A Flexibility Scheduling Method for Distribution Network Based on Robust Graph DRL Against State Adversarial Attacks

Ziyang Yin, *Student Member*, *IEEE*, Shouxiang Wang, *Senior Member*, *IEEE*, and Qianyu Zhao

***Abstract*—In the context of large-scale photovoltaic integration, flexibility scheduling is essential to ensure the secure and efficient operation of distribution networks (DNs). Recently, deep reinforcement learning (DRL) has been widely applied to scheduling problems. However, most methods neglect the vulnerability of DRL to state adversarial attacks such as load redistribution attacks, significantly undermining its security and reliability. To this end, a flexibility scheduling method is proposed based on robust graph DRL (RoGDRL). A flexibility gain improvement model considering temperature-dependent resistance is first proposed, which considers weather factors as additional variables to enhance the precision of flexibility analysis. Based on this, a state-adversarial two-player zero-sum Markov game (SA-TZMG) model is proposed, which converts the robust DRL scheduling problem into a Nash equilibrium problem. The proposed SA-TZMG model considers the physical constraints of state attacks that guarantee the maximal flexibility gain for the defender when confronted with the most sophisticated and stealthy attacker. A two-stage RoGDRL algorithm is proposed, which introduces the graph sample and aggregate (GraphSAGE) driven soft actor-critic to capture the complex feature about the neighbors of nodes and their properties via inductive learning, thereby solving the Nash equilibrium policies more efficiently. Simulations based on the modified IEEE 123-bus system demonstrates the efficacy of the proposed method.**

***Index Terms*—Distribution network, photovoltaic, flexibility scheduling, deep reinforcement learning, cyber attack.**

I. Introduction

OPERATIONAL flexibility denotes the capacity of the power system to maintain safe and efficient operation, which is critical for distribution networks (DNs) with high penetration rates of photovoltaic (PV) [1]. In DNs, operational flexibility fundamentally reflects the level of coordination and utilization of controllable resources within the system, where the essence of scheduling methods is precisely to enhance and apply flexibility [2]. Consequently, distribution system operators (DSOs) can improve operational flexibility by coordinating diverse controllable resources through optimized scheduling, an approach known as flexibility scheduling.

Recently, many research studies on flexibility scheduling have emerged [2]-[7]. In [2], a flexibility analysis framework is designed to fully exploit the controllability of various resources, thereby achieving the goals of improving operation costs, voltage distribution, and risk control through optimal scheduling. In [4], operational flexibility indices, encompassing node, system, and network transmission flexibility, are developed, offering a flexibility perspective for reinterpreting the scheduling problems of the DN. Based on this, [5] establishes a unified framework for quantifying and enhancing operational flexibility, aiming to achieve a feasible balance of the DSO's diverse flexibility demands, including reducing operation costs, improving voltage profiles, and alleviating branch congestion. The purpose of flexibility scheduling is to satisfy the DSO's comprehensive demands for the economical, safe, and clean operation of the DN by fully unleashing the regulation capabilities of controllable resources within the network [6]. Thus, it is necessary to integrate various controllable resources in the DN into a unified optimization framework.

The primary controllable resources for enhancing operational flexibility include energy storage systems (ESSs) [2], PV inverters [3], soft open points (SOPs) [5], and sectionalizing and tie switches [7]. The DSO requires a meticulously designed scheduling model to coordinate these discrete and continuous controllable resources to enhance operational flexibility while satisfying physical and operational constraints. However, most modeling methods for flexibility scheduling assume constant line resistance. Contrarily, numerous studies have demonstrated that line resistance is dynamically variable [8], [9]. Specifically, [9] formulates an optimal power flow that considers transmission line conductor temperatures to improve the accuracy of optimal power flow analysis. Thus, including temperature-dependent resistance is crucial for enhancing the precision of flexibility scheduling.

Flexibility scheduling is a typical mixed-integer nonlinear programming problem. The most common methods for solving such problems include heuristic algorithms and mathe-

Manuscript received: April 17, 2024; revised: August 13, 2024; accepted: October 19, 2024. Date of CrossCheck: October 19, 2024. Date of online publication: November 15, 2024.

This work was supported by the National Natural Science Foundation of China (No. 52077149), Key Program for National Natural Science Foundation of China Joint Funds (No. U2166202), and National Natural Science Foundation of China Young Scientist Fund (No. 52407134).

Z. Yin, S. Wang, and Q. Zhao (corresponding author) are with Key Laboratory of Smart Grid of Ministry of Education, Tianjin University, Tianjin 300072, China (e-mail: zyyin@tju.edu.cn; sxwang@tju.edu.cn; zhaoqianyu@tju.edu.cn).

DOI: 10.35833/MPCE.2024.000409

matical programming methods such as mixed-integer second-order cone programming (MISOCP) and linearized approximation programming (LAP). However, these methods face three primary challenges. ① The solution quality of heuristic algorithms cannot be guaranteed, and they typically require extensive computation [10]. ② Commercial solvers exhibit relatively low efficiency in solving MISOCP problems [11]. ③ While LAP offers higher computational efficiency, it necessitates the imposition of assumptions and simplifications to ensure solvability of the scheduling model. These may deviate the model from reality, thereby reducing the accuracy of the solution [12]. By contrast, the deep reinforcement learning (DRL) avoids the need for undue simplifications and assumptions in the DN model. It generates an optimal policy, i.e., decision-making rules for the optimization problem, rather than a singular optimal solution. Thus, the trained DRL algorithm can achieve real-time and online decision-making without iteration based on the learned policy and current state [13]. To further enhance the decision-making performance, some studies have introduced an innovative approach by integrating graph neural networks (GNNs) with DRL, i.e., graph DRL (GDRL), and applied it to DN optimization [14]-[16]. The rationale for this integration is the ability of GNNs to effectively capture the complex topological structure of the DN and the relationships between nodes [17], significantly enhancing the adaptability of the model to dynamic changes.

Although the DRL is increasingly being used to address complex DN scheduling challenges, its weaknesses are becoming more apparent. DRL is notably susceptible to disturbances from adversarial noise, with the neural network policies of DRL being highly vulnerable to state adversarial attacks [18]. These attacks introduce slight input perturbations, leading to unpredictable errors and potentially severe security implications [19]. Among the numerous cyber-attacks currently faced by power systems, false data injection attack (FDIA) is considered one of the most serious threats to secure system operation [20]. Load redistribution (LR) is a type of FDIA triggered by false load data, which consequently affects operational actions and leads to economic loss and physical damage to devices due to incorrect operational decisions [21]. Given that DRL makes decisions based on real-time measurements, LR presents an effective strategy for attackers to introduce state adversarial attacks into trained DRL models, thereby significantly undermining their decision-making capabilities. Studies in [18] and [22] explored the vulnerability of the DRL model to data perturbations in power network reconfiguration and optimal power flow. The findings reveal that small disturbances in input data can lead to drastically different control decisions and introduce significant risks. Therefore, it is crucial to improve the defense mechanisms and robustness of DRL models against state attacks before implementing them in actual DNs.

To enhance the robustness of DRL models, the adversarial DRL framework is proposed to identify and adapt to potential adversarial attacks. Specifically, adversarial attacks are defined as attack agents and participate in the training process of defense agents (i.e., robust DRL model). By integrating strategies such as adversarial training and noise injection, this framework strengthens the resistance of the DRL model to input perturbations [23]. In [24], a robust adversarial reinforcement learning method based on a two-player zero-sum Markov game (TZMG) is proposed to improve the robustness against changes in environmental parameters. Similarly, [25] introduces a TZMG model for the cybersecurity in power grids, employing reinforcement learning to simulate attacker behaviors and aid defenders in devising superior strategies for relay protection. Nonetheless, reinforcement learning experiences diminished efficiency and convergence in large-scale scenarios. To address this, [26] presents an alternating training robust DRL based on the state-adversarial Markov decision process (SA-MDP), notably enhancing defenders' capabilities against state adversarial attacks. In [27], an adversary-based robust DRL approach is proposed to strengthen the resilience of DRL-based demand response management systems against cyber-attacks.

Despite these advancements, the security and robustness of the DRL against state adversarial attacks in the optimal DN scheduling remain underexplored. Furthermore, the current robust adversarial DRL framework typically allows the attack agent to arbitrarily modify state values within a specified range, which is impractical. This is because, in power systems, adversarial attacks that fail to adhere to physical characteristics and constraints are easily detected by bad data detection (BDD) mechanisms and, thereby, would not be considered for further decision-making [22]. This situation causes the absence of decision-making experience with stealthy adversarial samples during the adversarial training stage [27], making the learned robust DRL model sensitive to more realistic state attack signals.

To address the aforementioned challenges, this paper proposes a flexibility scheduling method based on robust GDRL (RoGDRL) to enhance the robustness of the DRL-driven scheduling system against state adversarial attacks. Initially, a mathematical model of flexibility scheduling accounting for temperature-dependent resistance is constructed, which improves the operational flexibility and economic efficiency by coordinating various flexibility resources such as tie switches, ESSs, SOPs, PVs, and static var compensators (SVCs). Subsequently, a novel state-adversarial TZMG (SA-TZMG) model for flexibility scheduling is proposed. This model frames the challenge of DRL-based scheduling under state adversarial attacks as a Nash equilibrium problem. Then, a two-stage RoGDRL algorithm with an alternating adversarial training framework is developed to solve the game model. The proposed algorithm utilizes a graph sample and aggregate driven soft actor-critic (SAGESAC) agent to extract feature representations from graph-structured states. The graph sample and aggregate (GraphSAGE) [28] enhances the ability of the proposed algorithm to capture the operational characteristics of the DN and accelerates the learning process. Experimental results demonstrate the effectiveness of the proposed method. The main contributions are summarized as follows.

1) The proposed method integrates weather variables and employs a steady-state thermal balance function to accurately assess temperature-dependent resistance. This enhances the accuracy of flexibility gain evaluation and the reliability

of scheduling decisions.

2) The proposed SA-TZMG model incorporates realistic physical constraints of state attacks, enabling an attacker to generate LR samples that evade the BDD mechanism. This ensures that the defender can make informed decisions to mitigate the impact of more stealthy state adversarial attacks.

3) By combining SAGESAC with an alternating adversarial training framework, the proposed two-stage RoGDRL algorithm demonstrates exceptional robustness against state adversarial attacks and is highly competitive in enhancing operational flexibility compared with existing DRL algorithms.

The remainder of this paper is organized as follows. The mathematical model of flexibility gain improvement is formulated and converted into an SA-TZMG in Sections II and III, respectively. A novel two-stage RoGDRL algorithm is formulated in Section IV. Case study results are presented in Section V. The conclusions are shown in Section VI.

II. Mathematical Model of Flexibility Gain Improvement

A. Definition of Flexibility Gain

To address the flexibility scheduling problem in DNs, a flexibility gain indicator that encompasses node flexibility, branch transfer flexibility, and economic efficiency is first proposed. The flexibility gain during period t can be expressed as:

$$F_t = f_{\mathrm{N},t} + f_{\mathrm{B},t} + f_{\mathrm{C},t} \tag{1}$$

where $f_{\mathrm{N},t}$ is the node flexibility gain; $f_{\mathrm{B},t}$ is the branch transfer flexibility gain; and $f_{\mathrm{C},t}$ is the cost flexibility gain. The flexibility gain metric evaluates improvements in operational flexibility from multiple perspectives following the implementation of scheduling strategies. The sub-indicators for node, branch transfer, and cost flexibility gains share uniform dimensionality, allowing their aggregation into a comprehensive index through summation. A detailed description is provided below.

