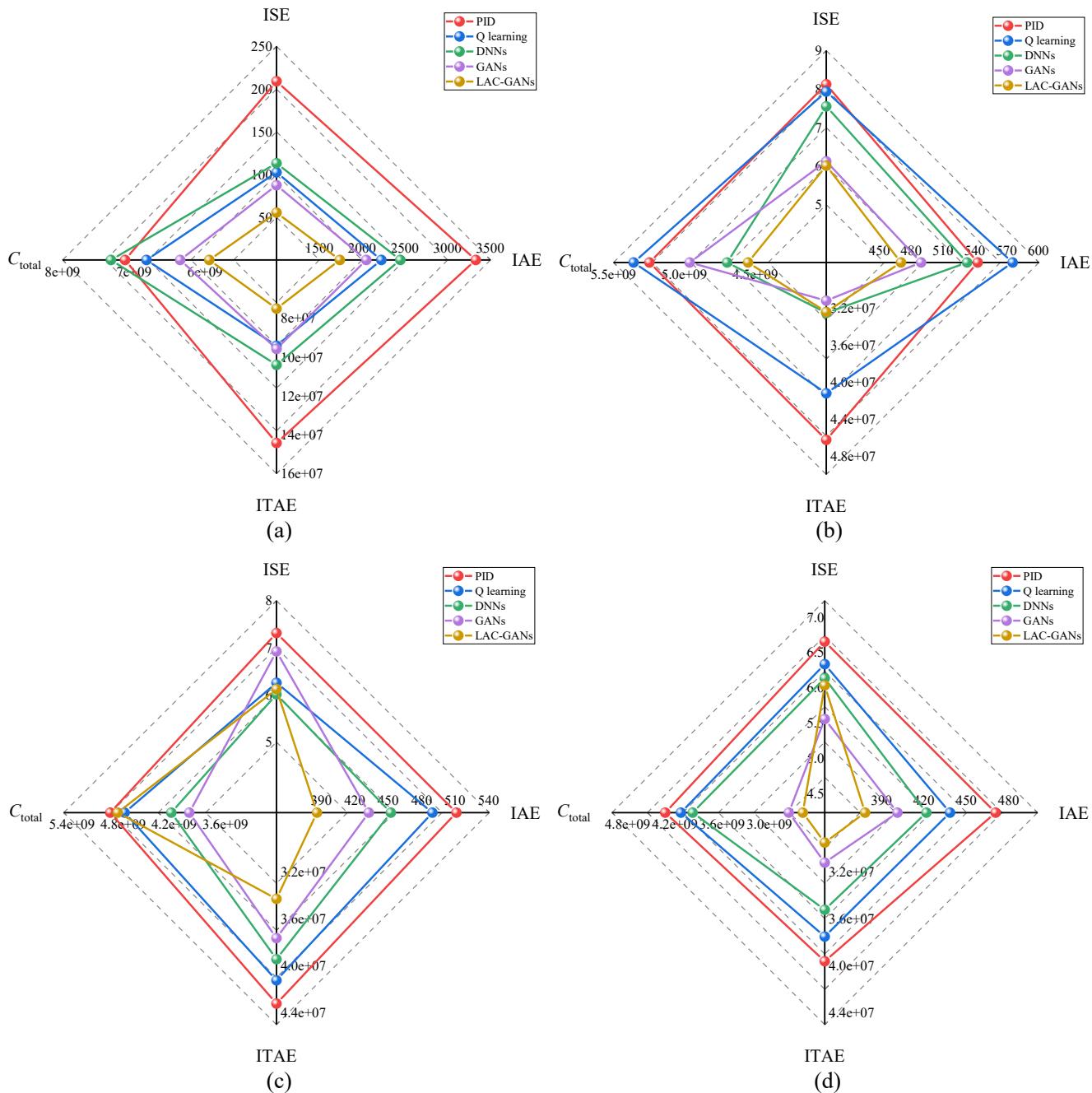


Fig. 17. Evaluation indices in different areas: (a) Area 1; (b) Area 2; (c) Area 3; (d) Area 4.

