

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript. The published version of the article is available from the relevant publisher.

Adaptive Deep Reinforcement Learning Algorithm for Distribution System Cyber Attack Defense with High Penetration of DERs

Alaa Selim, *Student Member, IEEE*, Junbo Zhao, *Senior Member, IEEE*, Fei Ding, *Senior Member, IEEE*, Fei Miao, *Sung-Yeul Park Member, IEEE*

Abstract—With grid modernization, smart inverters are increasingly used to execute advanced controls for distribution network reliability. However, this also increases the cyber-attack space. This paper focuses on the defense approaches to restore the system to normal operation circumstances in the presence of cyber-attacks. A unique deep reinforcement learning (DRL) method is developed to minimize voltage violations and reduce power losses for impacted feeders. The defense problem is reformulated as a Markov decision-making process to dynamically control DERs while minimizing load shedding. This is achieved via an improved soft actor-critic (SAC)-based DRL algorithm, which can govern DER set points and load-shedding scenarios in discrete and continuous modes via the auto-tune entropy and Gaussian policy features. Numerical comparison results on the modified IEEE 123-node system with other control approaches, such as Volt-VAR (VV), Volt-Watt (VW), and model predictive control (MPC) show that the proposed method can eliminate voltage violations and provide feasible control actions that perform complete mitigation of cyber-threats.

Index Terms—Cyber attack, Active distribution systems, Renewable generation, Deep reinforcement learning.

NOMENCLATURE

γ_{losses}	Weight constant for power losses
γ_{ES}	Weight constant for energy storage dispatch
γ_v	Weight constant for voltage violations
p_t^{atk}, q_t^{atk}	Total active/reactive power after cyber-attack is triggered at time t
$p_t^{losses}, q_t^{losses}$	Total active/reactive power losses for the distribution network per time step
p_t^{ES}, q_t^{ES}	Total active/reactive power dispatched by energy storage elements per time step
p_t^{pv}, q_t^{pv}	Total active/reactive power dispatched by solar PV per time step
$p_t^{DER,K}, q_t^{DER,K}$	Total active/reactive power supplied by number of K distributed energy resource (DER) per time step
p_t^{grid}, q_t^{grid}	Total active/reactive power supplied by the grid network per time step
p_t^{unc}, q_t^{unc}	Uncertainties in total active/reactive power generation per time step

p_t^{ch}, q_t^{ch}	Total active/reactive power charged to the energy storage elements per time step
p_t^{dis}, q_t^{dis}	Total active/reactive power discharged by the energy storage elements per time step
p_t^d, q_t^d	Total active/reactive power demand for all connected loads per time step
$p_t^{d,sh}, q_t^{d,sh}$	Total active/reactive power shedded by controller per time step
α, β, ρ, μ	penalty and reward constants for tuning the reward function
E_t^{ES}	Remaining energy of ES elements computed at each time step
SW_t	Tie switching status, which is determined by DRL agent at each time step t
i	it represents a single node of the distribution system
N	Total number of all nodes of the distribution network
T	Total time of control and optimisation for the studied actions
$S_t^{ES}, S_{min}, S_{max}$	State of charge boundaries of energy storage system
v_i^t	Voltage of single node per time step
V_{vio}	Voltage violations index
ϵ	Random Noise Sample
μ	Mean of Gaussian Policy
ϕ	Concatenated State and Action Representation
σ	Standard Deviation of Gaussian Policy
A_B	Action Bias
A_S	Action Scale
a	Action Representation
b	Bias Vector
P	Probability
Q	Q-Network Branches (Q1, Q2)
s	State Representation
W	Weight Matrix
x	Intermediate Neural Network Activations
y_t	Transformed Activation
LSM	Log Standard Deviation Minimum/Maximum
ReLU	Rectified Linear Unit
SAC	Soft Actor-Critic
DER	Distributed energy resource
DRL	Deep reinforcement learning
SAC	Soft actor critic
VV	Volt-VAR
MPC	Model predictive control
VW	Volt-Watt

This material is based upon work supported by the U.S. Department of Energy's Office of Cybersecurity, Energy Security, and Emergency Response. A. Selim, J. Zhao and S. Park are with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269 USA. F. Ding is with the National Renewable Energy Laboratory. (e-mail: alaa.selim@uconn.edu, junbo@uconn.edu).

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher.
 I. INTRODUCTION

Globally, there have been several notable cases of cyber-physical attacks against electricity infrastructure, such as the Ukraine case [1], [2] in 2015 and 2019 attack on the California power grid [3]. These incidents demonstrate the devastating repercussions that can come from cyber-physical attacks on power systems. To defend against these kinds of attacks, power systems must have robust security measures. The widely studied cyber attacks include false data injection attacks [4], malware attacks [5], physical attacks [6], denial of service attacks [7] and insider attacks [8].

In the context of cyber-physical security, smart attackers can initiate false data injection attacks (FDIA) [9], where a slight change in any of the controllable devices (i.e., smart inverters, smart ring main units and digital relays), can result in disturbing the networking security without being detected by existing defense approaches. In this paper, we propose a learning-based approach for the mitigation of cyber-attacks on connected loads and DERs. DRL is advocated here because of its capability of achieving a good control strategy in defending against cyber-attacks. In [10], DRL is proposed to mitigate stealthy attacks even under the sudden disconnection of any compromised links. A complimentary multi-agent DRL detection algorithm using a deep Q-network algorithm is developed. It mainly focuses on detecting FDIA and labeling it without a mitigation strategy. In [11], the DRL is used to endow the strategy with the adaptability of uncertain cyber-attack scenarios and the ability of real-time decision-making. The DRL is used for determining the optimal reclosing time (their action space). In [12], [13], the DRL-based approach for generating proper VV/VW curves is proposed. The action space aims to control the VV/VW curves instead of set points in mitigating cyber attacks. The resultant policy from that work successfully mitigates adversary induce voltage oscillatory behavior. To illustrate more about the role of DRL in mitigating cyber-attack scenarios, [10] has demonstrated the conceptual model for maintaining cyber security. It is shown that for an offensive type of attack, the use of artificial intelligence to develop the attack strategy cannot be easily detected.

The literature gap on defense techniques, such as those in [14]–[20] can be summarized as follows. First, recent research work is more focused on the detection algorithms for cyber-attacks. Second, in current synthetic cyber-attack models, the reality of cyber-attack scenarios should be accurately modeled to form a probabilistic framework for all impacts of the developed attack. This can be addressed by Markov's decision. Third, defense algorithms should consider more feasible action spaces and not be limited to controlling a certain device. This shall include controlling inverters' operation set points, topology configuration, switching loads, etc. The challenge here is how the proposed defense method is able to figure out all control set points for various resources at the same time that [21]. In our previous work [22], a DRL-based method is proposed to defend against cyber attacks by controlling DER set points as well as minimizing power losses. However, this method could not handle larger-scale systems with high DER penetrations as well as deal with extreme cases, where

proper load shedding is needed. In [23], for the cybersecurity in DERs and smart inverters, the focus is on the security of the physical device layer and the architectural challenges. The paper brings to light realistic examples, such as the threat of malware introduction in DER systems during deployment (identified as "DER.3" in NESCOR threat models), which can compromise system integrity and lead to grid disruptions. Another critical example is the unauthorized alteration of the field DER Energy Management System, potentially causing substantial grid instability or inefficiencies.

Moreover, the Department of Energy (DOE) report [24] highlights specific vulnerabilities unique to DERs, such as the threat of DER ransomware, which can disrupt operations if targeting a substantial portion of DER systems or their aggregators. Similarly, botnet attacks and supply chain compromises pose significant risks, where attackers might target DER hardware and software suppliers, introducing unauthorized access and control. These specialized threats necessitate robust and targeted cybersecurity strategies, focusing on the distinct challenges posed by the decentralized and interconnected nature of DERs within the evolving digital grid infrastructure. Drawing an analogy from the NIST Technical Note 2182 [25], it is equally important for DERs to maintain robust security measures. This ensures that in critical situations, control actions are reliably executed to transition the power system into a secure state. In essence, the security of DERs is a cornerstone in ensuring the reliable and safe operation of the broader power system, especially under challenging conditions. In this study, we conduct a comprehensive comparison between our proposed strategy and existing methods for countering cyber attacks. We focus on critical metrics like DER control, voltage stability, minimization of power losses, and system robustness against various attack scenarios. These metrics, shown in Table I, demonstrate our DRL method's main targets.