1) Node Flexibility Gain

Node flexibility, which reflects the local states of flexibility demand and supply [29], is fundamental to operational flexibility. Node voltage is a crucial assessment metric for node flexibility, with voltage deviations beyond permissible limits indicating an extreme lack of node flexibility [5]. Hence, the degree of improvement in nodal voltage deviation before and after implementing scheduling strategies is used to quantify the node flexibility gain.

$$f_{\mathrm{N},t} = \frac{\max\limits_{i \in \Omega_{\mathrm{i}}} \left(\left| 1 - U_{i,t}^{\mathrm{o}} \right| \right) - \max\limits_{i \in \Omega_{\mathrm{i}}} \left(\left| 1 - U_{i,t}^{\mathrm{n}} \right| \right)}{\max\limits_{i \in \Omega_{\mathrm{i}}} \left(\left| 1 - U_{i,t}^{\mathrm{o}} \right| \right) + \lambda} \tag{2}$$

where $U_{i,t}^{\mathrm{o}}$ and $U_{i,t}^{\mathrm{n}}$ are the per-unit voltage values at node i during period t before and after the control, respectively; Ω_{i} is the node set; and λ is a small constant, which is set to be 10^{-8} to avoid a division by zero [30]. This approach ensures that the denominator of the formula does not become zero, even when $\max\limits_{i \in \Omega_i} \left(\left| 1 - U_{i,t}^{\mathrm{o}} \right| \right)$ is very small or zero, thus avoiding computational anomalies.

2) Branch Transfer Flexibility Gain

Branch transfer flexibility signifies the ability of the DN to relocate local flexibility for spatial-temporal balancing [4]. The branch capacity directly reflects the ability of the DN to balance flexibility supply and demand, i.e., the branch transfer flexibility [5]. Hence, this paper quantifies the branch transfer flexibility gain by calculating the proportion of the branch loading rate reduction before and after the implementation of the scheduling strategy.

$$f_{\mathrm{B},t} = \frac{\operatorname*{mean}\limits_{ij \in \Omega_{\mathrm{B}}} \left(I_{ij,t}^{\mathrm{o}} / I_{ij,\max} \right) - \operatorname*{mean}\limits_{ij \in \Omega_{\mathrm{B}}} \left(I_{ij,t}^{\mathrm{n}} / I_{ij,\max} \right)}{\operatorname*{mean}\limits_{ij \in \Omega_{\mathrm{B}}} \left(I_{ij,t}^{\mathrm{o}} / I_{ij,\max} \right) + \lambda} \tag{3}$$

where $I_{ij,t}^{\mathrm{o}}$ and $I_{ij,t}^{\mathrm{n}}$ are the currents of branch ij during period t before and after the control, respectively; $I_{ij,\max}$ is the carrying capacity of branch ij; and Ω_{B} is the branch set.

3) Cost Flexibility Gain

Node and branch transfer flexibilities form the foundational framework for quantitative analysis of DN flexibility [29]. Meanwhile, the economic efficiency often represents a critical consideration in time-series scheduling problems [2]. Hence, the cost flexibility gain is constructed to quantify the rate of change in operation cost, which includes network power loss, device power loss, PV power curtailment loss, and device operation costs.

$$\begin{cases} f_{\mathrm{C},t} = \dfrac{\gamma_{\mathrm{L}} \sum\limits_{ij \in \Omega_{\mathrm{B}}} \left(I_{ij,t}^{\mathrm{o}} \right)^2 r_{ij,t}^{\mathrm{o}} + \gamma_{\mathrm{pv}} P_{\mathrm{o},t}^{\mathrm{cur}} - \left(\gamma_{\mathrm{L}} P_t^{\mathrm{loss}} + \gamma_{\mathrm{pv}} P_{\mathrm{n},t}^{\mathrm{cur}} + \gamma_{\mathrm{A}} \Delta d_t \right)}{\gamma_{\mathrm{L}} \sum\limits_{ij \in \Omega_{\mathrm{B}}} \left(I_{ij,t}^{\mathrm{o}} \right)^2 r_{ij,t}^{\mathrm{o}} + \gamma_{\mathrm{pv}} P_{\mathrm{o},t}^{\mathrm{cur}} + \lambda} \\ \text{s.t. } P_t^{\mathrm{loss}} = \sum\limits_{ij \in \Omega_{\mathrm{B}}} \left(I_{ij,t}^{\mathrm{n}} \right)^2 r_{ij,t}^{\mathrm{n}} + \sum\limits_{i \in \Omega_{\mathrm{sop}}} P_{\mathrm{loss},i,t}^{\mathrm{sop}} \end{cases} \tag{4}$$

where γ_{L} is the electricity price; γ_{pv} is the generation revenue of PV; γ_{A} is the switching action cost; $r_{ij,t}^{\mathrm{o}}$ and $r_{ij,t}^{\mathrm{n}}$ are the resistances of branch ij during period t before and after the control, respectively; Δd_t is the number of actions for discrete devices; $P_{\mathrm{loss},i,t}^{\mathrm{sop}}$ is the power loss of converter i of the SOP during period t; $P_{\mathrm{o},t}^{\mathrm{cur}}$ and $P_{\mathrm{n},t}^{\mathrm{cur}}$ are the PV power curtailments during period t before and after the control, respectively; and Ω_{sop} is the SOP set.

Enhancing the cost flexibility gain can ensure low-cost operation while mitigating the phenomenon of PV power curtailment, thereby improving the utilization rate of PVs. This encourages the investment enthusiasm of DSOs and PV investors for further large-scale PV deployment in the future, thereby promoting shared interests among multiple stakeholders.

4) Discussion on Relationship Between Flexibility Gain and Operational Flexibility

In this study, the relationships between the three sub-indicators of flexibility gain and the operational flexibility is outlined as follows.

1) Node flexibility gain quantitatively reflects the reduction in voltage deviation. A larger $f_{\mathrm{N},t}$ indicates a smaller node voltage deviation after implementing scheduling strategies. Nodes with sufficient flexibility can support the reduction of the voltage deviation to the desired range [5]. Thus, increasing the node flexibility gain can improve the node flexibility.

2) The branch transfer flexibility gain quantitatively reflects the proportion of reduction in branch loading rate. A larger $f_{\mathrm{B},t}$ indicates a lower branch loading rate after implementing scheduling strategies. The lower the loading rate, the greater the remaining capacity of the branch available for transfer flexibility [5]. Thus, enhancing the branch transfer flexibility gain can improve the ability of DN to support flexible transmission, i.e., branch transfer flexibility.

3) The cost flexibility gain quantitatively reflects the proportion of reduction in the system operation cost. A larger $f_{\mathrm{C},t}$ indicates a lower operation cost after implementing scheduling strategies. In DNs, utilizing or enhancing the operational flexibility incurs certain costs, which represent the value attribute of flexibility and are crucial for evaluating the availability and usability of scheduling strategies [29]. Thus, by enhancing the cost flexibility gain, it is possible to achieve economical operation while increasing the flexibility of the node and branch.

Moreover, from the perspective of DN operational performance, improving the flexibility gain can reduce the energy loss cost and PV power curtailment, enhance the voltage stability, and mitigate the branch congestion. Traditionally, achieving these objectives simultaneously often requires a multi-objective optimization framework. However, the proposed flexibility gain indicator simplifies this into a single-objective optimization problem, diminishing the complexity inherent in the multi-objective scheduling problem.

B. Flexibility Gain Improvement Model Considering Temperature-dependent Resistance

The accuracy of flexibility gain calculations depends on the precision of power flow analysis. However, many existing scheduling methods overlook the characteristics of dynamic branch resistance changes during power flow analysis. This is because the resistance of metallic conductors changes with their temperature. Specifically, the relationship is governed by:

$$r_{ij,t}=r_{ij}^{\mathrm{c}}[1+\varpi(T_{ij,t}-T_{\mathrm{c}})] \tag{5a}$$

$$T_{ij,t}=(r_{ij,t}-r_{ij}^{\mathrm{c}})\varpi^{-1}(r_{ij}^{\mathrm{c}})^{-1}+T_{\mathrm{c}} \tag{5b}$$

where $r_{ij,t}$ is the resistance of branch ij during period t; r_{ij}^{c} is the resistance of branch ij in the reference temperature T_{c}; $T_{ij,t}$ is the conductor temperature of branch ij during period t; and ϖ is the temperature constant.

The branch temperature is determined by weather factors and branch current, adhering to the steady heat balance function as specified in IEEE Std 738-2012 [8]:

$$\begin{cases} q_{ij,t}^{\mathrm{s}}+I_{ij,t}^{2}r_{ij,t}=q_{ij,t}^{\mathrm{c}}+q_{ij,t}^{\mathrm{r}} \\ \text{s.t. } q_{ij,t}^{\mathrm{s}}=\rho S_{Q}(\sin\delta_{t})\bar{A}_{ij} \\ \qquad q_{ij,t}^{\mathrm{r}}=0.0178D\varepsilon\dfrac{T_{ij,t}^{4}-T_{\mathrm{a},t}^{4}}{100^{4}} \\ \qquad q_{ij,t}^{\mathrm{c}}=\max\{q_{ij}^{\mathrm{c1}}l_{\mathrm{a}}(T_{ij,t}-T_{\mathrm{a},t}),q_{ij}^{\mathrm{c2}}l_{\mathrm{a}}(T_{ij,t}-T_{\mathrm{a},t})\} \end{cases} \tag{6}$$

where $q_{ij,t}^{\mathrm{c}}$ and $q_{ij,t}^{\mathrm{r}}$ are the heat dissipations via air convection and surface radiation of branch ij, respectively; $q_{ij,t}^{\mathrm{s}}$ is the calorific value of branch ij from the solar radiation; $I_{ij,t}$ is the current of branch ij during period t; S_Q is the solar radiation level; D is the diameter of the branch; ρ is the solar absorptivity; ε is the emissivity of the conductor; $T_{\mathrm{a},t}$ is the ambient temperature; $\bar{A}_{ij}$ is the projected area of the conductor per unit length of branch ij; δ_t is the solar radiation angle; and $q_{ij,t}^{c1}$, $q_{ij,t}^{c2}$, and l_{a} are the convection heat loss coefficients. The details can be referred to in [9].

The flexibility gain improvement model considering temperature-dependent resistance can be written as:

$$\begin{cases} \max\limits_{\boldsymbol{u}_t,\boldsymbol{s}_t}\ F(\boldsymbol{s}_t)=f_{N,t}+f_{B,t}+f_{C,t} \quad t=1,2,\ldots,N_t \\ \text{s.t. } \ \boldsymbol{0}=\boldsymbol{G}(\boldsymbol{s}_t,\boldsymbol{u}_t) \\ \qquad \boldsymbol{0}\leq\boldsymbol{H}(\boldsymbol{s}_t,\boldsymbol{u}_t) \end{cases} \tag{7}$$

where $\boldsymbol{G}(\cdot)$ and $\boldsymbol{H}(\cdot)$ are the vectors of equality and inequality constraint equations, encompassing power balance, operational safety, equipment operational limits, and temperature-dependent resistance constraints [14], [31]; N_t is the number of scheduling periods; $\boldsymbol{u}_t$ is the control vector for topology, ESSs, SOPs, SVCs, and PVs during period t; and $\boldsymbol{s}_t$ is the vector of operation states, including the load demand, PV output, and weather factors, during period t.

Problem (7) can be transformed into a Markov decision process (MDP) and addressed using DRL algorithms. However, state adversarial attacks pose a substantial risk, as they can disturb the inputs to the DRL model. Such disturbances can alter the decisions of the model, impacting the results. LR attacks represent a common type of state adversarial attack for power systems.