Table 11
RMSE and MAPE obtained by general EEMD parameters.

N_{std}	H	Mean RMSE (kW)	Mean MAPE (%)	Max RMSE (kW)	Max MAPE (%)
0.01	50	18.2724	1.3681	18.5326	1.3724
0.01	80	19.3475	1.5568	19.5468	1.5849
0.01	100	22.3645	1.7564	22.9524	1.8026
0.05	50	19.8752	1.5236	20.1437	1.5347
0.05	80	19.3578	1.5720	19.6327	1.5925
0.05	100	18.3684	1.3214	18.8657	1.3620
0.1	50	19.6758	1.5436	19.9782	1.5683
0.1	80	22.9681	1.8779	23.2186	1.9035
0.1	100	21.4379	1.8136	21.7825	1.8357

Table 12
Performance comparison under different weight thresholds.

Weight threshold	$ \Delta f $ (Hz)	$C_{total} (\times 10^9 \$)$	Training time (s)	Computation time (s)	Computation memory (kB)
0	0.0367	6.0153	31247.4824	0.0975	821
10%	0.0371	6.1258	30214.2156	0.0802	738
20%	0.0382	6.1434	28924.2387	0.0770	656
30%	0.0389	6.1569	27648.6459	0.0623	574
40%	0.0395	6.1657	21007.3410	0.0469	492
50%	0.0492	6.9814	19207.3145	0.3256	410
60%	0.0575	7.1045	17014.2187	0.2817	328
70%	0.1196	11.2741	13047.2145	0.2647	246
80%	0.2135	16.4025	11724.2146	0.2386	164
90%	0.4845	22.0128	9234.0154	0.1835	82

II.

4.1. IEEE two-area system (Case I)

Case I (Fig. 5) includes the generator units and multiple distributed energies (i.e., wind, solar, and energy storage units). The generator units (e.g., thermal and hydroelectric power units) can be represented by a governor, a turbine, and a power limitation [54]. The energy storage unit includes a power conversion system, a time-delay conversion, a mathematical energy storage model, and a power limitation [55]. Fig. 6 shows the power grid sketch of Case I. Table 3 lists the system parameters of Case I. The parameters of the comparison algorithms in Case I are given in Table 4.

(a) Pre-training of GANs

The GANs training data are the history state s_t and history action a_t in Case I, are not required to be labeled. First, the training data are decomposed into relatively simple and regular components by the EEMD. In addition, the evolutionary strategy is applied in the training of the GANs. Finally, the GANs with evolutionary strategies are compared to the traditional GANs; the network structure of the two GANs is the same.

Compared to the traditional GANs, the GANs with evolutionary strategy can obtain a higher starting fitness score with fewer training steps and display considerable convergence stability (Fig. 7(a)). In addition, the targets selected in each evolutionary step are recorded during training (Fig. 7(b)). The MSE and MAE variations are more easily selected at the beginning training process than the Huber and Log-Cosh variations. As the generator approaches 2×10^4 steps, more the Huber and Log-Cosh variations are selected. For the GANs with evolutionary strategy, the different variation loss mitigates the limitations of individual adversarial training objectives and improves convergence stability. From the iteration curves (Fig. 7(a)) and error comparison (Table 5), the EEMD can effectively improve the training speed and accuracy of GANs with two updating strategies.

(b) Online training of LAC-GANs

After being pre-trained, the final surviving offspring (i.e., optimal generator) is selected to participate in online training of LAC-GANs. A continuous sinusoidal load is added into the system of Case I for dynamic learning and training exploration of the LAC-GANs with optimal decisions. The sinusoidal load is set as Fig. 7. Meanwhile, to simulate the randomness of the environment, the Gaussian white noise with 40 dB is added to the sinusoidal load (Fig. 8).