TABLE I
 EVALUATION OF RESEARCH PAPERS ON CYBER ATTACK MITIGATION

Criteria	[26]	[27]	[28]	This study
DERs Focus		✓	✓	✓
Multiple Attack Scenarios	✓	✓	✓	✓
Voltage Stability			✓	✓
Power Loss Minimization				✓
Real-Time Response	✓	✓	✓	✓
Learning-Based			✓	✓
Scalability			✓	✓
Adaptability to Threats	✓	✓	✓	✓
Comprehensive Recovery		✓		✓
High Dimension Control				✓
Topology Control				✓

Fig. 1 encapsulates the core scope of our paper, where it shows a cyber attack scenario on DERs and loads, executed by compromising set points through communication protocols and IP addresses. Attackers may inject manipulated Ethernet TCP/IP frames into the Advanced Metering Infrastructure (AMI) system, adhering to the ANSI C12.22 protocol. This action distorts the load profile messages transmitted by smart meter collectors to the Head-end System (HES) and subsequently to the substation controllers, leading to incorrect decisions in DER coordination and control [29] and [30]. The

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher. attack creates a faux overload scenario, prompting the system to inject excessive power. This leads to voltage violations and losses. If unmitigated, it could escalate into a system-wide cascade failure (red table). To demonstrate the attack's impact, we simulate it on the electrical distribution network, triggering it randomly at different times of day and locations. We then solve the power flow using OpenDSS [31] or RSCAD (using Real-time simulators (RTDS)) [32] simulations to assess voltage violation counts and power losses. Finally, our DRL-based control system responds to these events, aiming to restore the system to a stable state (green table) based on its trained policy.

methods show that our method has fewer power losses and can stabilize voltage oscillations.

- Co-simulation model is conducted to solve the power flow with respect to the coupled scenario of the cyber attack and defense algorithm as shown in Fig. 1. This allows having a direct feedback relation between attack and defense methods. Additionally, this co-simulation aims to provide real attack scenarios and set up a detailed framework for mitigating the cyber scenarios.

II. PROBLEM FORMULATION

A. System Model

To investigate the cyber attack scenarios for loads and DERs, the electrical distribution system needs to be modeled and co-simulated to obtain the power flow solution for each attack and defense step. Before modeling the test system, the objectives of the defense algorithm need to be clearly stated, where each control action is handled by the defense algorithm to change state variables of the attacked system from x_t to x_{t+1} through a non-linear relation of f . This current state of x_t is used to compute y_t after solving the power flow through the analytical or the co-simulation model. Firstly, the defense problem objectives can be formulated as an optimization model with DERs setting points and load shedding being the decision variables (u_t). At each step t of the time horizon T , controlling the system to return back to its normal operation aims to minimize a multi-objective function used in our previous work [22] as shown below:

$$\begin{aligned} \min Y = & \sum_{t \in T} [\gamma_{losses} (p_t^{losses})] \\ & + \sum_{t \in T} [\gamma_{bat} (p_t^d - p_{t+1}^d)] \\ & + \sum_{t \in T} \sum_{i \in N} [\gamma_v (v_t^i - V_{nom})^2] \end{aligned} \quad (1)$$

subject to:

$$p_t^{ch} + p_t^{dis} = p_t^{ES}, \quad (2)$$

$$0 \leq p_t^{ch} \leq p_{ES,max}, \quad (3)$$

$$-p_{ES,max} \leq p_t^{dis} \leq 0, \quad (4)$$

$$S_{t+1}^{ES} = S_t^{ES} + \Delta T (\alpha_{ch} p_t^{ch} + \frac{p_t^{dis}}{\alpha_{dis}}), \quad (5)$$

$$S_{min} \leq S_t^{ES} \leq S_{max}, \quad (6)$$

$$P_t^{Total} = \delta P_t^{grid} + \sigma P_t^{DER}, \quad (7)$$

$$P_t^{grid} + P_{i,t}^{ES} + P_{i,t}^{pv} = P_{i,t}^d + P_{i,t}^{uc}, \quad (8)$$

$$P_i^{pv,min} < P_t^{pv} < P_i^{pv,max}, \quad (9)$$

$$q_i^{pv,min} < q_t^{pv} < q_i^{pv,max}, \quad (10)$$

$$p_i^{grid,min} < p_t^{grid} < p_i^{grid,max}, \quad (11)$$

$$q_i^{grid,min} < q_t^{grid} < q_i^{grid,max}, \quad (12)$$

$$-P_i^{ES,max} < P_{i,t}^{ES} < P_i^{ES,max}, \quad (13)$$

$$0.95 \leq |V_{i,t}| \leq 1.05, \forall i \in \mathcal{N}, \quad (14)$$

$$P_{i,t} + jQ_{i,t} = \sum_{k \in \mathcal{N}_i} Y_{ik} |V_i| |V_k| \angle (\theta_{V_i} - \theta_{V_k} - \theta_{Y_{ik}}), \forall i \in \mathcal{N}, \quad (15)$$

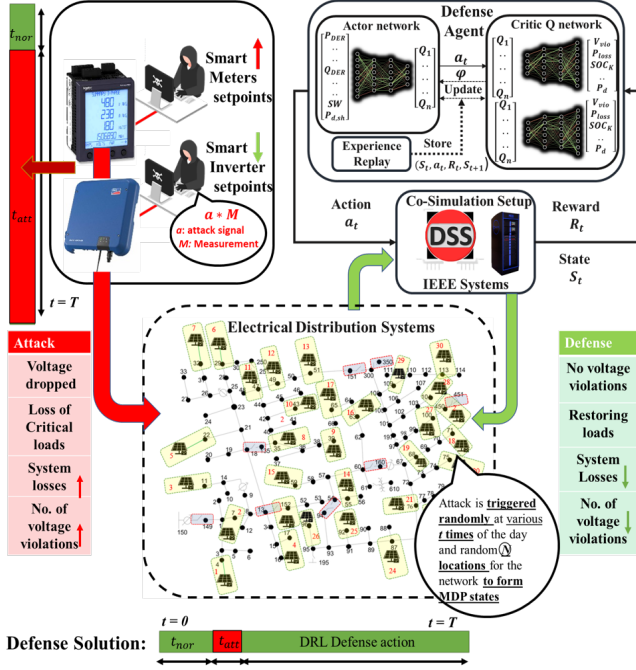


Fig. 1. Cyber attacks defense scenario in the distribution network.

This paper further extends our previous work in [22] to develop an adaptive DRL algorithm able to deal with high DER penetration as well as extreme attack scenarios. It aims to mitigate the cyber attacks induced voltage violations and power losses. The main contributions are:

- A novel defense algorithm is developed to control high-dimension set of actions simultaneously for achieving the optimal mitigation scenario of cyber threats while prioritizing each action based on the cyber attack's numerous conditions. The cyber attack defense problem is reformulated as a Markov decision-making process (MDP). The original soft actor-critic (SAC) method has been improved to include the advanced features of auto-tune entropy and Gaussian policy for controlling DERs' set points and load-shedding scenarios during the several scenarios of cyber attacks.
- The proposed method can effectively allocate DERs' set points and control power flow through the network while avoiding any infeasible switching combinations by checking the connectivity from the source to all the nodes of the system. Comparison results with the VV/VW

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher.

where the first term in the objective function minimizes the total power losses of the network at each time step and the second term is to minimize the load shedding. The third term is to minimize the number of voltage violations. All these objectives are subject to constraints defined from (2) to (14). (15) calculates the power flow using active power $P_{i,t}$ and reactive power $Q_{i,t}$ at node i at time t , considering the admittance Y_{ik} between node i and its neighboring nodes k in \mathcal{N}_i , the voltage magnitudes $|V_i|$ and $|V_k|$ at nodes i and k , and the phase angle differences derived from the voltage angles θ_{V_i} , θ_{V_k} , and the admittance angle $\theta_{Y_{ik}}$, applicable for all nodes i in the network set \mathcal{N} .