C. Modeling of LR Attacks

Referring to [20], LR attacks are based on the following assumptions. ① The measurement attack on balancing and zero-injection nodes is ignored due to its detectability and correctability. ② When the power output from the PV system is zero, the manipulation of the PV output is infeasible. ③ To conceal the attack from system operators, the false data injections do not exceed the normal data by ξ (percentage). Based on those, a valid LR attack during period t for the DN with a high penetration of PV can be modeled as:

$$\Delta\boldsymbol{SB}_t=-\boldsymbol{SF}\cdot\boldsymbol{KD}\cdot(\Delta\boldsymbol{S}_t^{\mathrm{pv}}-\Delta\boldsymbol{S}_t^{\mathrm{load}}) \tag{8a}$$

$$\begin{cases} \boldsymbol{1}^{\mathrm{T}}\Delta\boldsymbol{S}_t^{\mathrm{pv}}=0 \\ \boldsymbol{1}^{\mathrm{T}}\Delta\boldsymbol{S}_t^{\mathrm{load}}=0 \end{cases} \tag{8b}$$

$$\begin{cases} \Delta\boldsymbol{S}_t^{\mathrm{load}}\in[-\xi\boldsymbol{S}_t^{\mathrm{load}},\xi\boldsymbol{S}_t^{\mathrm{load}}] \\ \Delta\boldsymbol{S}_t^{\mathrm{pv}}\in[-\xi(\boldsymbol{S}_{\max}^{\mathrm{pv}}-\boldsymbol{S}_t^{\mathrm{pv}}),\xi(\boldsymbol{S}_{\max}^{\mathrm{pv}}-\boldsymbol{S}_t^{\mathrm{pv}})] \end{cases} \tag{8c}$$

where $\Delta\boldsymbol{S}_t^{\mathrm{load}}$ and $\Delta\boldsymbol{S}_t^{\mathrm{pv}}$ are the power injections of false loads and PVs, respectively; $\Delta\boldsymbol{SB}_t$ is the false branch power measurement injection; $\boldsymbol{SF}$ and $\boldsymbol{KD}$ are the shifting factor and load incidence matrices, respectively; $\boldsymbol{S}_t^{\mathrm{load}}$ and $\boldsymbol{S}_t^{\mathrm{pv}}$ are the load demand and PV output matrices, respectively; and $\boldsymbol{S}_{\max}^{\mathrm{pv}}$ is the PV capacity matrix. Equation (8a) ensures that the attacks can bypass the BDD mechanism, while (8b) ensures that the sum of power injections of false loads or PVs is equal to zero [21]. The BDD is typically based on the principle of residual testing, where the residual r_{e} represents the Euclidean norm of the difference between the measurement vector and its estimation [22]. By comparing the residual r_{e} with a residual threshold τ, if $r_{\mathrm{e}}\leq\tau$, the state against attack can bypass the BDD; otherwise, the adversarial attack fails.

In the robust adversarial framework [27], the attacker deploys state adversarial attacks, as described in (8), aiming to disrupt the input of the DRL controller to diminish flexibility gain. In response, the defender implements a robust DRL controller to maximize flexibility gain, counteracting the attacker's efforts. This interaction forms the basis of a two-player zero-sum game, where one participant's gain is equivalent to the other's loss. To develop a robust DRL model for flexibility scheduling in the face of state attack, the competition between attackers and defenders is conceptualized as a TZMG within the context of state adversarial attacks.

III. Modeling of SA-TZMG

The MDP is a mathematical framework that provides a formalism for modeling sequential decision-making problems. TZMG is considered an extension of MDP to game-theoretic scenarios. It consists of six-tuple as follows:

$$\left\langle \boldsymbol{S}, \boldsymbol{A}^{\mathrm{D}}, \boldsymbol{A}^{\mathrm{A}}, \boldsymbol{P}, R, \gamma \right\rangle \tag{9}$$

where $\boldsymbol{S}$ is the game state space; $\boldsymbol{A}^{\mathrm{D}}$ and $\boldsymbol{A}^{\mathrm{A}}$ are the action spaces for two players; $\boldsymbol{P}$ is the state transition function; R is the reward function; and γ is the discount factor. In TZMG, while one player attempts to maximize the cumulative reward from the game, the other player seeks to minimize this value. In this work, the maximizer is represented as the defender, while the minimizer is defined as the attacker.

Notably, in the conventional TZMG model, actions of both defender and attacker influence the environment simultaneously, leading to the generation of rewards [32]. This model, however, does not accurately reflect attack-defense interaction patterns under state adversarial conditions. Consequently, the TZMG is augmented through integration with SA-MDP, culminating in the developing of an SA-TZMG. This model more effectively reflects the interaction between the attacker and the defender. This interaction process and its effects on the environment are represented in Fig. 1.

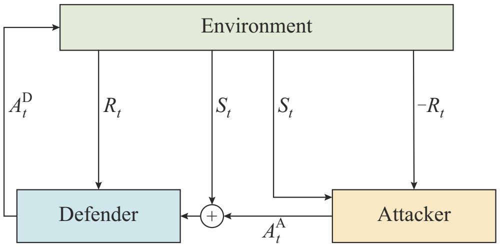


Fig. 1. Interaction between players and its effects on environment.

As shown in Fig. 1, the attacker's adversarial attack $\boldsymbol{A}_t^A$ primarily targets the defender's perception of the state rather than exerting a direct influence on the environment. The modeling process of SA-TZMG is as follows.

1) State space: the state of the DN is usually defined as the Euclidean space. However, stacking the features of nodes by a specific order may cause a loss of topology information and dependencies between nodes [33]. Consequently, there is a critical need to devise a graph data structure that encapsulates spatiotemporal information, thereby more accurately mirroring the operation state of the system.

Graph data can be represented as $\boldsymbol{S}=(\Omega_{\mathrm{i}}, \Omega_{\mathrm{e}}, \boldsymbol{X})$, where Ω_{e} is the set of edges; and $\boldsymbol{X}$ is the set of features, including the node feature matrix $\boldsymbol{X}_{\mathrm{bus}}$ and the feature matrix $\boldsymbol{X}_{\mathrm{edge}}$. For the DN, $\boldsymbol{X}_{\mathrm{edge}}$ is the adjacency matrix. Moreover, the node feature is the system operation state. In summary, the state during period t can be represented as $\boldsymbol{S}_t=(\Omega_{\mathrm{i}}, \Omega_{\mathrm{e}}, \boldsymbol{X}_{\mathrm{bus},t}, \boldsymbol{X}_{\mathrm{edge},t})$. $\boldsymbol{X}_{\mathrm{bus},t}$ during period t can be written as:

$$\boldsymbol{X}_{\mathrm{bus},t}=[\boldsymbol{P}_{\mathrm{inj},t}, \boldsymbol{Q}_{\mathrm{inj},t}, \boldsymbol{T}_{\mathrm{a},t}, \boldsymbol{V}_{\mathrm{w},t}, \boldsymbol{A}_{\mathrm{w},t}, \boldsymbol{Q}_{\mathrm{s},t}]\in \mathbb{R}^{n\times d} \tag{10}$$

where the operation state mainly comprises the active power injection of the node $\boldsymbol{P}_{\mathrm{inj},t}$, reactive power injection of the node $\boldsymbol{Q}_{\mathrm{inj},t}$, and weather factors, including wind speed $\boldsymbol{V}_{\mathrm{w},t}$, wind angle $\boldsymbol{A}_{\mathrm{w},t}$, temperature $\boldsymbol{T}_{\mathrm{a},t}$, and solar irradiance $\boldsymbol{Q}_{\mathrm{s},t}$; and n and d are the numbers of rows and columns of the feature matrix, respectively.

Remarke: currently, the primary method for collecting weather information is through the automatic weather station (AWS). It can transmit weather data via wired or wireless communication and is highly automated, accurate, and reliable [34]. Thus, a possible implementation of the proposed method is to deploy AWS in the DN to collect the required weather data, which are then transmitted to the control center via a communication network. Additionally, to further ensure the accuracy of data collection, it can be linked with geographic information systems. This linkage allows for precisely attributing measured weather information to the specific branches, thereby providing more accurate and timely data support [35].

2) Defender's action: the defender's action $\boldsymbol{A}^{\mathrm{D}}$ is decomposed into a Cartesian product of two sub-spaces. The continuous sub-action space $\boldsymbol{A}_{\mathrm{c}}$ is the control strategy of ESS, SVC, PV, and SOP. The discrete sub-action $\boldsymbol{A}_{\mathrm{d}}$ is the control strategy for topology. the defender's action during period t can be represented as:

$$\boldsymbol{A}_t^{\mathrm{D}}=\{\boldsymbol{A}_{\mathrm{d},t}, \boldsymbol{A}_{\mathrm{c},t}\}=\{\boldsymbol{a}_t^{\mathrm{dnr}},[\boldsymbol{a}_t^{\mathrm{pv}}, \boldsymbol{a}_t^{\mathrm{svc}}, \boldsymbol{a}_t^{\mathrm{sop}}, \boldsymbol{a}_t^{\mathrm{ess}}]\} \tag{11}$$

where $\boldsymbol{a}_t^{\mathrm{pv}}$, $\boldsymbol{a}_t^{\mathrm{svc}}$, $\boldsymbol{a}_t^{\mathrm{ess}}\in[-1,1]$ are the control actions for PV, SVC, and ESS, respectively, detailed in [14]; and $\boldsymbol{a}_t^{\mathrm{dnr}}$ and $\boldsymbol{a}_t^{\mathrm{sop}}$ are the control actions for topology and SOP, respectively, detailed in [31].

3) Attacker's action: the attacker's action at each time step, which is a continuous variable, is defined as $\boldsymbol{a}_{\mathrm{att},t}=[\boldsymbol{a}_{\mathrm{att,load},t}, \boldsymbol{a}_{\mathrm{att,pv},t}]\in[-\xi\boldsymbol{1}^{\mathrm{T}}, \xi\boldsymbol{1}^{\mathrm{T}}]$, where $\boldsymbol{a}_{\mathrm{att,load},t}$ and $\boldsymbol{a}_{\mathrm{att,pv},t}$ are the attack actions targeting the load and PV state variables, respectively. This study sets ξ to be 0.3 [21]. During period t, the attacker can alter the load demand and PV output, thereby modifying the apparent power of the node injection to disrupt the defender's observed state $\boldsymbol{S}_t$.

$$[\Delta\boldsymbol{S}_t^{\mathrm{load}}, \Delta\boldsymbol{S}_t^{\mathrm{pv}}]=\boldsymbol{a}_{\mathrm{att},t}[\boldsymbol{S}_t^{\mathrm{load}}, \boldsymbol{S}_{\mathrm{max}}^{\mathrm{pv}}-\boldsymbol{S}_t^{\mathrm{pv}}]^{\mathrm{T}} \tag{12}$$

Diverging from existing adversarial attack modeling approaches, this study further constrains the attacker's actual executed actions to ensure compliance with the LR mechanism as:

$$\boldsymbol{A}_t^{\mathrm{A}}=\begin{cases}[\Delta\boldsymbol{S}_t^{\mathrm{load}}, \Delta\boldsymbol{S}_t^{\mathrm{pv}}]-[M(\Delta\boldsymbol{S}_t^{\mathrm{load}})\boldsymbol{1}^{\mathrm{T}}, M(\Delta\boldsymbol{S}_t^{\mathrm{pv}})\boldsymbol{1}^{\mathrm{T}}] & \eta_t=0\\ \boldsymbol{0} & \eta_t=1\end{cases} \tag{13}$$

where $M(\cdot)$ is the operation of calculating the mean value; and η_t is a Boolean variable. If attacker's action satisfies the actual physical constraints in (8), $\eta_t=0$; otherwise, $\eta_t=1$.