From the results in Fig. 9(a), the PID and Q-learning cannot follow load perturbations steadily in a short time. In contrast, the DNNs and GANs can predict the next systemic action by pre-training, can effectively reduce frequency fluctuations. Compared to DNNs and GANs, the LAC-GANs can stably follow the load perturbations after about 1000 s by dynamic learning and training exploration. From Fig. 9(b), the frequency deviation obtained by LAC-GANs has smaller dispersion and randomness. The calculation time is the time gap between the controller receiving the message and issuing the control command. The control period of microgrids is typically 4 s to 8 s. The computation time and memory of LAC-GANs are less than the Q learning, DNNs, and GANs, which validated that the multi-path lightweight method effectively improves the training efficiency of LAC-GANs (Table 6). The above results show that the LAC-GANs possess strong learning ability and dynamic control performance.

(c) Online operation with real-life resident load

The real-life resident load in Fig. 10(a) is added to Case I for evaluating algorithm robustness and adaptability. Since the power output of

wind and solar power units depends on uncontrollable weather, their power outputs are considered as random disturbances (Fig. 10(b)). The simulation time is set to 86400 s or 24 h.

From the curves of frequency deviation (Fig. 11) and ACE (Fig. 12), the curve obtained by the LAC-GANs is smoother with lower amplitude than that of the comparison algorithms; the LAC-GANs can provide more precise and stable control commands. Compared to the comparison algorithms in Area A of Table 7 and Fig. 13, the average absolute $|\Delta f|$ and the average absolute ACE $|e_{ACE}|$ obtained by the LAC-GANs are at least 13.333% and 14.279% smaller, respectively. Meanwhile, the ISE, IAE, ITAE, and C_{total} obtained by LAC-GANs are at least 21.274%, 15.286%, 17.2179%, and 14.2075% smaller than the comparison algorithms, respectively. Thus, the LAC-GANs have higher control performance with lower economic consumption.

Compared to the PID and Q learning, the LAC-GANs have learned enough history prior knowledge in the random environment by the pre-training and the online training. Therefore, the LAC-GANs can provide more accurate control commands in online operation with real-life resident loads. Compared to the DNNs and GANs, the LAC-GANs have greater dynamic learning ability by combining with the actor-critic strategy. Thus, the LAC-GANs have higher control performance by continuous learning.

The above analysis in Area A (Case I) show that: (i) the LAC-GANs possess strong robustness and can maintain great control performance in the real-life resident load; (ii) the LAC-GANs can achieve better control and optimization performances.

4.2. China Southern power grid (Case II)

Case II is a large capacity grid with the four southern provinces of China, including four areas: Guangdong, Guangxi, Yunnan, and Guizhou (Fig. 14). Case II is applied to verify the stability and performance of the LAC-GANs in a complex real-life environment. The LAC-GANs are added to four areas to regulate the system frequency. Table 8 lists the system parameters of Case II. The parameters of the comparison algorithms in Case II are given in Table 9.

From the curves of frequency deviations (Fig. 15) and ACE (Fig. 16), the amplitudes of frequency deviation and ACE of LAC-GANs are maintained at -0.05 Hz to 0.05 Hz and -50 MW to 50 MW, respectively. The LAC-GANs exhibit strong stability and convergence than the other four comparison algorithms. After the training process of the LAC-GANs, the LAC-GANs have stable control performance in a random environment. Therefore, the LAC-GANs still have great control performance in complex multi-area environments. In addition, the LAC-GANs continuously dynamically learn from the online operation. Thus, most evaluation indices of the LAC-GANs are overall superior to the other four comparison algorithms (Table 10 and Fig. 17). For an example of Area 1, The $|\Delta f|$ and $|e_{ACE}|$ acquired by the LAC-GANs are at least 15.8767% and 18.3738% lower than the comparison algorithms, respectively. Furthermore, the ISE is at least 36.7097% lower, the IAE is at least 15.1119% lower, the ITAE is at least 21.6346% lower, and the C_{total} is at least 6.4379% lower than the four comparison algorithms.

The above analyses show that: (i) the LAC-GANs can coordinate control multiple distributed energies with great adaptability and robustness in complex multi-area environments; (ii) the LAC-GANs have greater online learning and dynamic optimization capabilities in online operations.