B. Cyber Attack Modeling

Cyber attack is modeled based on the FDIA model to cause stealthy changes in the local measurements of certain nodes that are connected either to controllable loads or DERs. In an FDIA attack, the malicious data targets a subset of measurements in an additive manner through random time samples, which makes it quite difficult to be distinguished by system operators. The FDI measurements model is defined as follows:

$$y_k^f = y_k + a_k \cdot 1_\tau(t), \quad (16)$$

where y_k^f is the modified attacked sensor measurement; a_k is the attack signal of FDIA; $1_\tau(t)$ is the indicator function of the attack concerns within the system studied at the attack time τ .

$$1_\tau(t) = \begin{cases} 1 & \text{if } t \geq \tau, \\ 0 & \text{if } t < \tau. \end{cases} \quad (17)$$

The physical systems exchange the control input signals and the sensor measurements through the cyber network. When attackers gain access to the network, they may alter these two types of control-related data, causing a deviation from the physical state. In the case of sensor attack, an attacker modifies the sensor's measurement by adding an attack signal as follows:

$$P_t^{atk} = m \times P_{t,n}^{DER}, \quad \forall t \in T_{att}, n \in N_{att}, \quad (18)$$

$$P_t^{atk} = n \times P_{t,n}^{Load}, \quad \forall t \in T_{att}, n \in N_{att}, \quad (19)$$

$$Q_t^{atk} = k \times Q_{t,n}^{DER}, \quad \forall t \in T_{att}, n \in N_{att}, \quad (20)$$

$$Q_t^{atk} = l \times Q_{t,n}^{Load}, \quad \forall t \in T_{att}, n \in N_{att}, \quad (21)$$

where m , n , k and l are the percentages of power that are increased or decreased by the cyber attacker. These percentages in our case are in the range of 50:150 depending on the scenarios shown in numerical results section. The fluctuations in percentages resulting from cyber attacks can lead to voltage violations at specific nodes of N_{att} and attack time T_{att} , as depicted in Fig. 2

C. Model Predictive Control Method

Model predictive control (MPC) can be used to tackle the cyber attack optimization problem [33], which involves optimizing the voltage magnitudes and power losses in a power system. By simulating the system's response to various control

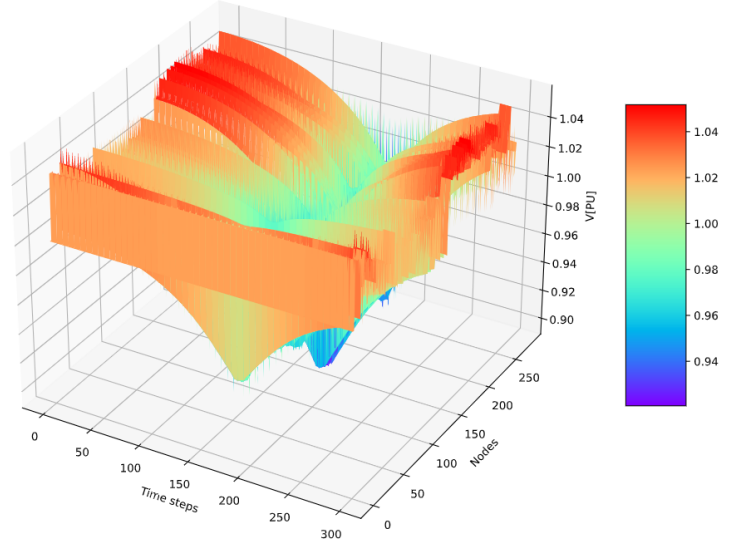


Fig. 2. Voltage profile for IEEE-123 bus system under attack scenario.

operations, the system's future behavior can be obtained. This allows us to evaluate the impacts of various control actions on the cyber attack optimization problem's objectives and constraints. In this paper, we will utilize the MPC in the steady-state mode for solving the cyber attack optimization problem over a time horizon of one day. One possible benefit of employing MPC for cyber attack optimization is that it enables the controller to anticipate and adjust to changing load demand and system conditions due to cyber attacks. MPC can also be used to optimize the coordinated control actions of various devices in the power system. It decides the action similar to DRL for finding the best-coordinated actions for that probabilistic scenario. Additionally, MPC is employed as a validation tool for approximating global optimality in managing power flow within electrical distribution networks as shown in Algorithm 1. Despite MPC's predictive strengths, its inability to guarantee global optimality in nonlinear AC power flow scenarios is a known limitation [34] and [35]. This shortcoming has driven the exploration of alternative methodologies like DRL, which can integrate policy formulation, and address these nonlinear complexities more effectively [36].

III. PROPOSED EXTENDED SAC-BASED DRL CONTROL FOR CYBER ATTACK MITIGATION

The cyber-attack problem is first cast into the Markov decision process (MDP). The environment is configured so that the agent learns how to suppress cyber attacks. This impact can be viewed through voltage violations and power congestion. At each time step, the system experiences different conditions, which are represented by the state vector. Actions are taken at each time step based on the updated state from the last time step and with respect to the boundaries of the system input actions. Consequently, a reward function is formulated

Algorithm 1 Model Predictive Control for Cyber Attack Mitigation

- 1) **Require:** Co-simulation OpenDSS environment, NetworkModel
- 2) Initialize OpenDSS with NetworkModel
- 3) Define optimization weights: $\gamma_v, \gamma_{losses}, \gamma_{bat}$
- 4) Set Ranked buses for DERs deployment
- 5) **For** $t = 1$ **to** $MaxSteps$ **do**
 - a) Load or DERs under cyber attack, causing network violations
 - b) **For** $iter = 1$ **to** $MaxIterations$ **do**
 - i) Define control variables $P_t^{DER,k}, q_t^{DER,K}, SW_t, P_t^{d,sh}$
 - ii) Apply constraints based on AC power flow limits
 - iii) Solve MPC optimization problem
 - A) **Method:** Minimize objective function in (1)
 - iv) **If** Optimal solution found **then**
 - A) Implement control actions in OpenDSS
 - B) Update Min Voltage violations, Min Loss if improvements found
 - c) Report Min Voltage violations, Min Loss for step t

to reflect the effectiveness of control actions at each time step. The key elements for the MDP are shown below.

State Space S_t : the state S_t is used to represent the system status at each time step and is defined as follows:

$$S_t = [E_t^{ES}, P_t^d, V_{vio}]. \quad (22)$$

Actions Space a_t : the set of available actions at each time step and determines the continuous operational set points of the DERs based on the available power that can be generated and energy storage limits. Actions also include discrete operation of load shedding and tie switching, where load shedding is controlled in discrete steps based on the μ factor, and tie switches are switched on/off to form a combination pattern at each time step that should be feasible. Formally, we have

$$a_t = [P_t^{DER,k}, q_t^{DER,K}, SW_t, P_t^{d,sh}]. \quad (23)$$

Reward Function: the reward function represents the multi-objective function we would like to maximize. The reward in this problem is related to minimizing the number of shedding loads and power losses in the network while maintaining energy reserve in energy storage and penalizing the voltage violation at each node for all time steps. These four objectives are weighted in the reward formulation with respect to their impacts on the system performance. Our target is to maximize the reward function as defined below:

$$\text{Maximize } \sum_{t=0}^T r(s_t, a_t), \quad (24)$$

$$r(s_t, a_t) = \sum_{t=0}^T (-\alpha(P_{loss,t}) - \beta(V_{vio}) \quad (25)$$

$$-\rho(V^{Nor} - V_{i,t}) - \mu(P_{i,t-1}^d - P_{i,t}^d)).$$

A. Original SAC Algorithm

The MDP in this paper can be solved by the SAC algorithm [37], where the function approximators for both the soft Q-function and policy are leveraged. A parameterized soft Q-function and a tractable policy will be used. The parameters of these networks are θ and ϕ . For example, the soft Q-function can be modeled as expressive neural networks, and the policy as a Gaussian with mean and covariance is given by the neural network. SAC has been shown to have a greater advantage to handle stochastic models of uncertain and intermittent DERs via the entropy term:

$$\pi = \arg \max E \left[\sum_{t=0}^{\infty} \gamma^t (R(s_t, a_t) + \alpha H(\pi(\cdot | s_t))) \right], \quad (26)$$

where γ is a future discount coefficient; $R(s_t, a_t)$ represents the expected discount reward with state s_t with actions taken following the actor policy π ; $H(\pi(\cdot | s_t))$ is the entropy term; $\alpha > 0$ is the trade-off coefficient. Adding entropy term is to encourage the agent to explore more possibilities in action space. Inside SAC, there are two neural networks known as actor and critic networks, where the actor-network is designed to find the best action corresponding to the current state and the critic network is designed to find the Q-value of the executed action in the current state. The critic network computes target for the Q function as:

$$y(s_t, r_t, s_{t+1}) = r + \gamma (Q(s_{t+1}, a_{t+1}) - \alpha \log \pi_{\theta}(a_{t+1} | s_{t+1})). \quad (27)$$

In our problem, SAC is mainly used to try all possible combinations of actions at that current step to suppress the voltage violations. The advantage of SAC appears here in dealing with the extensive probabilistic framework of state variables, which is quite challenging to be solved by conventional control methods. As the agent works on finding the best reward values for all probabilistic cyber attack situations, the application of SAC in solving the cyber attack optimization problem broadens the solution space and provides more global optimal solutions.