4) Reward: the reward in SA-TZMG is one of the critical factors determining the game result. It is represented as the immediate reward provided by the environment when the defender acts $\boldsymbol{A}^{\mathrm{D}}$ and the attacker acts $\boldsymbol{A}^{\mathrm{A}}$. The reward during period t is expressed as the flexibility gain with a penalty term in the paper.

$$R_t(\boldsymbol{S}_t,\boldsymbol{A}_t^{\mathrm{D}},\boldsymbol{A}_t^{\mathrm{A}})=F_t-\kappa_t\vartheta \tag{14}$$

where $\kappa_t=0$ if $\boldsymbol{A}_t^{\mathrm{D}}$ satisfies the operational security constraints, otherwise, $\kappa_t=1$; and ϑ is a negative constant.

Reference [36] has proven that TZMG has a Nash equilibrium joint policy (π_θ,π_φ), where π_θ and π_φ are the policies of the defender and attacker, respectively. Nash equilibrium defines the highest payoff a defender can achieve when facing the most powerful attacker, which is equivalent to the max-min solution. In SA-TZMG, the defender still seeks to maximize its resilience against the most powerful attacker. Thus, the Nash equilibrium joint policy can be written as:

$$\begin{cases}\max\limits_{\pi_\varphi}\min\limits_{\pi_\theta}\mathcal{V}(\boldsymbol{S}_t,\boldsymbol{A}_t^{\mathrm{D}},\boldsymbol{A}_t^{\mathrm{A}})=\min\limits_{\pi_\theta}\max\limits_{\pi_\varphi}\mathcal{V}(\boldsymbol{S}_t,\boldsymbol{A}_t^{\mathrm{D}},\boldsymbol{A}_t^{\mathrm{A}})\\ \text{s.t. } \mathcal{V}(\boldsymbol{S}_t,\boldsymbol{A}_t^{\mathrm{D}},\boldsymbol{A}_t^{\mathrm{A}})=\mathbb{E}\left[\sum\limits_{\Delta t=0}^{T}\gamma^{\Delta t}R_{t+\Delta t+1}(\boldsymbol{S}_t,\boldsymbol{A}_t^{\mathrm{D}},\boldsymbol{A}_t^{\mathrm{A}})\right]\\ \quad\boldsymbol{A}_t^{\mathrm{D}}\sim\pi_\varphi(\cdot|\boldsymbol{S}_t)\\ \quad\boldsymbol{A}_t^{\mathrm{A}}\sim\pi_\theta(\cdot|\boldsymbol{S}_t)\end{cases} \tag{15}$$

where $\mathcal{V}(\cdot)$ is the expected cumulative reward function; T is the total number of optimization periods; and $\mathbb{E}[\cdot]$ is the expected value function. Traditional DRL algorithms mainly focus on solving single-agent MDP and are not designed to directly address max-min problems [32]. To end this, the following section introduces a two-stage RoGDRL algorithm.

IV. Two-stage RoGDRL Algorithm

A. Framework of Two-stage RoGDRL Algorithm

In TZMG, once one player's policy is fixed, the max-min problem becomes a single-agent MDP, and a deterministic policy is sufficient to achieve optimality [32]. Thus, inspired by [32] and [37] on solving the max-min problem, this work proposes a two-stage RoGDRL algorithm to tackle the problem (15). It consists of Stage I, which focuses on attacker's policy learning, and Stage II, which is dedicated to robust policy learning.

In Stage I, the training of the attacker aims to ascertain the optimal attacker's policy π_θ, keeping the defender's policy π_φ fixed and aiming to minimize the defender's cumulative reward. It is framed as a constrained minimization problem:

$$\begin{cases}\min\limits_{\pi_\theta}\mathcal{V}(\boldsymbol{S}_t,\boldsymbol{A}_t^{\mathrm{D}},\boldsymbol{A}_t^{\mathrm{A}})\\ \text{s.t. } \boldsymbol{A}_t^{\mathrm{D}}\sim\pi_\varphi(\cdot|\boldsymbol{S}_t+\boldsymbol{A}_t^{\mathrm{A}})\end{cases} \tag{16}$$

It is essential to highlight that π_φ is pre-trained at this stage. The rationale behind pre-training π_φ without the influence of adversarial attacks lies in its effectiveness in establishing an optimal initial exploration strategy for the attacker [27].

In Stage II, the defender focuses on augmenting its resilience to state attacks by developing a robust defense policy, with the attacker's policy held constant. This stage is characterized as a constrained maximization problem:

$$\begin{cases}\max\limits_{\pi_\varphi}\mathcal{V}(\hat{\boldsymbol{S}}_t,\boldsymbol{A}_t^{\mathrm{D}},\boldsymbol{A}_t^{\mathrm{A}})\\ \text{s.t. } \hat{\boldsymbol{S}}_t=\boldsymbol{S}_t+(\boldsymbol{A}_t^{\mathrm{A}}\sim\pi_\theta(\cdot|\boldsymbol{S}_t))\end{cases} \tag{17}$$

where $\hat{\boldsymbol{S}}_t$ is the perturbed state after FDIA. The policy learning tasks for both stages can be addressed by employing the proposed SAGESAC. The specific algorithm for these two stages will be detailed below.

B. Stage I: Attacker's Policy Learning

At this stage, state attack signals generated by the attacker are continuous variables. Thus, the soft actor-critic (SAC) is adopted as the foundational framework to learn π_θ. The objective can be expressed as a maximizing problem with a negative reward function:

$$\begin{cases}J(\pi_\theta)=\max\sum\limits_{t=0}^{T}\mathbb{E}\left[-R_t(\boldsymbol{S}_t,\boldsymbol{A}_t^{\mathrm{D}},\boldsymbol{A}_t^{\mathrm{A}})+\alpha\mathcal{H}(\pi_\theta(\boldsymbol{S}_t))\right]\\ \text{s.t. } \boldsymbol{A}_t^{\mathrm{D}}\sim\pi_\varphi(\cdot|\boldsymbol{S}_t+\boldsymbol{A}_t^{\mathrm{A}})\\ \quad\mathcal{H}(\pi_\theta(\boldsymbol{S}_t))=\mathbb{E}_{\boldsymbol{A}_t^{\mathrm{D}}\sim\pi_\theta}\left[-\ln\pi_\theta(\boldsymbol{A}_t^{\mathrm{D}}|\boldsymbol{S}_t)\right]\end{cases} \tag{18}$$

where $\mathcal{H}(\cdot)$ is the policy entropy, which is a measure of the randomness in the action selection of the policy, encouraging exploration by penalizing certainty in action choices; $\mathbb{E}_{\boldsymbol{A}_t^{\mathrm{D}}\sim\pi_\theta}$ denotes the average calculation under all possible actions $\boldsymbol{A}_t^{\mathrm{D}}$; α is the temperature parameter used to balance the relationship between the policy entropy and expected rewards; and $-\ln\pi_\theta(\boldsymbol{A}_t^{\mathrm{D}}|\boldsymbol{S}_t)$ is the negative value of the logarithmic probability of action $\boldsymbol{A}_t^{\mathrm{D}}$ given the state $\boldsymbol{S}_t$.

To exploit the critical attributes of the observation, this study introduces GraphSAGE as a feature extractor to efficiently encapsulate the characteristics of graph-structured states. Traditional feature extractor, named graph convolutional network (GCN), utilizes a transductive learning approach that requires static graph structures [17]. Nonetheless, the correlation among neighboring nodes is subject to temporal fluctuations, attributed primarily to DN reconfiguration, a phenomenon that occurs routinely in the DN. In contrast, GraphSAGE, as one of the most advanced GNN frameworks, employs an inductive learning strategy capable of sampling and aggregating features from neighboring nodes of a node. This methodology makes GraphSAGE particularly advantageous for application within large-scale, dynamic graphs [28]. The critical steps of GraphSAGE are as follows.

1) Sampling from neighboring nodes. For each node i, a fixed number of neighboring nodes are randomly sampled from its neighboring node set $N(i)$, thereby reducing the number of neighboring nodes to be processed and, consequently, the computational complexity of the model.

2) Aggregating features from neighboring nodes. Specific aggregators are used to aggregate features of the sampled neighboring nodes, obtaining a comprehensive representation of neighborhood features.

$$\boldsymbol{h}_{\mathcal{N}(i)}^{k}=AGG(\{\boldsymbol{h}_m^{k-1},\forall m\in\mathcal{N}(i)\}) \tag{19}$$

where AGG denotes the aggregation operation, and in practice, it can be various aggregators such as mean aggregator; $\boldsymbol{h}_{\mathcal{N}(i)}^{k}$ is the k^{th} aggregated neighborhood feature; and $\boldsymbol{h}_{m}^{k-1}$ is the $(k-1)^{\text{th}}$ aggregated neighborhood feature for node m.

3) Updating feature representation of node. $\boldsymbol{h}_{\mathcal{N}(i)}^{k}$ is combined with the feature of the current node $\boldsymbol{h}_{i}^{k-1}$ (e.g., through concatenation). The feature representation of the node is then updated using a neural network layer (e.g., a fully connected layer).

$$\boldsymbol{h}_{i}^{k}=ReLU(\boldsymbol{w}_{k}\cdot CONCAT(\boldsymbol{h}_{i}^{k-1},\boldsymbol{h}_{\mathcal{N}(i)}^{k})+\boldsymbol{b}_{k}) \tag{20}$$

where $\boldsymbol{w}_k$ and $\boldsymbol{b}_k$ are the learnable coefficient matrices; $ReLU(\cdot)$ is the ReLU activation function; and $CONCAT(\cdot)$ denotes the concatenation operation. In the subsequent stage of the GraphSAGE process, the newly generated features $\boldsymbol{h}_{i}^{k}$ will be utilized. Following K aggregation layers, a feature vector $\boldsymbol{H}=\boldsymbol{h}_{i}^{K}$, $i\in\Omega_{\text{b}}$ is produced as the output.

As shown in Fig. 2, the proposed SAGESAC incorporates two GraphSAGE layers (SAGE 1 and SAGE 2) to modify the policy and critic networks within the SAC.

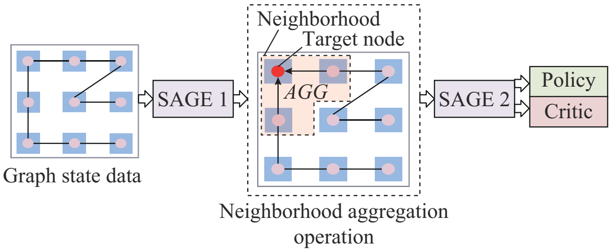


Fig. 2. Feature extraction process driven by GraphSAGE.