4.3. Discussion

Case study results show that the LAC-GANs have great online learning ability and dynamic control performance. Meanwhile, by decomposing complex power data, the EEMD can effectively improve the training speed of GANs; the GANs with evolutionary strategy mitigate the limitations of individual adversarial training objectives; thus,

the training stability of the GANs is improved. In addition, the multi-path lightweight method can effectively prune the sub-path with small parameter values of the LAC-GANs. Finally, the LAC-GANs can effectively regulate the frequency with low generation costs in a complex real-life environment.

The parameters of the EEMD and the weight threshold of the LAC-GANs determine the control performance of the LAC-GANs. Therefore, this paper further discusses these two parameters.

(a) Parameters of the EEMD

The proper N_{std} and H are beneficial for data decomposition with the smaller model error of the GANs. Generally, 50, 80, and 100 are the common values of H ; 0.01, 0.05, and 0.1 are the common values of N_{std} . Different EEMD parameters are set for comparison (Table 11). The dataset is the historical power data obtained from Case II. As shown in Table 10, when $N_{\text{std}} = 0.01$, $H = 50$, the mean and maximum values of the RMSE and MAPE are the smallest among all the comparison parameters. Therefore, the N_{std} and H are set to 0.01 and 50, respectively.

(b) Weight threshold of the LAC-GANs

The purpose of pruning is to alleviate the problem of parameters redundancy and reduce the consumption of time and storage resources. The weight threshold directly determines the pruning impact. Different weight thresholds are set for comparison in Area 1 of Case II (Table 12). As shown in Table 12, when the weight threshold is smaller than 40% of total weight, the Δf and C_{total} increase slowly. When the weight threshold is greater than 40% of total weight, the Δf and C_{total} increase rapidly. The weight threshold is set to 40% of the total weight to consider both the control performance and pruning impact of the LAC-GANs.

5. Conclusions

This paper proposes lightweight actor-critic generative adversarial networks for the smart generation control. After the training process of lightweight actor-critic generative adversarial networks, this method has great adaptive and online optimization ability in two complex cases. The following are the main characteristics of the lightweight actor-critic generative adversarial networks.

- (1) The lightweight actor-critic generative adversarial networks introduce the ensemble empirical mode decomposition and the evolutionary strategy. The ensemble empirical mode decomposition can decompose the original complex power data into more regular and simpler sub-data to improve the training speed of generative adversarial networks. The evolutionary strategy can optimize generative adversarial networks by different loss functions to mitigate the limitations of individual adversarial training objectives.
- (2) The lightweight actor-critic generative adversarial networks combine the generative adversarial networks and the actor-critic strategy. As a result, the lightweight actor-critic generative adversarial networks can interact with the environment in real-time by the actor-critic strategy, have adaptive and optimization-seeking abilities. In addition, the training data of lightweight actor-critic generative adversarial networks does not require to be labeled, which reduces the labor and time resources for labeling.
- (3) The multi-path lightweight method based on the nonconvex group minimax concave penalized regularization can prune the lightweight actor-critic generative adversarial networks for reducing the calculation time and storage resources. Furthermore, the difference of convex decompositions can convert the nonconvex function into a convex function for solving the

difficulty of optimizing the nonconvex group minimax concave penalized regularization.

- (4) The smart generation control based on the proposed method can coordinate multiple distributed energies. By analyzing the results of Case 1 and Case 2, the lightweight actor-critic generative adversarial networks can achieve excellent control performance and economic efficiency in complex multi-area environments.

In future works, the lightweight actor-critic generative adversarial networks could collaborate with more distributed energy for mitigating the pressure on generating units. In addition, the lightweight actor-critic generative adversarial networks could improve the structure to increase control performance. Moreover, to manage the microgrid with dynamic topology, extendable width learning could be applied to the lightweight actor-critic generative adversarial networks. Finally, to ensure the lightweight actor-critic generative adversarial networks can be applied to the daily operation of the power grid, the method could be repeatedly tested several times in different real-life systems.

CRediT authorship contribution statement

Kunlun Han: Investigation, Supervision, Software, Validation. **Kai Yang:** Data curation, Formal analysis, Investigation, Software, Validation, Visualization, Writing – original draft. **Linfei Yin:** Conceptualization, Funding acquisition, Project administration, Methodology, Resources, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant. 52107081 and the National Natural Science Foundation of Guangxi Province under Grant. AD19245001 and 2020GXNSFBA159025.