B. Improved SAC Algorithm

This improved version of SAC focuses on the idea of incorporating the ability to switch between Gaussian and deterministic policies within the SAC framework, thereby adding an extra layer of adaptability to the algorithm. By providing the flexibility to select the most suitable policy type, the proposed approach can enhance the algorithm's performance across various domains. The Gaussian policy, with its inherent stochasticity, can encourage exploration during the learning process, while the deterministic policy can offer more stability in the learned behavior. Investigating the trade-offs and potential synergies between these two policy types could lead to more robust and efficient SAC agents. In this paper, we propose a novel SAC implementation that leverages both Q-network and Gaussian policy architectures and aligns with the model architecture discussed in [38]. The Q-network architecture consists of two separate branches, Q1 and Q2,

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher.

each with an input layer and two hidden layers. The input layer receives the concatenated state and action representations, denoted as ϕ . The hidden layers employ ReLU activation functions. Mathematically, the Q-network forward pass can be described as follows:

- $\phi = \text{concat}(s, a)$: Concatenation of the state s and action a to form the input ϕ .
- $x_1 = \text{ReLU}(W_1 \cdot \phi + b_1)$: First layer of transformation with weights W_1 , bias b_1 , and ReLU activation function.
- $x_1 = \text{ReLU}(W_2 \cdot x_1 + b_2)$: Second layer with weights W_2 , bias b_2 , and ReLU activation.
- $Q_1 = W_3 \cdot x_1 + b_3$: Output of the first Q-value branch with weights W_3 and bias b_3 .

For the second Q-value branch, similar steps are followed:

- $x_2 = \text{ReLU}(W_4 \cdot \phi + b_4)$: First layer with weights W_4 and bias b_4 .
- $x_2 = \text{ReLU}(W_5 \cdot x_2 + b_5)$: Second layer with weights W_5 and bias b_5 .
- $Q_2 = W_6 \cdot x_2 + b_6$: Output of the second Q-value branch with weights W_6 and bias b_6 .

The Gaussian policy consists of a two-layer neural network, followed by separate layers for calculating the mean and log-standard deviation of the action distribution. These layers are then clamped to predefined limits, ensuring stable learning. The forward pass of the Gaussian policy can be expressed as:

- $x = \text{ReLU}(W_7 \cdot s + b_7)$: This is the first transformation layer applying ReLU activation function on the product of weights W_7 and state s added to bias b_7 .
- $x = \text{ReLU}(W_8 \cdot x + b_8)$: The second transformation layer utilizes ReLU activation and operates on the product of weights W_8 and the output of the previous layer together with bias b_8 .
- $\mu = W_9 \cdot x + b_9$: The mean (μ) of the action distribution is calculated as the product of weights W_9 and the output of the second layer plus bias b_9 .
- $\log(\sigma) = \text{clamp}(W_{10} \cdot x + b_{10}, \text{LSM})$: The logarithm of the standard deviation (σ) of the action distribution is clamped between lower and upper limits (denoted here as LSM) after applying weights W_{10} and bias b_{10} to the output of the second layer.

The Gaussian policy in the algorithm generates actions using the reparameterization trick and a tanh transformation to maintain action limits. This process is detailed in the following equations:

- $x_t = \mu + \sigma \cdot \epsilon$: A sampled action is created by adding noise scaled by σ to the mean action μ . ϵ is the noise factor.
- $y_t = \tanh(x_t)$: The tanh function normalizes the sampled action to ensure it falls within action bounds.
- $a = y_t \cdot A_S + A_B$: The scaled action a is derived by scaling the normalized action y_t and adjusting it by a baseline A_B .
- $\log(p) = \log\left(\frac{\exp\left(-\frac{(x_t - \mu)^2}{2\sigma^2}\right)}{\sigma\sqrt{2\pi}}\right) - \log(A_S \cdot (1 - y_t^2) + \epsilon)$: This equation calculates the log probability of the sampled action, factoring in the Gaussian distribution and the tanh normalization.

- $\mu_d = \tanh(\mu) \cdot A_S + A_B$: The deterministic mean action μ_d is calculated by normalizing the mean μ with tanh, scaling, and baseline adjustment.

These steps ensure that the actions chosen by the policy are both random and constrained within the specified action space.

This section introduces the Soft Actor-Critic (SAC) implementation, a DRL algorithm combining Q-networks and Gaussian policy architectures. This proposed SAC implementation offers a robust and flexible framework for improved performance and exploration. The implementation of the SAC algorithm is summarized as follows:

- **Initialization:** The class is initialized with input dimensions, action space, and other hyper-parameters, such as learning rate, gamma, tau, and alpha. It creates instances of the Q-network and Gaussian Policy classes and their respective Adam optimizers. Additionally, it creates a target Q-network and initializes it with the weights of the Q-network.
- **Action Selection:** This method takes a state as input and uses the Gaussian policy to either sample an action (during training) or return a deterministic action (during evaluation).
- **Parameter Update:** We update here the parameters of the Q-network, policy network, and (optionally) the entropy coefficient alpha. This method takes a memory buffer and performs the following steps:
 - Sample a batch of transitions from the memory buffer.
 - Compute the target Q-value for the next state using the target Q-network and the reparameterized next state action from the policy network.
 - * $a_{t+1}, \log p(a_{t+1}|s_{t+1}) = \text{policy sample}(s_{t+1})$: This step samples a new action a_{t+1} and its log probability from the policy network, given the next state s_{t+1} .
 - * $Q_{1,t+1}, Q_{2,t+1} = \text{critic target}(s_{t+1}, a_{t+1})$: The target critic network estimates the Q-values $Q_{1,t+1}$ and $Q_{2,t+1}$ for the next state and action.
 - * $Q_{\text{target}} = r_t + \gamma(\min(Q_{1,t+1}, Q_{2,t+1}) - \alpha \log p(a_{t+1}|s_{t+1}))$: The target Q-value is calculated using the reward r_t , discount factor γ , the minimum of the estimated Q-values, and the entropy term $\alpha \log p(a_{t+1}|s_{t+1})$ to encourage exploration.

This process helps in evaluating the future value of actions, guiding the policy network towards more rewarding decisions.

- Update the Q-network parameters by minimizing the mean squared error between the predicted Q-values and the target Q-values.
 - * $Q_1, Q_2 = \text{critic}(s_t, a_t)$: The critic network computes two Q-value predictions, Q_1 and Q_2 , for the current state-action pair (s_t, a_t) .
 - * $\mathcal{L}_{Q_1} = \text{MSE}(Q_1, Q_{\text{target}})$: The mean squared error (MSE) loss for Q_1 is calculated by comparing it with the target Q-value.
 - * $\mathcal{L}_{Q_2} = \text{MSE}(Q_2, Q_{\text{target}})$: Similarly, MSE loss for Q_2 is calculated.

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher.

* $\mathcal{L}_Q = \mathcal{L}_{Q_1} + \mathcal{L}_{Q_2}$: The total loss for the Q-network is the sum of the individual losses for Q_1 and Q_2 .

This process refines the critic network, enabling it to better approximate the true Q-values, which guides the policy network's learning.

– Update the policy network parameters by maximizing the expected return minus the expected entropy of the policy.

* $a_t, \log p(a_t|s_t) = \text{policy sample}(s_t)$: The policy network samples an action a_t for the current state s_t , and computes the log probability of this action.

* $Q_{1_t}, Q_{2_t} = \text{critic}(s_t, a_t)$: The critic network evaluates the action by providing Q-values Q_{1_t} and Q_{2_t} .

* $\mathcal{L}_\pi = \text{mean}(\alpha \log p(a_t|s_t) - \min(Q_{1_t}, Q_{2_t}))$: The loss for the policy network is computed as the mean of the difference between the scaled log probability of the chosen action and the minimum of the two Q-values.

This process aims to refine the policy network by maximizing the expected return (as indicated by the Q-values) while maintaining a balance with the policy's entropy, promoting exploration.