The output of the graph policy network can be parameterized by a Gaussian distribution $\mathbb{N}(\cdot)$, which can be denoted as:

$$\pi_{\theta}(\cdot|\boldsymbol{S}_{t})=\mathbb{N}(\mu(\boldsymbol{S}_{t}),\sigma^{2}(\boldsymbol{S}_{t})) \tag{21}$$

where $\mu(\boldsymbol{S}_t)$ and $\sigma(\boldsymbol{S}_t)$ are the mean and variance of the action for the policy network, respectively. Then, the mini-batch data from the replay buffer are sampled, and θ is typically updated using gradient descent. The loss function is expressed as:

$$J(\pi_{\theta})=\mathbb{E}_{\boldsymbol{S}_{t}\sim\mathcal{D},\boldsymbol{A}_{t}^{\text{A}}\sim\pi_{\theta}}[\alpha\mathcal{H}(\pi_{\theta}(\boldsymbol{S}_{t}))-Q_{\beta}(\boldsymbol{S}_{t},\boldsymbol{A}_{t}^{\text{A}})] \tag{22}$$

$$\theta\leftarrow\theta+\tau\nabla_{\theta}J(\pi_{\theta}) \tag{23}$$

where $J(\pi_\theta)$ is the loss function of the policy network; $\nabla_{\theta}J(\pi_{\theta})$ is the differential form of $J(\pi_\theta)$; τ is the learning rate; $\mathbb{E}_{\boldsymbol{S}_{t}\sim\mathcal{D},\boldsymbol{A}_{t}^{\text{A}}\sim\pi_{\theta}}$ denotes the average calculation under all possible states $\boldsymbol{S}_t$ and actions $\boldsymbol{A}_{t}^{\text{A}}$; $\mathcal{D}$ is the replay buffer; and $Q_{\beta}(\boldsymbol{S}_{t},\boldsymbol{A}_{t}^{\text{A}})$ is the Q-function value, which is parameterized by the graph critic network. The Q-function updates the parameters via the following method:

$$\begin{cases}J(Q_{\beta})=\mathbb{E}_{(\boldsymbol{S}_{t},\boldsymbol{A}_{t}^{\text{A}})\sim\mathcal{D}}[(Q_{\beta}(\boldsymbol{S}_{t},\boldsymbol{A}_{t}^{\text{A}})-\boldsymbol{y})^{2}]\\ \boldsymbol{y}=-R_{t}+\gamma\mathbb{E}_{\boldsymbol{A}_{t+1}^{\text{A}}\sim\pi_{\theta}}[\tilde{Q}_{\tilde{\beta}}(\boldsymbol{S}_{t+1},\boldsymbol{A}_{t+1}^{\text{A}})+\alpha\mathcal{H}(\pi_{\theta}(\boldsymbol{S}_{t}))]\end{cases} \tag{24}$$

where $J(Q_\beta)$ is the loss function of the Q-function; $\mathbb{E}_{(\boldsymbol{S}_{t},\boldsymbol{A}_{t}^{\text{A}})\sim\mathcal{D}}$ denotes the average calculation under all possible states $\boldsymbol{S}_t$ and actions $\boldsymbol{A}_{t}^{\text{A}}$; $\mathbb{E}_{\boldsymbol{A}_{t+1}^{\text{A}}\sim\pi_{\theta}}$ denotes the average calculation under all possible actions $\boldsymbol{A}_{t+1}^{\text{A}}$; and $\tilde{Q}_{\tilde{\beta}}(\boldsymbol{S}_{t+1},\boldsymbol{A}_{t+1}^{\text{A}})$ is the target Q-function value. Periodically, the parameters of the critic network are copied to the target critic network to stabilize learning.

$$\beta\leftarrow\beta+\tau\nabla_{\beta}J(Q_{\beta}) \tag{25}$$

$$\tilde{\beta}\leftarrow\upsilon\beta+(1-\upsilon)\beta \tag{26}$$

where β and $\tilde{\beta}$ are the sets of parameters for the critic and target critic networks, respectively; $\nabla_{\beta}J(Q_{\beta})$ is the differential form of $J(Q_\beta)$; and υ is the soft update coefficient. Following the training framework of the naive SAC, the attacker can learn an attacker's policy π_θ to achieve the worst-case performance under a limited state attack.

C Stage II: Robust Policy Learning

This stage addresses a robust policy learning problem in a hybrid discrete-continuous action space. It is worth noting that the robust policy learning aims to enhance decision-making resilience under state adversarial attacks. Thus, the perturbed state $\hat{\boldsymbol{S}}_t$ is used to train the neural network. R_t obtained during the training process is the flexibility gain for $\boldsymbol{A}_{t}^{\text{D}}$ performed by the DN environment under a clean state $\boldsymbol{S}_t$.

The SAGESAC is extended by introducing two parallel graph policy networks to address policy learning issues in mixed discrete-continuous action spaces. One is designated for generating discrete actions, and the other is for generating continuous actions. Its objective function is still to maximize the sum of expected rewards and policy entropy. However, it uniquely accounts for the policy entropy of discrete and continuous actions. The objective function can be expressed as:

$$J(\pi_{\varphi})=\max\sum_{t=0}^{T}\mathbb{E}[R_{t}(\boldsymbol{S}_{t},\boldsymbol{A}_{t}^{\text{D}},\boldsymbol{A}_{t}^{\text{A}})+\alpha\mathcal{H}(\pi_{\varphi_{\text{d}}}(\hat{\boldsymbol{S}}_{t}))+\alpha\mathcal{H}(\pi_{\varphi_{\text{c}}}(\hat{\boldsymbol{S}}_{t}))] \tag{27}$$

where $\pi_{\varphi_{\text{d}}}$ is the discrete action policy network, employing the Gumbel-Softmax function for selecting discrete actions $\boldsymbol{A}_{\text{d},t}$ [16]; $\pi_{\varphi_{\text{c}}}$ is the continuous action policy network, and its structure for outputting continuous actions $\boldsymbol{A}_{\text{c},t}$ can be referenced in (21); $\pi_{\varphi}:=\{\pi_{\varphi_{\text{d}}},\pi_{\varphi_{\text{c}}}\}$ is the robust joint policy; and $\boldsymbol{A}_{t}^{\text{D}}=[\boldsymbol{A}_{\text{d},t},\boldsymbol{A}_{\text{c},t}]$ is sampled from two policies. Regarding both the discrete and continuous action policy networks, their loss functions can be defined as:

$$\begin{cases}J(\pi_{\varphi_{\text{d}}})=\mathbb{E}_{\hat{\boldsymbol{S}}_{t}\sim\mathcal{D},\boldsymbol{A}_{t}^{\text{D}}\sim\pi_{\varphi}}[\alpha\mathcal{H}(\pi_{\varphi_{\text{d}}}(\hat{\boldsymbol{S}}_{t}))-Q_{\psi}(\hat{\boldsymbol{S}}_{t},\boldsymbol{A}_{t}^{\text{D}})]\\ J(\pi_{\varphi_{\text{c}}})=\mathbb{E}_{\hat{\boldsymbol{S}}_{t}\sim\mathcal{D},\boldsymbol{A}_{t}^{\text{D}}\sim\pi_{\varphi}}[\alpha\mathcal{H}(\pi_{\varphi_{\text{c}}}(\hat{\boldsymbol{S}}_{t}))-Q_{\psi}(\hat{\boldsymbol{S}}_{t},\boldsymbol{A}_{t}^{\text{D}})]\end{cases} \tag{28}$$

where $\mathbb{E}_{\hat{\boldsymbol{S}}_{t}\sim\mathcal{D},\boldsymbol{A}_{t}^{\text{D}}\sim\pi_{\varphi}}$ denotes the average calculation under all possible states $\hat{\boldsymbol{S}}_t$ and actions $\boldsymbol{A}_{t}^{\text{D}}$; and $Q_{\psi}(\hat{\boldsymbol{S}}_{t},\boldsymbol{A}_{t}^{\text{D}})$ is the Q-function value for robust policy learning.

The gradient descent method is also employed to optimize the loss functions of the policy network, aiming to learn the optimal parameters as follows:

$$\begin{cases}\varphi_{\text{d}}\leftarrow\varphi_{\text{d}}+\tau\nabla_{\varphi_{\text{d}}}J(\pi_{\varphi_{\text{d}}})\\ \varphi_{\text{c}}\leftarrow\varphi_{\text{c}}+\tau\nabla_{\varphi_{\text{c}}}J(\pi_{\varphi_{\text{c}}})\end{cases} \tag{29}$$

where $\nabla_{\varphi_{\text{d}}}J(\pi_{\varphi_{\text{d}}})$ and $\nabla_{\varphi_{\text{c}}}J(\pi_{\varphi_{\text{c}}})$ are the differential forms of

$J(\pi_{\varphi_d})$ and $J(\pi_{\varphi_c})$, respectively.

The critic network Q_ψ is updated by minimizing the following loss function $J(Q_\psi)$:

$$\begin{cases} J(Q_\psi)=\mathbb{E}_{(\hat{\boldsymbol{S}}_t,\boldsymbol{A}_t^{\mathrm{D}})\sim\mathcal{D}}[(Q_\psi(\hat{\boldsymbol{S}}_t,\boldsymbol{A}_t^{\mathrm{D}})-\boldsymbol{y})^2] \\ \boldsymbol{y}=-R_t+\gamma\mathbb{E}_{\boldsymbol{A}_{t+1}^{\mathrm{D}}\sim\pi_\varphi}[\tilde{Q}_{\tilde{\psi}}(\hat{\boldsymbol{S}}_{t+1},\boldsymbol{A}_{t+1}^{\mathrm{D}})+\alpha\mathcal{H}(\pi_\varphi(\hat{\boldsymbol{S}}_t)] \end{cases} \tag{30}$$

$$\psi\leftarrow\psi+\tau\nabla_\psi J(Q_\psi) \tag{31}$$

$$\tilde{\psi}\leftarrow\upsilon\psi+(1-\upsilon)\psi \tag{32}$$

where $\mathbb{E}_{(\hat{\boldsymbol{S}}_t,\boldsymbol{A}_t^{\mathrm{D}})\sim\mathcal{D}}$ denotes the average calculation under all possible states $\hat{\boldsymbol{S}}_t$ and actions $\boldsymbol{A}_t^{\mathrm{D}}$; $\mathbb{E}_{\boldsymbol{A}_{t+1}^{\mathrm{D}}\sim\pi_\varphi}$ denotes the average calculation under all possible actions $\boldsymbol{A}_{t+1}^{\mathrm{D}}$; $\nabla_\psi J(Q_\psi)$ is the differential form of $J(Q_\psi)$; $\tilde{Q}_{\tilde{\psi}}(\hat{\boldsymbol{S}}_{t+1},\boldsymbol{A}_{t+1}^{\mathrm{D}})$ is the target Q-function value for robust policy learning; and ψ and $\tilde{\psi}$ are the sets of parameters for the critic network and target critic network, respectively. A robust scheduling policy under the attacker's policy π_θ can be obtained by optimizing π_{φ_d} and π_{φ_c}.

D. Alternate Training of Stage I and Stage II

Following the pre-training of the defender's policy, a two-stage alternate training sequence is initiated. In the first stage, the pre-training defender's parameters are held constant while the attacker's parameters are optimized to learn the attacker's policy. After completing C_1 training episodes, the process shifts to optimize the defender's parameters, keeping the attacker's parameters static, to develop a robust defense policy. After training the defender for C_2 episodes, this cycle is then repeated. This iterative training strategy ensures continuous improvement and adaptation of both agents. The alternate training process of RoGDRL is shown in Algorithm 1.