References

- [1] Wen J, Zhao D, Zhang C. An overview of electricity powered vehicles: Lithium-ion battery energy storage density and energy conversion efficiency. *Renew Energy* 2020;162:1629–48.
- [2] Matamala Y, Feijoo F. A two-stage stochastic Stackelberg model for microgrid operation with chance constraints for renewable energy generation uncertainty. *Appl Energy* 2021;303:117608. <https://doi.org/10.1016/j.apenergy.2021.117608>.
- [3] Makolo P, Zamora R, Lie T-T. The role of inertia for grid flexibility under high penetration of variable renewables-A review of challenges and solutions. *Renew Sustain Energy Rev* 2021;147:111223. <https://doi.org/10.1016/j.rser.2021.111223>.
- [4] Ding Y, Mao M, Chang L. Conservative power theory and its applications in modern smart grid: Review and prospect. *Appl Energy* 2021;303:117617. <https://doi.org/10.1016/j.apenergy.2021.117617>.
- [5] Xi L, Wu J, Xu Y, Sun H. Automatic Generation Control Based on Multiple Neural Networks With Actor-Critic Strategy. *IEEE Trans Neural Networks Learn Syst* 2021;32(6):2483–93.
- [6] Xi L, Chen J, Huang Y, Xu Y, Liu L, Zhou Y, et al. Smart generation control based on multi-agent reinforcement learning with the idea of the time tunnel. *Energy* 2018;153:977–87.
- [7] Çetin G, Özkaraca O, Keçebaş A. Development of PID based control strategy in maximum exergy efficiency of a geothermal power plant. *Renew Sustain Energy Rev* 2021;137:110623. <https://doi.org/10.1016/j.rser.2020.110623>.
- [8] Hui H, Ding Yi, Lin Z, Siano P, Song Y. Capacity allocation and optimal control of inverter air conditioners considering area control error in multi-area power systems. *IEEE Trans Power Syst* 2020;35(1):332–45.
- [9] Zhang ZJ, Yue D, Dou CX, Zhang HF. Multiagent system-based integrated design of security control and economic dispatch for interconnected microgrid systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 2021;51:2101–12.
- [10] Li H, Misra S. Reinforcement learning based automated history matching for improved hydrocarbon production forecast. *Appl Energy* 2021;284:116311. <https://doi.org/10.1016/j.apenergy.2020.116311>.