– Update the entropy coefficient alpha by minimizing the difference between the log-probability of actions and the target entropy.

$$\mathcal{L}_\alpha = -\text{mean}(\log \alpha (\log p(a_t|s_t) + H_{\text{target}})) \quad (28)$$

The Replay Memory is vital in DRL algorithms for managing past agent experiences as developed in [39]. These experiences are structured as tuples $(s_t, a_t, r_t, s_{t+1}, d_t)$, symbolizing the state, action, reward, next state, and done flag. Key functionalities include i) A cyclic buffer with predefined capacity, allowing for the storage and cyclic update of experiences; ii) The push operation, which introduces new experiences at the current index, ensuring a continuous update with new data; iii) The sample function, which randomly selects experiences, dividing them into separate arrays for each element of the tuple and iv) Save buffer and load buffer methods, which assist in preserving and retrieving the state of the buffer, beneficial for ongoing training sessions.

Let $B(t) = \{(s_i, a_i, r_i, s'_i, d_i) | 1 \leq i \leq n\}$ denote the buffer at time t . The push operation can be formally represented as:

$$B(t+1) = \begin{cases} B(t) \cup (s, a, r, s', d), \\ B(t) \setminus \{(s_1, a_1, r_1, s'_1, d_1)\} \cup (s, a, r, s', d), \end{cases} \begin{cases} \text{if } |B(t+1)| \leq \text{capacity}, \\ \text{if } |B(t+1)| > \text{capacity}. \end{cases} \quad (29)$$

The sample operation randomly retrieves a batch of size k from the buffer, represented as:

$$\{(s'_j, a'_j, r'_j, s'_{j+1}, d'_j) | j \in I\}, \quad (30)$$

where I is a set of k unique integers randomly selected from $\{1, 2, \dots, n\}$. This particular buffer's cyclical and random sampling capabilities contribute to more efficient exploration and robust learning by preventing overfitting to recent experiences and maintaining a diverse set of past interactions.

In the original SAC algorithm, the entropy term is fixed to a constant value. However, some variations of SAC incorporate an automatic entropy tuning mechanism, which allows the entropy term to be adjusted dynamically. The equation below is used for the optimization of the temperature or entropy coefficient (α) in the SAC algorithm. The purpose of this optimization process is to automatically tune the entropy coefficient to strike a balance between exploration and exploitation, depending on the task and desired target entropy.

$$\min_{\alpha} \mathbb{E}_{(s,a) \sim B} \left[\alpha \log \frac{\pi_{\theta}(a|s)}{p(a|s)} - \alpha \bar{H} \right]. \quad (31)$$

The expectation term is taken over state-action pairs (s, a) sampled from the replay buffer B . The optimization process seeks to find the optimal value for the temperature α that minimizes the difference between the log-likelihood ratio of the policy distribution $\pi_{\theta}(a|s)$ and the target distribution $p(a|s)$, weighted by the temperature parameter α , and the product of the temperature parameter α and the desired target entropy \bar{H} . In essence, the optimization algorithm adjusts the temperature α to achieve a balance between exploration (higher entropy) and exploitation (lower entropy) by minimizing the difference between the actual entropy and the target entropy. When the entropy is too high, the algorithm will decrease the temperature α to encourage more exploitation; when the entropy is too low, the algorithm will increase the temperature α to encourage more exploration and will also a reduced sensitivity to hyperparameters. Another improvement feature in this SAC algorithm is the Gaussian policies, which can reflect a wide range of action distributions, allowing the DRL agent to explore and adapt to many settings and tasks. Gaussian rules can also be used to reflect uncertainty or risk in decision-making, allowing the DRL agent to balance exploration and exploitation. Moreover, Gaussian policies can be more computationally efficient than other types of policies, such as categorical or discrete policies, which can be advantageous for high-dimensional or computationally heavy DRL tasks, the case in the cyber-attack problem.

For topology change [40], given a set of n switches, each of which can be in a "Close" (1) or "Open" (0) state, we represent each combination of switch states as a binary number. Let $S = [s_1, s_2, \dots, s_n]$ represent the state of each switch, where s_i is the state of the i^{th} switch. We then convert this binary representation into a decimal index using the following formula:

$$\text{Index} = \sum_{i=1}^n b_i \times 2^{n-i}. \quad (32)$$

The binary equivalent of each switch state is denoted as b_i . The DRL algorithm uses the SW_t action to dynamically control these switch states in real-time, systematically exploring switch state combinations to minimize power losses. Each SW_t action corresponds to a specific index, enabling the DRL to efficiently determine the optimal switch configuration for power loss reduction while adapting to network changes. Algorithm 2 presents the implementation of our improved SAC approach for cyber attack mitigation in distribution networks.

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher.

Algorithm 2 Improved SAC algorithm for cyber attack defense in distribution system

Perform load/DER attack using m and l to nodes identified
Perform the improved SAC training phase

Initialize Q network parameters ω_0 , policy network parameters ϕ_0 , and alpha α_0 . Empty replay buffer \mathcal{D} , learning rates $\lambda_\omega, \lambda_\phi, \lambda_\alpha$, batch size BS , start step t_S Collect a buffer \mathcal{B}_0 of initial states s_0 from environment resets.

```

for  $0 : t_s : T_{max}$  do
     $a_t \sim \pi_\phi(a_t | s_t) \rightarrow$  record  $P_t^{DER,k}, SW_t$ 
     $s_{t+1} \sim p(s_{t+1} | s_t, a_t) \rightarrow$  observe  $P_t^{loss}, V_t^i$ 
     $B \leftarrow B \cup \{(s_t, a_t, r(s_t, a_t), s_{t+1})\} \rightarrow$ 
    conduct voltage matrix with corresponding rewards
    Sample a batch of transitions
     $BS = \{(s, a, r(s, a), s')\}_{i=1}^{BS}$  from  $B$ .
     $\rightarrow$  for best convergence towards the optimal policy.
    Record voltage matrix at highest  $r(s_t, a_t)$  value end
    Compare highest reward voltage matrix with the original
    attacked one
    if ( $V_t^t > 1.05$  or  $V_t^t < 0.95$ ) then
        Maximize the total number of episodes
        Hyper-tune learning rates  $\lambda_\omega, \lambda_\phi, \lambda_\alpha$ , batch size  $BS$ 
        Increase the maximum limit of  $a_t$  space vector for
        larger DER set points
    end
    else
        End SAC training and save its policy parameters
    end

```

violation has been observed during the normal operation. In addition, 8 tie switches have been added to the network and no sectionalizing sections are considered for operation. The learning environment is designed according to OpenAI Gym, which is a common interfacing library to define the DRL environment for the agent. The implementation of the SAC algorithm in our study is based on the PyTorch framework. We have carefully configured the SAC algorithm, ensuring that both the actor and critic networks are optimally structured based on the improved structure in Section III.B. The detailed configuration of these networks, along with other critical hyperparameters of the SAC algorithm, are comprehensively listed in Table III. The target network is updated by $\tau = 0.005$ and a random process is applied for better exploration with $\alpha = 0.1$, $\beta = 0.1$ and $\rho = 0.1$. The DRL training is implemented on a laptop computer with 3.6GHZ Intel i7 processor and 32.0 GB RAM. The existing Volt-Watt, Volt-VAR and MPC approaches are implemented using OpenDSS. MPC is implemented using the CVXPY library [41]. We are using the Splitting Conic Solver (SCS) in conjunction with OpenDSS in this CVXPY-based framework, which is appropriate for handling linear optimization tasks within a non-linear power system context. In this MPC setup, we define the control and prediction horizon to be 1 and 290 respectively and the number of control steps is 62, the same as DRL. Note that there are 60 control actions used for controlling DERs and two control actions for tie switches and load-shedding respectively. The attacks are introduced into the environment through Load and DERs' settings change to simulate four different scenarios shown in Table IV, where the DRL agent is being trained for a window of 290 time-steps (i.e., time-step = 5 mins, the mean time of the attack is 24 hrs). During this window, DRL is exploring the best policy to deal with all DERs set points and tie-switches, that eliminate voltage violations and network power losses after solving the power flow. The implementation of the proposed algorithm can be found in **Algorithm. 2**.

The analysis of training curves for the standard and proposed SAC models, shown in Fig. 4, demonstrates a significant advancement in reward optimization with our proposed SAC. The standard SAC often plateaued at suboptimal early-stage solutions, particularly in mitigating voltage violations. By contrast, our proposed SAC variant exhibited enhanced progression and effectiveness in overcoming these limitations, as evidenced by its superior performance in the training assessments. For the Volt-VAR function, as detailed in [42], the monitored voltage in per unit (p.u.) is applied within the Volt-VAR curve to determine the desired reactive power value in p.u. Similarly, in the smart inverter volt-watt function, the monitored voltage in p.u. is utilized in the volt-watt curve to calculate the active power limit value in p.u.