Algorithm 1: alternate training process of RoGDRL

```
Input: number of alternate periods $C$, and numbers of episodes $C_1$ and $C_2$
   for training Stages I and II
Output: parameters of attacker and defender
  for alternate periods of $1, 2, \ldots, C$ do
  Get the optimal defender policy
  for episodes of $1, 2, \ldots, C_1$ do
    for $t = 1, 2, \ldots, N_t$ do
      Output the action $\boldsymbol{A}_t^{\mathrm{A}}\sim\pi_\theta(\cdot|\boldsymbol{S}_t)$ with fixed $\pi_\varphi$
      Calculate the reward $R_t$
      Store $\{\boldsymbol{S}_t, \boldsymbol{A}_t^{\mathrm{A}}, -R_t, \boldsymbol{S}_{t+1}\}$ in buffer and update parameters of $\pi_\theta$
    end for
  end for
    for episodes of $1, 2, \ldots, C_2$ do
      for $t = 1, 2, \ldots, N_t$ do
        Output the action $\boldsymbol{A}_t^{\mathrm{D}}\sim\pi_\varphi(\cdot|\boldsymbol{S}_t+\boldsymbol{A}_t^{\mathrm{A}})$ with fixed $\pi_\theta$
        Calculate the reward $R_t$
        Store $\{\hat{\boldsymbol{S}}_t, \boldsymbol{A}_t^{\mathrm{A}}, R_t, \hat{\boldsymbol{S}}_{t+1}\}$ in buffer and update parameters of $\pi_\varphi$
      end for
    end for
 end for
```

V. Case Study

This paper uses the load and PV data from Jinan, China in 2020 to generate plenty of load and PV profiles. Then, the resultant load and PV instances are normalized to match the scale of power demands in the simulated system to train the proposed method. The weather data of the region in 2020 are from Solcast [38]. Next, the proposed algorithm is trained using Pytorch on the NVIDIA RTX 3090 GPU with 12 GB RAM. The modified IEEE 123-bus system in Fig. 3 is used to verify the proposed method. The range of the voltage is [0.93 p.u., 1.07 p.u.]. γ_{L} and γ_{pv} are set to be 400 ¥/WMh and 800 ¥/WMh, respectively. γ_{A} is set to be ¥2. Detailed parameter settings of SOP, SVCs, ESSs, PVs, and hyperparameter settings for RoGDRL can be found in [39].

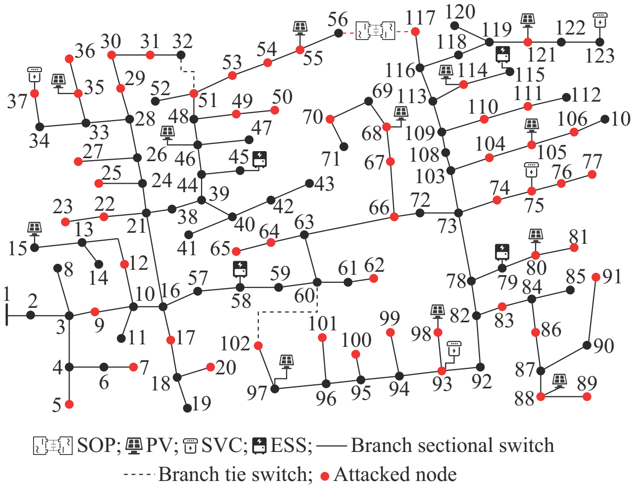


Fig. 3. Modified IEEE 123-bus system.

A. Analysis of Superiority of Proposed Algorithm

The proposed algorithm is compared with the existing GDRL algorithms, including graph attention soft actor-critic (GATSAC) [10] and graph convolutional network soft actor-critic (GCNSAC) [14]. The cumulative reward curves for different algorithms are shown in Fig. 4. While SAGESAC, GATSAC, and GCNSAC are trained under nominal conditions, the proposed algorithm is specifically trained in environments with state adversarial attacks.

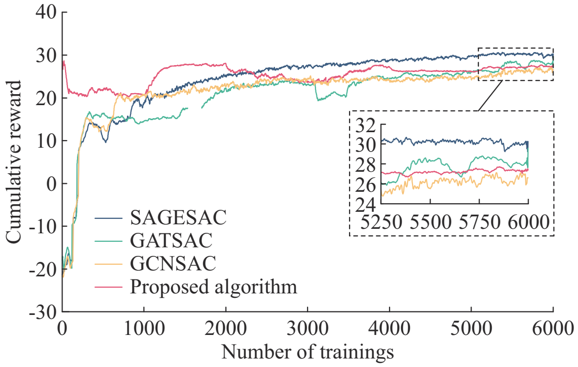


Fig. 4. Cumulative reward curves for different algorithms.

Figure 4 shows that in the final stages of training, the cumulative reward obtained by SAGESAC exhibits minor fluctuations around a fixed value, indicating gradual convergence of the proposed algorithm. This suggests that the algorithm has mastered a scheduling policy capable of improving flexibility gains. Moreover, compared with GATSAC and GCNSAC, the integration of GraphSAGE significantly enhances the feature recognition capability of the proposed algorithm in large-scale complex systems, enabling SAGESAC to achieve higher cumulative rewards.

Furthermore, the cumulative reward curves of the proposed algorithm demonstrate significant oscillations during the adversarial training process. At the stage where the attack strategy is being learned, a decrease in the cumulative reward curve indicates that the attacker's adversarial strategy has successfully disrupted the defender's decision-making process. On the contrary, an increase in the cumulative reward curve indicates that the defender is learning how to effectively counteract the attacker's strategy, thus progressively enhancing the quality of its decision-making. This illustrates that the defender adjusts its strategy in response to adversarial challenges to maximize long-term rewards. As time progresses, the cumulative reward curve tends to stabilize, implying that the proposed algorithm becomes increasingly efficient and robust in counteracting the impacts of state adversarial attacks through adversarial training.

Notably, the final reward performance of the proposed algorithm, which operates in an adversarial training environment, is lower than that of SAGESAC and GATSAC, both of which operate in a clean environment, free from adversarial attacks. This difference arises because the proposed algorithm is designed to address the max-min problem, as described in (15), rather than solely maximizing rewards, in contrast to SAGESAC and GATSAC. By sacrificing some reward optimality, the proposed algorithm enhances its robustness against state attacks. Although its decision outcomes are not optimal, the proposed algorithm maintains commendable performance stability in the face of state adversarial attacks. This aspect will be explored further in subsequent analyses.

B. Analysis of Superiority of Proposed SA-TZMG Model

In adversarial training, a powerful and stealthy attacker is crucial, ensuring the defender can achieve optimal rewards in worst-case scenarios [26]. Thus, to demonstrate the necessity of considering actual physical constraints of state attacks, three types of attack scenarios are established.

1) Scenario A: without considering the physical constraints and BDD mechanism.

2) Scenario B: without considering the physical constraints.

3) Scenario C: considering the physical constraints.

The attack vectors generated in Scenarios A and B are consistent, and the only difference is whether BDD is performed to eliminate anomalous attack vectors. The test rewards of three algorithms after encountering these three attack scenarios and the perturbation residual statistics for Scenarios A and C are shown in Fig. 5.

As shown in Fig. 5(a), in Scenario A, the lack of the consideration for the physical constraints and BDD mechanism results in relatively larger state disturbances, which leads to a significant reduction in the rewards of the three algorithms. However, as shown in Fig. 5(b), the attack signals generated in Scenario A have a 62.2% probability of exceeding the residual threshold across all load levels. This implies that only 47.8% of the generated attacks can bypass the BDD mechanism. Since it is not possible to guarantee that all generated attacks are effective, this results in the highest reward for Scenario B in Fig. 5(a). In other words, the effect of state adversarial attacks is the weakest when actual physical constraints are not considered. In comparison, Scenario C enables all attack signals to bypass the BDD mechanism, thus ensuring the effectiveness of the attacks. It can be concluded that actual physical constraints enhance the stealth and precision of state adversarial attacks, making them more reflective of real-world attack scenarios.

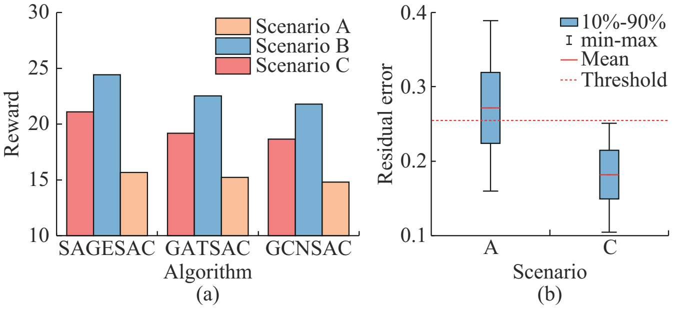


Fig. 5. Rewards of three algorithms in different attack scenarios and perturbation residual statistics for scenarios A and C. (a) Rewards. (b) Perturbation residual statistics.

The impact of state adversarial attacks on system operational performance is further analyzed through the following five cases.

1) Case 1: naive scheduling without considering attack.

2) Case 2: naive scheduling considering attack.

3) Case 3: robust scheduling without considering attack.

4) Case 4: robust scheduling considering attack.

5) Case 5: without control.

In this context, naive scheduling employs the SAGESAC to output scheduling strategies, while robust scheduling utilizes the proposed algorithm for its strategy output. The optimization results in different cases are shown in Table I.

TABLE I
COMPARISON OF OPTIMIZATION RESULTS IN DIFFERENT CASES

Case	Flexibility gain	Operation cost (¥)	The maximum voltage deviation	The maximum average loading rate
1	30.67	615.86	0.0588	0.4889
2	21.66	1511.01	0.0793	0.5311
3	27.38	749.66	0.0679	0.5233
4	28.55	650.86	0.0651	0.5116
5		3746.50	0.0909	0.5594

In Table I, the flexibility gain for Case 2 decreases by 29.37% due to the distortion of original state observations by state adversarial attacks, misleading the SAGESAC. This distortion has the effect of increasing operation costs, raising branch loading rates, and causing potential voltage violations. When comparing Case 4 with Case 1, it can be observed that the proposed algorithm demonstrates effective resistance to state adversarial attacks, with a reduced flexibility gain of only 6.91%. Furthermore, the ability of the proposed algorithm to adapt to unknown external attacks through adversarial training broadens the decision-making experience. For example, Case 3 shows relatively favorable decision outcomes compared with Case 5.

Comparison between Case 3 and Case 4 reveals that the proposed algorithm exhibits a 4.09% decrease in flexibility gain in scenarios without state attacks. This indicates that while adversarial training enhances the robustness of the proposed algorithm against state adversarial attacks, it may lead to overfitting of the neural network policy to adversarial features. Such overfitting results in a slight degradation of algorithmic decision-making performance when processing clean state data under normal (attack-free) conditions.

In summary, the proposed algorithm significantly enhances the robustness against state adversarial attacks while still maintaining a relatively high operational flexibility. Although this algorithm sacrifices some decision-making performance, it is justified because naive DRL algorithms can be severely compromised in the presence of state adversarial attacks.

C. Analysis of Effectiveness of Proposed Flexibility Scheduling Method

To demonstrate the effectiveness of the proposed method, this subsection first analyzes the flexibility gain on the test day and the corresponding changes in node voltage deviation, average branch loading rate, and operation cost. The operation data of PV and load on the test day, with a time resolution of one hour, are shown in Fig. 6. The operational performance of the DN before and after implementing the proposed method is presented in Fig. 7.

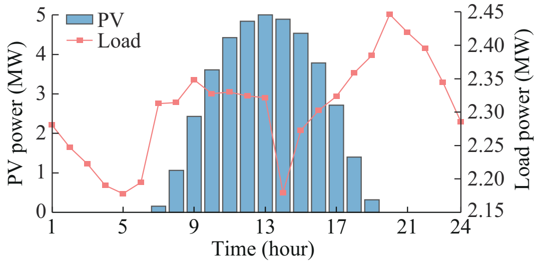


Fig. 6. Operation data of PV and load on test day.

As shown in Fig. 7, the proposed method significantly reduces the maximum voltage deviation, average branch loading rate, and operation cost by enhancing the flexibility gain. In Fig. 7(a), the flexibility gain during periods of 19-24 hours exceeds that during periods of 1-7 hours. This is because ESSs, key devices that support system flexibility, implement an effective charging and discharging strategy to shift the PV output during the day to peak load demand periods at night. This strategy balances power supply and demand, thereby enhancing the operational flexibility of the DN. Notably, the flexibility gain significantly increases during periods of 8-9 hours and 17-18 hours. During these periods, the load fully absorbs the high PV output, thereby reducing the maximum voltage deviations, average loading rates, and operation cost. Consequently, the system has an inherent degree of flexibility, which is further enhanced by implementing effective scheduling strategies.