- [11] Dabbaghjamanesh M, Moeini A, Kavousi-Fard A. Reinforcement learning-based load forecasting of electric vehicle charging station using Q-learning technique. *IEEE Trans Ind Inf* 2021;17(6):4229–37.
- [12] Perera ATD, Kamalaruban P. Applications of reinforcement learning in energy systems. *Renew Sustain Energy Rev* 2021;137:110618. <https://doi.org/10.1016/j.rser.2020.110618>.
- [13] Kheshti M, Ding L, Bao W, Yin M, Wu Q, Terzija V. Toward intelligent inertial frequency participation of wind farms for the grid frequency control. *IEEE Trans Ind Inf* 2020;16(11):6772–86.
- [14] Li J, Yu T, Zhang X. Coordinated load frequency control of multi-area integrated energy system using multi-agent deep reinforcement learning. *Appl Energy* 2022;306:117900. <https://doi.org/10.1016/j.apenergy.2021.117900>.
- [15] Nagamani G, Adhira B, Soundararajan G. Non-fragile extended dissipative state estimation for delayed discrete-time neural networks: Application to quadruple tank process model. *Nonlinear Dyn* 2021;104(1):451–66.
- [16] Yin L, Luo S, Ma C. Expandable depth and width adaptive dynamic programming for economic smart generation control of smart grids. *Energy* 2021;232:120964. <https://doi.org/10.1016/j.energy.2021.120964>.
- [17] Wang J, Zou B, Liu M, Li Y, Ding H, Xue K. Milling force prediction model based on transfer learning and neural network. *J Intell Manuf* 2021;32(4):947–56.
- [18] Vlachas PR, Pathak J, Hunt BR, Sapsis TP, Girvan M, Ott E, et al. Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics. *Neural Networks* 2020;126:191–217.
- [19] Carvalho J, Plastino A. On the evaluation and combination of state-of-the-art features in Twitter sentiment analysis. *Artif Intell Rev* 2021;54(3):1887–936.
- [20] Yinka-Banjo C, Ugot O-A. A review of generative adversarial networks and its application in cybersecurity. *Artif Intell Rev* 2020;53(3):1721–36.
- [21] Dong W, Chen X, Yang Q. Data-driven scenario generation of renewable energy production based on controllable generative adversarial networks with interpretability. *Appl Energy* 2022;308:118387. <https://doi.org/10.1016/j.apenergy.2021.118387>.
- [22] Mao X, Li Q, Xie H, Lau RYK, Wang Z, Smolley SP. On the effectiveness of least squares generative adversarial networks. *IEEE Trans Pattern Anal Mach Intell* 2019;41(12):2947–60.
- [23] Han W, Wang L, Feng R, Gao L, Chen X, Deng Ze, et al. Sample generation based on a supervised Wasserstein generative adversarial network for high-resolution remote-sensing scene classification. *Inf Sci* 2020;539:177–94.
- [24] Lv W, Xiong J, Shi J, Huang Y, Qin S. A deep convolution generative adversarial networks based fuzzing framework for industry control protocols. *J Intell Manuf* 2021;32(2):441–57.
- [25] Tang C, He Z, Li Y, Lv JC. Zero-shot learning via structure-aligned generative adversarial network. *IEEE Trans Neural Networks Learn Syst* 2021;109:1–14.
- [26] Wang F, Zhang Z, Liu C, Yu Y, Pang S, Duić N, et al. Generative adversarial networks and convolutional neural networks based weather classification model for day ahead short-term photovoltaic power forecasting. *Energy Convers Manage* 2019;181:443–62.
- [27] Fan C, Ding C, Xiao L, Cheng F, Ai Z. Deep belief ensemble network based on MOEA/D for short-term load forecasting. *Nonlinear Dyn* 2021;105(3):2405–30.
- [28] Vidnerová P, Neruda R. Vulnerability of classifiers to evolutionary generated adversarial examples. *Neural Networks* 2020;127:168–81.
- [29] Wang C, Xu C, Yao X, Tao D. Evolutionary generative adversarial networks. *IEEE Trans Evol Comput* 2019;23(6):921–34.
- [30] Yin L, Zhang B. Time series generative adversarial network controller for long-term smart generation control of microgrids. *Appl Energy* 2021;281:116069. <https://doi.org/10.1016/j.apenergy.2020.116069>.
- [31] Liang J, Tang W. Sequence generative adversarial networks for wind power scenario generation. *IEEE J Sel Areas Commun* 2020;38(1):110–8.
- [32] Ye Y, Qiu D, Sun M, Papadaskalopoulos D, Strbac G. Deep reinforcement learning for strategic bidding in electricity markets. *IEEE Trans Smart Grid* 2020;11(2):1343–55.