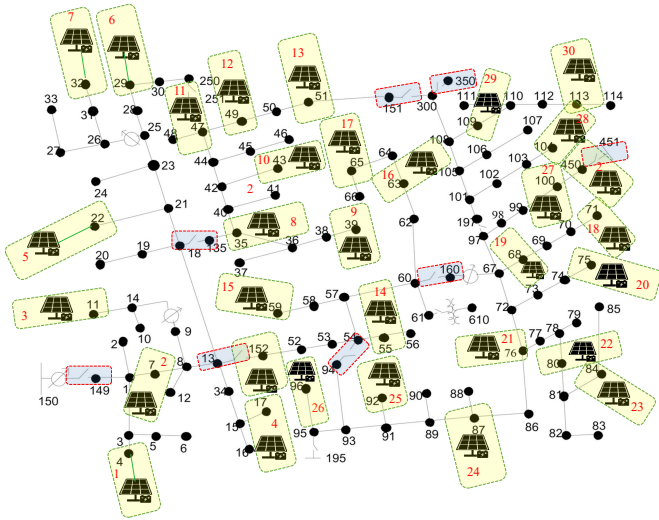


Fig. 3. Modified IEEE 123-node system with tie switches and DERs.

IV. NUMERICAL RESULTS

A modified IEEE 123-node system with three-phase loads and 60 DERs (i.e., 30 ES units and 30 PVs with installed capacities of 1000 kW each unit), is used for testing as shown in Fig. 3 and Table II. The system is modeled using OpenDSS and is configured to be in the grid-connected mode. At the first solution evaluated using OpenDSS, no voltage

A. Scenario 1: Load Altering Attack

This section evaluates the effectiveness of the proposed approach for load-altering attacks. Following the settings shown in Table V for initiating the load-altering attacks, FDIA takes place inside the local measurements of the loads'

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher.

TABLE II
PARAMETERS OF CONTROLLABLE DEVICES

Type	Capacity (kW)	Bus Location
Solar PV	1000	4,7,11,17,22,29,32,35,39,43,47,49,51,55,59,63,65,68,71,75,76,80,84,87,92,96,100,104,109,113
ES	1000	4,7,11,17,22,29,32,35,39,43,47,49,51,55,59,63,65,68,71,75,76,80,84,87,92,96,100,104,109,113
Tie switches	-	150-149, 13-152, 18-135, 60-160, 97-197, 151-300, 54-94, 61-610

TABLE III
DRL-PARAMETERS

Parameters	Value
α	0.1
β	0.1
ρ	0.1
τ	0.005
Batch size	256
γ	0.99
policy	Gaussian
Automatic entropy tuning	True
Learning rate	0.0005
Coefficient of target network's soft update	0.001
Number of hidden layers	2
Number of nodes	[64,64]
Activation function	ReLU
Maximum number of episodes	8000
Replay buffer size	1000000

zone identified in Fig. 5. In the co-simulation mode using OpenDSS, the defense algorithms are applied to the studied distribution system, and power flow results are used to update the state and output variables of the system. Thanks to OpenDSS, we smoothly performed Volt-VAR and Volt-Watt control algorithms using the internal functions of the program after defining the operational curves for both algorithms. During that generated attack, n and l are changed to 150% to randomly attack some loads in that region. This approach is meant to simulate a real attack that randomly infects smart meters within the clustered zone of the distribution network. Additionally, this attack is performed at random times during the test, to make the defense algorithm more robust to all kinds of loads within the available generation resources. Table. VI deduces an indication of the proposed methods for solving the cyber attack optimization problem under various attacking conditions. Results show that DRL and MPC can regulate the voltage within the standard limits, however, Volt-VAR and Volt-Watt controls fail to regulate the voltage after the attack happened. The performance of each method is then verified using a synthetic random attack case as shown in Fig. 6 to fairly compare the response of each defense algorithm and validate its reliability. It can be observed that the proposed DRL-based approach is able to successfully regulate the voltage within the security limit and has the least power losses among all approaches. As for topology configuration, it selects to close 7 out of 8 tie switches and no load shedding is executed at that time. Also, the total number of voltage violations is reduced to a minimum of 6 violations from 79 original

violations. These results are based on the snap mode solution after solving the power flow in OpenDSS. Volt-Watt and Volt-VAR approaches have more voltage violation issues and fail to regulate the voltage as planned and the power loss is 10% higher. The MPC-based method has fewer voltage violations than Volt-VAR and Volt-Watt control and its power loss is smaller than Volt-Watt and Volt-VAR approaches. It is worth mentioning that for the MPC approach, a reduction in voltage violations is observed by changing the set points of DERs. However, upon tuning its optimization weights and increasing the iterations, as delineated in Algorithm 1, it is found that the MPC fails to further reduce the voltage violations for the studied time steps beyond a certain limit. Even if the conventional settings of Volt-VAR and Volt-Watt approaches are adjusted through control modes in OpenDSS, they are still insufficient for mitigating voltage violations. This limitation underscores the necessity for a more advanced solution, thus strongly motivating the use of DRL to control these settings.

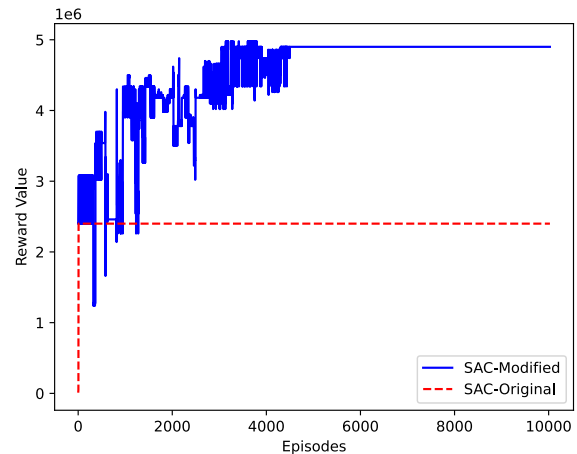


Fig. 4. Learning curve for the proposed improved SAC agent.

B. Scenario 2: DER Setting Point Attacks

This section evaluates the effectiveness of the proposed approach for DER-altering attacks. We follow the settings shown in Table VII for observing the standard scenarios of the DER-altering attacks, where the generation is reduced to cause power imbalance and violates the system security. It is found that attacks on DER nodes are sensitive to the attack timing and at the conditions of high penetration of the solar PVs percentage, it may cause many voltage violations as shown in Table VIII. Similar to what we did in the load attack scenario, the performance of each method is tested for a random DER-attack of 10 DERs at noon time and the results are shown in Fig. 7. It can be observed that the proposed DRL-based approach is able to successfully regulate the voltage within the security limit and has the least power losses among all approaches and also outperforms other conventional control methods. Based on the policy obtained by the DRL agent, it is able to figure out the right pattern of DER set points to

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript. The published version of the article is available from the relevant publisher.

TABLE IV
CYBER ATTACK PROPOSED CASES

Scenario	Attack Type	Attack Destination	Remarks
1	FDIA	Smart inverters of DERs	Changing m and k to be ranging from 50 to 90 % for n^{th} DERs
2	FDIA	Smart meters of Connected loads	Changing n and l to be ranging from 120 to 150 % for n^{th} Loads
3	FDIA	DERs and Connected loads	Changing m, k, l and n to ranging from 50 to 150%
4	FDIA	Connected loads	Changing m and l to be from 150% to 200% (resulting in load shedding scenarios)

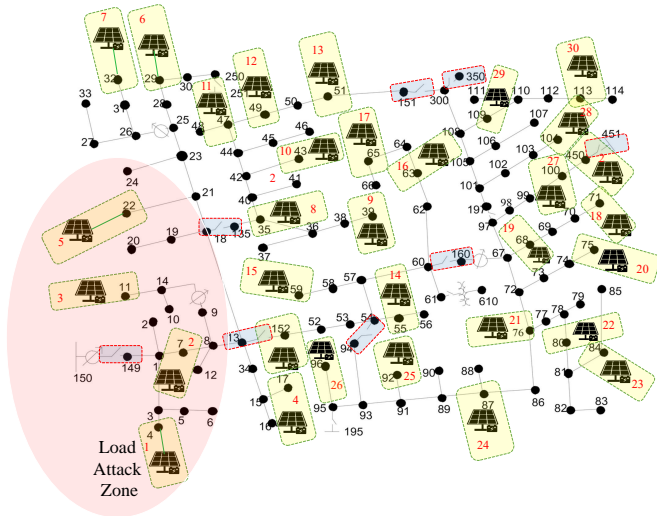


Fig. 5. Load attacks scenario in IEEE 123-node system.