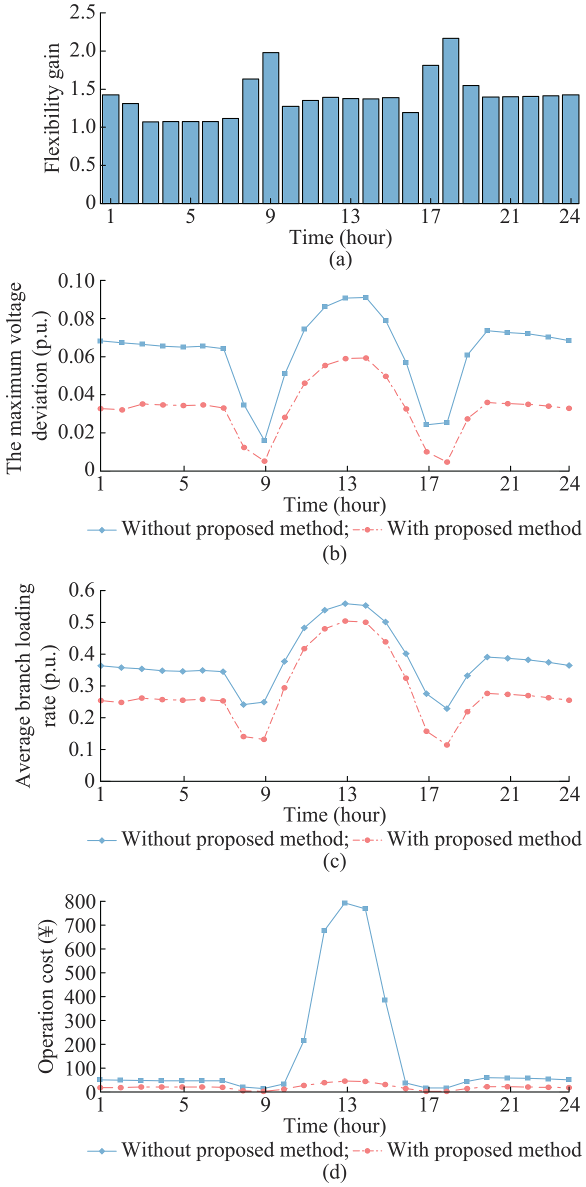


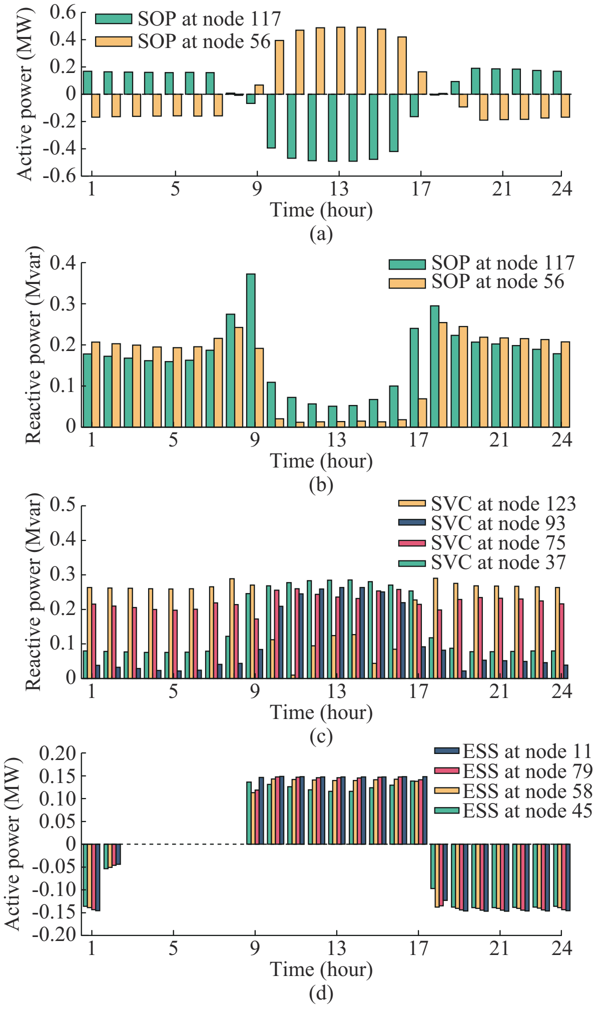
Fig. 7. Operational performance of DN. (a) Flexibility gain. (b) The maximum voltage deviation. (c) Average branch loading rate. (d) Operation cost.

As shown in Fig. 7(b), periods of 11-15 hours exhibit the highest PV output, while periods of 20-24 hours have the highest load demand. In the absence of effective scheduling in the DN, an excess of net power at nodes leads to voltage deviations that exceed acceptable limits, indicating a lack of sufficient node flexibility. Conversely, the proposed method addresses voltage violations by enhancing the flexibility gain, thus endowing the DN with a more adequate node flexibility.

In Fig. 7(c), the average branch loading rate of the system is significantly reduced compared with that before the proposed method is implemented, by an average decrease of 25.1%. This indicates that by enhancing the flexibility gain, the proposed method enables the system to have sufficient branch transfer flexibility, thereby better balancing the power demand and supply across different nodes.

In Fig. 7(d), the operation cost of the system is significantly reduced compared with that before the proposed method is implemented, by an average decrease of 30.1%, especially during period of 11-15 hours when the PV generation is high. During these periods, the DN struggles to accommodate all the PV generation, resulting in curtailments of 0.19 MW, 0.75 MW, 0.89 MW, 0.86 MW, and 0.40 MW, which leads to higher PV curtailment cost. In contrast, the proposed method enhances the flexibility gain, thereby ensuring that the system has sufficient operational safety margins and improving its PV accommodation capacity. Additionally, the proposed method improves node and branch transfer flexibilities while maintaining lower operation cost, thereby comprehensively enhancing the operational level of the DN.

To provide a detailed analysis of how the proposed method enhances the flexibility of the DN, Fig. 8 presents the scheduling strategies of different flexible resources.



(a)

(b)

(c)

(d)

Fig. 8. Scheduling strategies of different controllable resources. (a) Active power of SOPs. (b) Reactive power of SOPs. (c) Reactive power of SVCs. (d) Active power of ESSs.

As shown in Fig. 8, the scheduling strategies for flexibility resources are closely related to the PV penetration rate in the DN. This is because variations in the PV penetration exacerbate the net load volatility, leading to mismatches between PV generation and load demand, which in turn results in an insufficient flexibility [2].

In Fig. 8(a), due to the high number of PV installations at the end of the DN, the abundant PV output significantly exceeds the load demand during periods with high PV penetration. The SOP transfers active power from node 117 to node 56 to smooth power fluctuations as much as possible. During the early morning and nighttime, the SOP transfers a portion of the power required by end loads from node 56 to node 117. This demonstrates that through differentiated power transfer strategies, the SOP effectively addresses the issue of uneven spatial distribution of PV generation and load demand, thereby enhancing the flexibility of the DN.

In Fig. 8(b) and (c), SVCs and SOPs each provide local reactive power compensation through different mechanisms, thereby obviating the necessity for long-distance transmission of reactive power from the resource. These complementary strategies reduce power losses and improve voltage distributions.

In Fig. 8(d), during periods with high PV penetration, ESSs are charged to mitigate the supply-demand imbalance caused by excessive PV output. At night, ESSs are discharged to smooth the high load demand. This demonstrates that by implementing appropriate charging and discharging strategies for ESSs, the proposed method effectively addresses the temporal distribution imbalance between PV generation and load demand, thereby enhancing the flexibility of the DN. It is worth noting that during period of 3-9 hours, all ESSs reach their state of charge limits and cannot continue discharging, resulting in zero power output.

In summary, the proposed method effectively coordinates various controllable resources by maximizing flexibility gain. This alleviates the spatiotemporal mismatch between PV generation and load demand, thereby enhancing the flexibility of the DN.

To further illustrate the comprehensive enhancement of the operational efficiency of the DN through optimizing flexibility gain, three independent objectives, i.e., the maximum voltage deviation, operation cost, and average branch loading rate, are employed to formulate a multi-objective optimization model. The multi-objective particle swarm optimization (MOPSO) is used to generate the Pareto front, and the technique based on the order of preference by similarity to the ideal solution strategy is utilized to determine the optimal compromise solution. The maximum number of iterations is 500, with a population of 100. Optimization results are shown in Table II. The MOPSO results represent the statistical values computed from five independent runs. Time 14 and time 22 are identified as the positive peak and negative peak of the net load on the test day, respectively.

TABLE II
COMPARISON OF OPTIMIZATION RESULTS FROM DIFFERENT MODELS

Time	Model	Operation cost (¥)	The maximum voltage deviation (p.u.)	Average branch loading rate (p.u.)	Test time (s)
14	MOPSO	78.23±8.21	0.062±0.005	0.54±0.04	731.64
	Proposed	60.98	0.059	0.51	0.05
22	MOPSO	25.91±1.62	0.04±0.002	0.31±0.01	629.71
	Proposed	22.14	0.035	0.22	0.05

Table II shows that the operation cost and average branch loading rate achieved with the proposed SA-TZMG model are significantly lower than those obtained using MOPSO. Although the maximum voltage deviation with the proposed SA-TZMG model at time 14 is 3.5% higher than that with the MOPSO, this deviation remains within a safe range. Furthermore, the efficiency of the proposed SA-TZMG model in deriving solutions outperforms that of the MOPSO, as evidenced by a significant decrease in test time. Therefore, it can be concluded that the proposed SA-TZMG model achieves better system performance by enhancing operational flexibility compared with traditional multi-objective optimization models.

C. Impact Analysis of Temperature-dependent Resistance

Weather factors and power flow are the key determinants of line resistance. Thus, we analyze the impact of dynamic weather and system power flow on the flexibility gain of the system over a year. In 2020, the varying ranges of air temperature, wind speed, wind direction, and solar radiation in Jinan, China were −13 - 38 ℃, 0.1-9.3 m/s, 0°-360°, and 0-1002 J/m^2, respectively. The relative error results of the flexibility gain with and without considering weather factors are shown in Fig. 9.

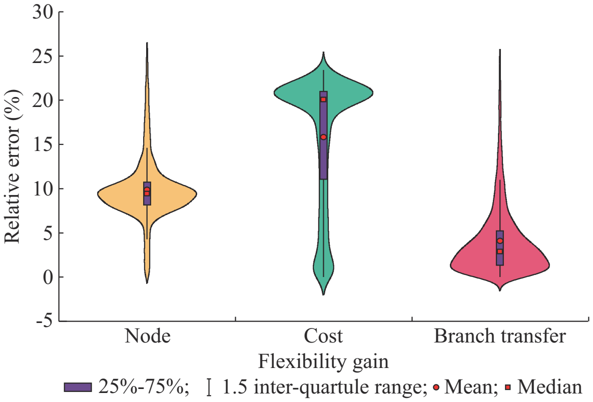


Fig. 9. Relative error results of flexibility gain with and without considering weather factors.