- [33] Sharif A, Li JP, Saleem MA, Manogran G, Kadry S, Basit A, et al. A dynamic clustering technique based on deep reinforcement learning for Internet of vehicles. *J Intel Manuf* 2021;32(3):757–68.
- [34] Pfau D, Vinyals O. Connecting generative adversarial networks and actor-critic methods. *arXiv:1610.01945*, 2016.
- [35] Peng, B, Li X, Gao J, Liu J, Wong KF. Adversarial advantage actor-critic model for task-completion dialogue policy learning. *arXiv:1710.11277*, 2017.
- [36] Flet-Berliac Y, Ferret J, Pietquin O, Preux P, Geist M. Adversarially guided actor-critic. *arXiv:2102.04376*, 2021.
- [37] Sadoughi N, Busso C. Speech-driven expressive talking lips with conditional sequential generative adversarial networks. *IEEE Trans Affective Comput* 2021;12(4):1031–44.
- [38] Shi Y, Huang J, Jiao Y, Yang Q. A semi-smooth newton algorithm for high-dimensional nonconvex sparse learning. *IEEE Trans Neural Networks Learn Syst* 2020;31(8):2993–3006.
- [39] Wang J, Chang Q, Chang Q, Liu Y, Pal NR. Weight noise injection-based MLPs with group lasso penalty: asymptotic convergence and application to node pruning. *IEEE Trans Cybern* 2019;49(12):4346–64.
- [40] Li Z, Yang ZY, Zhao HL, Xie SL. Direct-optimization-based DC dictionary learning with the MCP regularizer. *IEEE Trans Neural Networks Learn Syst* 2021;65:1–12.
- [41] An Y, Xue H. Indefinite twin support vector machine with DC functions programming. *Pattern Recogn* 2022;121:108195. <https://doi.org/10.1016/j.patcog.2021.108195>.
- [42] Wang B, Wang J. Energy futures and spots prices forecasting by hybrid SW-GRU with EMD and error evaluation. *Energy Econ* 2020;90:104827. <https://doi.org/10.1016/j.eneco.2020.104827>.
- [43] Ding Y, Chen Z, Zhang H, Wang X, Guo Y. A short-term wind power prediction model based on CEEMD and WOA-KELM. *Renewable Energy* 2022;189:188–98.
- [44] Zhou F, Huang Z, Zhang C. Carbon price forecasting based on CEEMDAN and LSTM. *Appl Energy* 2022;311:118601. <https://doi.org/10.1016/j.apenergy.2022.118601>.
- [45] Zhao C, Yen GG, Sun Q, Zhang C, Tang Y. Masked GAN for unsupervised depth and pose prediction with scale consistency. *IEEE Trans Neural Networks Learn Syst* 2021;32(12):5392–403.
- [46] Tao Xi, Hafid AS. DeepSensing: a novel mobile crowdsensing framework with double deep Q-network and prioritized experience replay. *IEEE Internet Things J* 2020;7(12):11547–58.
- [47] Zhang B, Hu W, Cao Di, Li T, Zhang Z, Chen Z, et al. Soft actor-critic-based multi-objective optimized energy conversion and management strategy for integrated energy systems with renewable energy. *Energy Convers Manage* 2021;243:114381. <https://doi.org/10.1016/j.enconman.2021.114381>.
- [48] Blakeman S, Mareschal D. A complementary learning systems approach to temporal difference learning. *Neural Networks* 2020;122:218–30.
- [49] Li J, He H, Li L, Chen G. A novel generative model with bounded-GAN for reliability classification of gear safety. *IEEE Trans Ind Electron* 2019;66(11):8772–81.
- [50] Shi W, Li Na, Chu C-C, Gadh R. Real-time energy management in microgrids. *IEEE Trans Smart Grid* 2017;8(1):228–38.
- [51] Yin L, Wang T, Wang S, Zheng B. Interchange objective value method for distributed multi-objective optimization: Theory, application, implementation. *Appl Energy* 2019;239:1066–76.
- [52] Zhang J, Cheng C, Yu S, Wu H, Gao M. Sharing hydropower flexibility in interconnected power systems: A case study for the China Southern power grid. *Appl Energy* 2021;288:116645. <https://doi.org/10.1016/j.apenergy.2021.116645>.
- [53] Lei W, Gastro O, Wang Y, Felicien NH, Hui Li. Intelligent modelling to predict heat transfer coefficient of vacuum glass insulation based on thinking evolutionary neural network. *Artif Intell Rev* 2020;53(8):5907–28.
- [54] Xi L, Yu Lu, Xu Y, Wang S, Chen Xi. A novel multi-agent DDQN-AD method-based distributed strategy for automatic generation control of integrated energy systems. *IEEE Trans Sustainable Energy* 2020;11(4):2417–26.
- [55] Li S, Gu C, Zhao P, Cheng S. A novel hybrid propulsion system configuration and power distribution strategy for light electric aircraft. *Energy Convers Manage* 2021;238:114171. <https://doi.org/10.1016/j.enconman.2021.114171>.