TABLE V
LOAD ALTERING ATTACK SETTINGS.

Load Change(%)	Vmin	Vmax	Plosses(%)
0	0.9619	1.0486	7.719
20	0.9394	1.0486	14.62
40	0.9392	1.0486	14.65
60	0.9390	1.0486	15.1
100	0.9389	1.0486	15.12
200	0.9366	1.0485	15.37

TABLE VI
STATISTICS OF THE COMPARISON RESULTS UNDER LOAD ALTERING ATTACKS.

Control method	Vmin	Vmax	Plosses(kW)	Plosses(%)
Non-control	0.939	1.049	585.421	14.620
Volt-Watt	0.939	1.048	593.520	15.100
Volt-VAR	0.939	1.049	585.421	14.62
DRL- tie switch	0.981	1.048	62.366	3.185
DRL-W/O tie switch	0.981	1.048	62.336	3.1845
MPC	0.965	1.060	407.858	10.950

execute during the timing of cyber attacks at high penetration of solar PVs and even when it was tested for slightly different percentages of penetration to develop a robust response against this type of attack.

C. Scenario 3: Combined Attacks

The third scenario tested is considered the worst case, in which the attack targets both DERs and connected loads at the same time. The attack is designed to reduce generation by 50% for 25 randomly selected loads and to increase overloading

TABLE VII
DER ALTERING ATTACK SETTINGS.

DER Change(%)	Vmin	Vmax	Plosses(%)
0	0.96188	1.0486	7.719
-10	0.94941	1.0486	32.67
-20	0.93921	1.0486	33.79
-40	0.92899	1.0486	45.21

TABLE VIII
STATISTICS OF THE COMPARISON RESULTS UNDER DER ALTERING ATTACKS.

Control method	Vmin	Vmax	Plosses(kW)	Plosses (%)
Non-control	1.004	1.213	619.629	20.640
Volt-Watt	1.004	1.213	619.562	16.583
Volt-VAR	0.982	1.209	601.545	15.620
DRL- tie switch	0.952	1.050	432.897	10.892
DRL-W/O tie switch	0.952	1.050	432.897	10.892
MPC	0.973	1.080	538.900	10.950

by 50% for 10 randomly selected DERs. The proposed DRL is trained on that scenario for the attack across all time horizons; note that the attack's timing must be considered. In the previous cases, the DRL and MPC mitigated voltage violations to achieve only 8 and 35 violations, respectively, as shown in Fig. 8. Volt-Var and Volt-Watt, on the other hand, fail to take the necessary steps to improve system security following the attack.

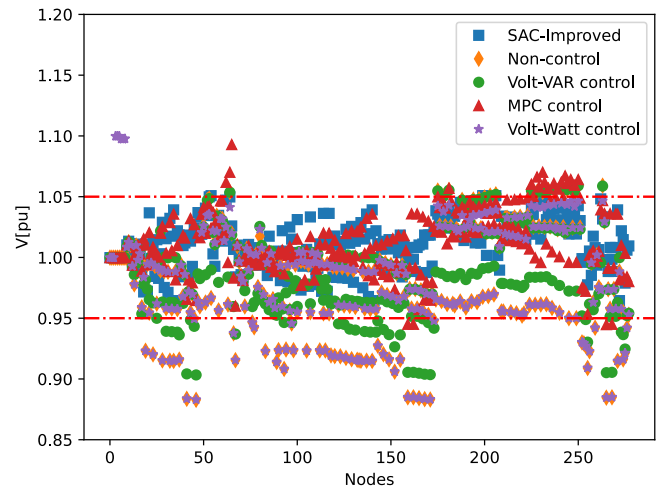


Fig. 6. Comparison results for different cyber defense algorithms-Scenario 1.

From the results shown in the table, it is found that the tie switch control has negligible impact on voltage violations.

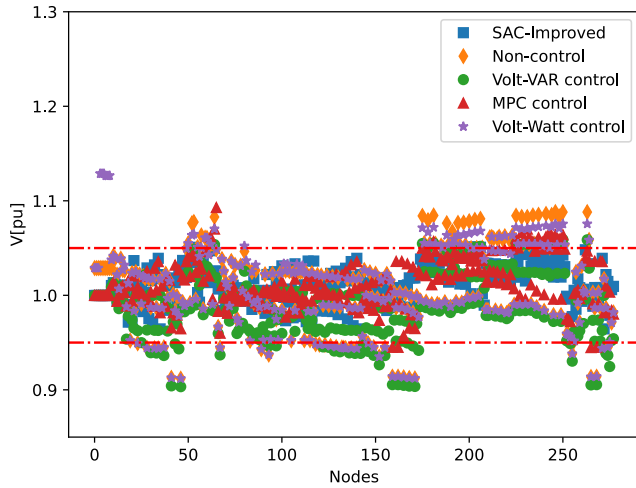


Fig. 7. Comparison results for different cyber defense algorithms-Scenario 2.

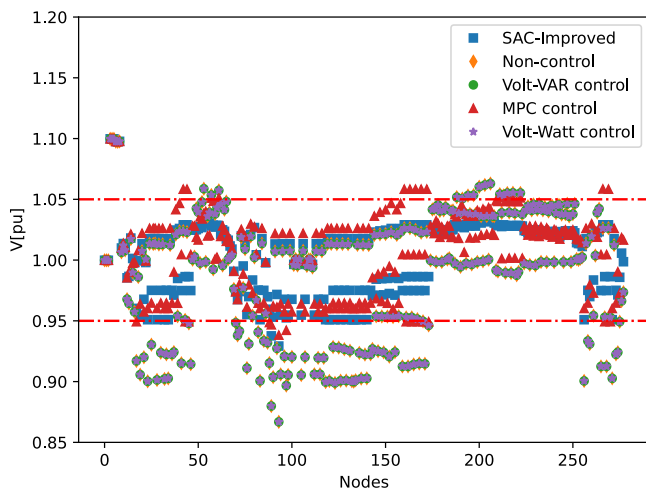


Fig. 8. Comparison results for different cyber defense algorithms-Scenario 3.

Consequently, we utilize a single snapshot solution to analyze the topology change and assess the impact of all switch combinations on power losses. In Fig. 9, we present a heatmap visualizing the power losses associated with each possible combination of switch states in our network model. Note that each axis represents a binary encoding of switch states. The x-axis corresponds to switches 1-4, and the y-axis to switches 5-8. Each index represents a binary number, with the least significant bit corresponding to the first switch in the group. For instance, an index of '3' on the x-axis (representing switches 1-4) would translate to '0011' in binary, indicating that switches 3 and 4 are in the 'Closed' state while switches 1 and 2 are in the 'Open' state. The y-axis should be read in the same manner for switches 5-8. The color intensity of each cell in the heatmap indicates the level of power losses

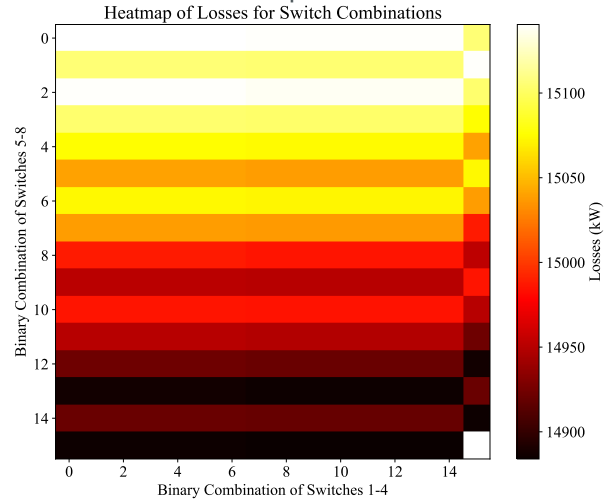


Fig. 9. Network losses due to switching of tie lines.

for the corresponding combination of switch states, facilitating comparative analysis of how different configurations affect the network. We observe that the power losses for each switch combination varied within a range of 10 to 100 kW. It's important to note that in larger-scale systems, these values can potentially be much higher.