Figure 9 illustrates that considering the temperature-dependent resistance results in notable variations of node flexibility gain, cost flexibility gain, and branch transfer flexibility gain. The relative error ranges for the node flexibility gain, cost flexibility gain, and branch transfer flexibility gain are 0.0129%-29.88%, 0.000412%-26.42%, and 0.000294%-28.59%, respectively. This is primarily due to the change in the thermal equilibrium point of the conductor in response to changing weather conditions, leading to a dynamic transition of both the conductor temperature and resistance towards a new equilibrium point, thereby inducing variations in line resistance. The dynamic line resistance introduces a significant error into the power flow analysis, ultimately affecting the flexibility gain calculation and decision-making processes.

VI. Conclusion

This study introduces a flexibility scheduling method for DNs based on RoGDRL. A mathematical model for flexibility scheduling with temperature-dependent resistance constraints is initially constructed. Based on this, an SA-TZMG model is proposed, which enhances the safety and robustness of the flexibility scheduling method. Finally, a two-stage RoGDRL algorithm based on SAGESAC is designed to achieve robust DRL-based flexibility scheduling, employing an alternate training method through alternating attack and defense. Numerical analysis indicates that:

1) Compared with the traditional DRL-based optimization methods, the proposed method demonstrates stronger robustness against state adversarial attacks.

2) Enhancing flexibility gain can comprehensively improve the operational performance of the DN, thereby better adapting to the large-scale integration of PV.

3) Considering temperature-dependent resistance is crucial in the optimization process to accurately model the dynamic changes of the line resistance, significantly impacting the accuracy of decision-making.

There are several directions for future work. Firstly, additional flexibility analysis indicators could be integrated into the flexibility gain to further enhance flexibility scheduling performance. Additionally, the constructed state-adversarial model could be extended based on the Stackelberg game with incomplete information to address information asymmetry between the attacker and defender, considering attack resource constraints.

References

[1] S. Zhang, S. Ge, H. Liu *et al.*, "Region-based flexibility quantification in distribution systems: an analytical approach considering spatio-temporal coupling," *Applied Energy*, vol. 355, p. 122175, Feb. 2024.

[2] X. Yang, C. Xu, H. He *et al.*, "Flexibility provisions in active distribution networks with uncertainties," *IEEE Transactions on Sustainable Energy*, vol. 12, no. 1, pp. 553-567, Jan. 2021.

[3] M. Rayati, M. Bozorg, R. Cherkaoui *et al.*, "Distributionally robust chance constrained optimization for providing flexibility in an active distribution network," *IEEE Transactions on Smart Grid*, vol. 13, no. 4, pp. 2920-2934, Jul. 2022.

[4] H. Ji, C. Wang, P. Li *et al.*, "Quantified analysis method for operational flexibility of active distribution networks with high penetration of distributed generators," *Applied Energy*, vol. 239, pp. 706-714, Apr. 2019.

[5] J. Jian, P. Li, H. Ji *et al.*, "DLMP-based quantification and analysis method of operational flexibility in flexible distribution networks," *IEEE Transactions on Sustainable Energy*, vol. 13, no. 4, pp. 2353-2369, Oct. 2022.

[6] P. Li, Y. Wang, H. Ji *et al.*, "Operational flexibility of active distribution networks: definition, quantified calculation and application," *International Journal of Electrical Power & Energy Systems*, vol. 119, p. 105872, Jul. 2020.

[7] Y. Su and J. Teh, "Two-stage optimal dispatching of AC/DC hybrid active distribution systems considering network flexibility," *Journal of Modern Power Systems and Clean Energy*, vol. 11, no. 1, pp. 52-65, Jan. 2023.

[8] *IEEE Draft Standard for Calculating the Current-temperature Relationship of Bare Overhead Conductors*, IEEE Std P738-2012 Draft 09 (Revision of IEEE Std 738-2006), pp. 1-67, 2012.

[9] C. Rakpenthai and S. Uatrongjit, "Temperature-dependent unbalanced three-phase optimal power flow based on alternating optimizations," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 3, pp. 3619-3627, Mar. 2024.

[10] Q. Xing, Z. Chen, T. Zhang *et al.*, "Real-time optimal scheduling for active distribution networks: a graph reinforcement learning method," *International Journal of Electrical Power & Energy Systems*, vol. 145, p. 108637, Feb. 2023.

[11] Y. Gao, W. Wang, J. Shi *et al.*, "Batch-constrained reinforcement learn-

ing for dynamic distribution network reconfiguration," *IEEE Transactions on Smart Grid*, vol. 11, no. 6, pp. 5357-5369, Nov. 2020.

[12] L. Zhang, H. Ye, F. Ding *et al*., "Increasing PV hosting capacity with an adjustable hybrid power flow model," *IEEE Transactions on Sustainable Energy*, vol. 14, no. 1, pp. 409-422, Jan. 2023.

[13] Z. Wu, Y. Li, W. Gu *et al*., "Multi-timescale voltage control for distribution system based on multi-agent deep reinforcement learning," *International Journal of Electrical Power and Energy Systems*, vol. 147, p. 108830, May 2023.

[14] D. Cao, J. Zhao, J. Hu *et al*., "Physics-informed graphical representation-enabled deep reinforcement learning for robust distribution system voltage control," *IEEE Transactions on Smart Grid*, vol. 15, no. 1, pp. 233-246, Jan. 2024.

[15] Y. Zhang, M. Yue, J. Wang *et al*., "Multi-agent graph-attention deep reinforcement learning for post-contingency grid emergency voltage control," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 3, pp. 3340-3350, Mar. 2024.

[16] R. Wang, X. Bi, and S. Bu, "Real-time coordination of dynamic network reconfiguration and volt-var control in active distribution network: a graph-aware deep reinforcement learning approach," *IEEE Transactions on Smart Grid*, vol. 15, no. 3, pp. 3288-3302, May 2024.

[17] T. Liu, A. Jiang, J. Zhou *et al*., "GraphSAGE-based dynamic spatial-temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 10, pp. 11210-11224, Oct. 2023.

[18] Y. Zheng, Z. Yan, K. Chen *et al*., "Vulnerability assessment of deep reinforcement learning models for power system topology optimization," *IEEE Transactions on Smart Grid*, vol. 12, no. 4, pp. 3613-3623, Jul. 2021.

[19] I. Ilahi, M. Usama, J. Qadir *et al*., "Challenges and countermeasures for adversarial attacks on deep reinforcement learning," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 2, pp. 90-109, Apr. 2022.

[20] P. Zhao, C. Gu, Y. Ding *et al*., "Cyber-resilience enhancement and protection for uneconomic power dispatch under cyber-attacks," *IEEE Transactions on Power Delivery*, vol. 36, no. 4, pp. 2253-2263, Aug. 2021.

[21] X. Wei, J. Lei, J. Shi *et al*., "A data-driven approach for quantifying and evaluating overloading dependencies among power system branches under load redistribution attacks," *IEEE Transactions on Smart Grid*, vol. 15, no. 4, pp. 4050-4062, Jul. 2024.

[22] L. Zeng, M. Sun, X. Wan *et al*., "Physics-constrained vulnerability assessment of deep reinforcement learning-based SCOPF," *IEEE Transactions on Power Systems*, vol. 38, no. 3, pp. 2690-2704, May 2023.

[23] J. Moos, K. Hansel, H. Abdulsamad *et al*., "Robust reinforcement learning: a review of foundations and recent advances," *Machine Learning and Knowledge Extraction*, vol. 4, no. 1, pp. 276-315, Mar. 2022.

[24] L. Pinto, J. Davidson, R. Sukthankar *et al.*, "Robust adversarial reinforcement learning," in *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia, Aug. 2017, pp. 2817-2826.

[25] Z. Ni and S. Paul, "A multistage game in smart grid security: a reinforcement learning solution," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2684-2695, Sept. 2019.

[26] H. Zhang, H. Chen, C. Xiao *et al.*, "Robust deep reinforcement learning against adversarial perturbations on state observations," in *Proceedings of International Conference on Learning Representation*, Red Hook, USA, Dec. 2020. pp. 21024-21037.

[27] L. Zeng, D. Qiu, and M. Sun, "Resilience enhancement of multi-agent reinforcement learning-based demand response against adversarial attacks," *Applied Energy*, vol. 324, p. 119688, Oct. 2022.

[28] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proceedings of Advances in Neural Information Processing Systems*, Long Beach, USA, Dec. 2017, pp. 1025-1035.

[29] C. Wang, P. Li, and H. Yu, "Development and characteristic analysis of flexibility in smart distribution network," *Automation of Electric Power Systems*, vol. 42, no. 10, pp. 13-21, Sept. 2018.

[30] C. Chen, L. Shen, F. Zou *et al.*, "Towards practical Adam: non-convexity, convergence theory, and mini-batch acceleration," *The Journal of Machine Learning Research*, vol. 23, no. 1, pp. 10411-10457, Jan. 2022.

[31] Z. Yin, S. Wang, and Q. Zhao, "Sequential reconfiguration of unbalanced distribution network with soft open points based on deep reinforcement learning," *Journal of Modern Power Systems and Clean Energy*, vol. 11, no. 1, pp. 107-119, Jan. 2023.

[32] Y. Zhu and D. Zhao, "Online minimax Q network learning for two-player zero-sum Markov games," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 3, pp. 1228-1241, Mar. 2022.

[33] W. Liao, B. Bak-Jensen, J. R. Pillai *et al*., "A review of graph neural networks and their applications in power systems," *Journal of Modern Power Systems and Clean Energy*, vol. 10, no. 2, pp. 345-360, Mar. 2022.

[34] A. Amin and M. Mourshed, "Weather and climate data for energy applications," *Renewable and Sustainable Energy Reviews*, vol. 192, p. 114247, Mar. 2024.

[35] S. Frank, J. Sexauer, and S. Mohagheghi, "Temperature-dependent power flow," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4007-4018, Nov. 2013.

[36] L. S. Shapley, "Stochastic games," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 39, no. 10, pp. 1095-1100, Oct. 1953.

[37] K. Shimizu and E. Aiyoshi, "Necessary conditions for min-max problems and algorithms by a relaxation procedure," *IEEE Transactions on Automatic Control*, vol. 25, no. 1, pp. 62-66, Feb. 1980.

[38] Solcast. (2023, Jul.). Solar API and solar weather forecasting tool. [Online]. Available: https://solcast.com.au

[39] Google Drive. (2024, Apr.). Parameter settings of algorithms and controllable resources. [Online]. Available: Available: https://drive.google.com/file/d/1ZIW7zBRXtc-9yBuuOw5JjWpPsb57oaTY/view? usp=sharing&usp=embed_facebook

Ziyang Yin received the B. S. and M. S. degrees in electrical engineering from Shandong University of Science and Technology, Qingdao, China, in 2018 and 2021, respectively. He is currently pursuing the Ph. D. degree in School of Electrical and Information Engineering, Tianjin University, Tianjin, China. His research interests include distributed generation, machine learning, and smart distribution system.

Shouxiang Wang received the B.S. and M.S. degrees in electrical engineering from Shandong University of Technology, Jinan, China, in 1995 and 1998, respectively, and the Ph.D. degree in electrical engineering from Tianjin University, Tianjin, China, in 2001. He is currently a Professor with the School of Electrical and Information Engineering, and Deputy Director of Key Laboratory of Smart Grid of Ministry of Education, Tianjin University. His research interests include distributed generation, microgrid, and smart distribution system.

Qianyu Zhao received the B.S. and M.S. degrees in electrical engineering and control science and engineering from Tiangong University, Tianjin, China, in 2012 and 2015, respectively, and the Ph.D. degree in electrical engineering from Tianjin University, Tianjin, China, in 2020. She is currently an Associate Professor with the School of Electrical and Information Engineering, Tianjin University. Her research interests include planning, assessment of energy storage and distributed generation, uncertainty analysis of distribution network.