D. Scenario 4: Load Shedding

This scenario is primarily used to validate the proposed approach's capacity to defend against a more stressful overloading scenario. These difficult conditions push the agent to learn about the trade-off relationship between load shedding and maintaining the maximum number of connected loads. It also demonstrates how the agent will optimally control load-shedding and determine which regions will be disconnected, as seen in Fig. 10. In this paper, we assume that each region has a similar power consumption level. Hence the emphasis will be on the number of disconnected loads. We may examine more scalable and realistic instances in future work to find the specific load names to be disconnected. During that produced attack, n and l are set to 200% to attack some loads in that region randomly. This method is intended to replicate a real-world attack that infects smart meters randomly inside a clustered distribution network zone. It can be observed that the proposed DRL-based approach is able to successfully regulate the voltage within the security limit and has the least power losses among all approaches as shown in Fig. 11. As for topology configuration, it selects to close 5 out of 8 tie switches and load-shedding is executed that time for a total of 57 percent of loads got disconnected. Also, the total number of voltage violations is reduced to a minimum of 8 violations instead of 149. For the scenarios involving severe cyber attacks that necessitate load shedding, our SAC approach strategically balances voltage regulation and load reduction. As depicted in Fig. 12, the SAC-generated profile demonstrates some voltage violations; however, this is a calculated trade-off to minimize the extent of load shedding. Note that utilizing tie switches during load shedding primarily focuses on minimizing power

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher.

losses within the network. However, the role of these switches in significantly reducing voltage violations is limited, as such violations are affected by a variety of elements beyond the scope of tie switch operations.

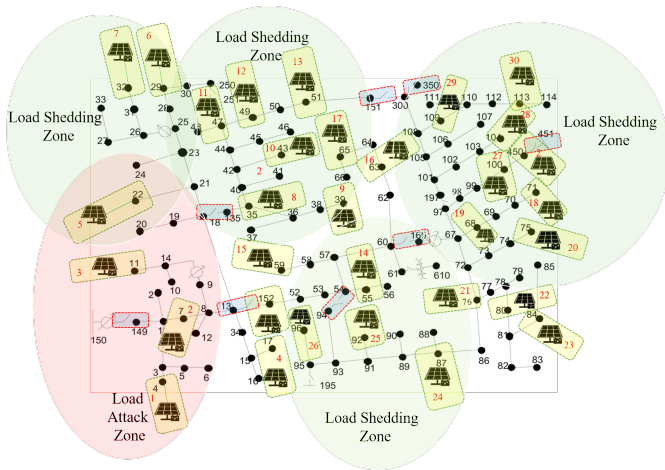


Fig. 10. Load attacks and shedding scenarios.

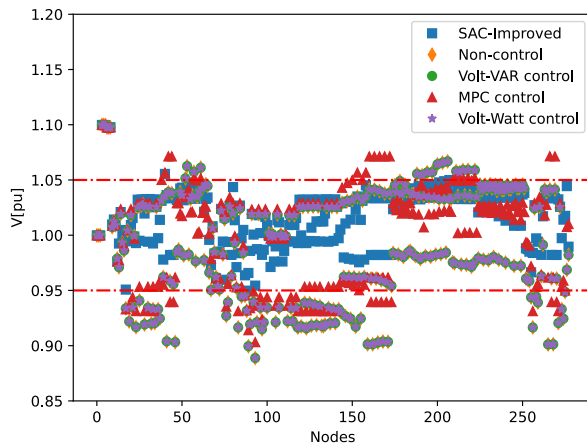


Fig. 11. Comparison results for different cyber defense algorithms-Scenario 4.

V. CONCLUSION

In this paper, the defense against cyber attacks on electrical grids is reformulated as an MDP problem and addressed using an enhanced DRL method. This strategy incorporates auto-tune entropy and a Gaussian policy specifically tailored for the dynamic management of DERs set points and load-shedding operations. This approach, unlike traditional MPC and other conventional methods, has demonstrated its efficacy in mitigating the impacts of voltage fluctuations and reducing power losses under a spectrum of cyber attack scenarios. Furthermore, it shows promise in adapting to multiple attack scenarios beyond the voltage regulation in systems with high

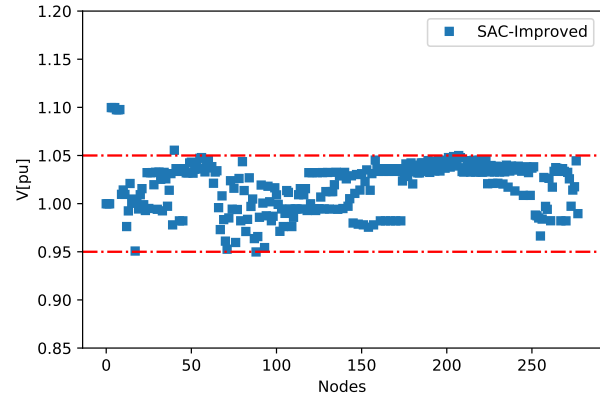


Fig. 12. The performance of the proposed DRL in mitigating voltage violations.

DERs penetration. Future work will extend this work to larger network scales, evaluating the adaptability and efficiency of the trained DRL agents in diverse network topologies and configurations. Additionally, we will explore adversary agents, where the agent must defeat this strong adversary during training time, thus becoming robust against a wide range of attacks.

REFERENCES

- [1] G. Liang, S. R. Weller, J. Zhao, F. Luo, and Z. Y. Dong, "The 2015 ukraine blackout: Implications for false data injection attacks," *IEEE Trans. Power Systems*, vol. 32, no. 4, pp. 3317–3318, 2016.
- [2] D. U. Case, "Analysis of the cyber attack on the ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, vol. 388, pp. 1–29, 2016.
- [3] "2019 attack on the power grid in california," <https://www.cnbc.com/2019/05/02/ddos-attack-caused-interruptions-in-power-system-operations-doe.html>. Accessed: 2022-12-30.
- [4] G. Wu, J. Sun, and J. Chen, "Optimal data injection attacks in cyber-physical systems," *IEEE Trans. cybern.*, vol. 48, no. 12, pp. 3302–3312, 2018.
- [5] S. Xu, Y. Xia, and H.-L. Shen, "Analysis of malware-induced cyber attacks in cyber-physical power systems," *IEEE Trans. Circuits and Systems II: Express Briefs*, vol. 67, no. 12, pp. 3482–3486, 2020.
- [6] I. Zografopoulos, J. Ospina, X. Liu, and C. Konstantinou, "Cyber-physical energy systems security: Threat modeling, risk assessment, resources, metrics, and case studies," *IEEE Access*, vol. 9, pp. 29775–29818, 2021.
- [7] T. Mahjabin, Y. Xiao, G. Sun, and W. Jiang, "A survey of distributed denial-of-service attack, prevention, and mitigation techniques," *International Journal of Distributed Sensor Networks*, vol. 13, no. 12, p. 1550147717741463, 2017.
- [8] D. M. Cappelli, A. P. Moore, and R. F. Trzeciak, *The CERT guide to insider threats: how to prevent, detect, and respond to information technology crimes (Theft, Sabotage, Fraud)*. Addison-Wesley, 2012.
- [9] D. An, Q. Yang, W. Liu, and Y. Zhang, "Defending against data integrity attacks in smart grid: A deep reinforcement learning-based approach," *IEEE Access*, vol. 7, pp. 110835–110845, 2019.
- [10] A. J. Abianeh, Y. Wan, F. Ferdowsi, N. Mijatovic, and T. Dragičević, "Vulnerability identification and remediation of fdi attacks in islanded dc microgrids using multiagent reinforcement learning," *IEEE Trans. Power Electronics*, vol. 37, no. 6, pp. 6359–6370, 2021.
- [11] F. Wei, Z. Wan, and H. He, "Cyber-attack recovery strategy for smart grid based on deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2476–2486, 2019.

Pursuant to the DOE Public Access Plan, this document represents the authors' peer-reviewed, accepted manuscript.

The published version of the article is available from the relevant publisher.

- [12] C. Roberts, S.-T. Ngo, A. Milesi, S. Peisert, D. Arnold, S. Saha, A. Scaglione, N. Johnson, A. Kocheturov, and D. Fradkin, "Deep reinforcement learning for DER cyber-attack mitigation," in *2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pp. 1–7, IEEE, 2020.
- [13] C. Roberts, S.-T. Ngo, A. Milesi, A. Scaglione, S. Peisert, and D. Arnold, "Deep Reinforcement Learning for Mitigating Cyber-Physical DER Voltage Unbalance Attacks," in *2021 American Control Conference (ACC)*, pp. 2861–2867, IEEE, 2021.
- [14] S. Pan, T. Morris, and U. Adhikari, "Classification of disturbances and cyber-attacks in power systems using heterogeneous time-synchronized data," *IEEE Trans. Ind. Informat.*, vol. 11, no. 3, pp. 650–662, 2015.
- [15] T. Zhou, K. Xiahou, L. Zhang, and Q. Wu, "Multi-agent-based hierarchical detection and mitigation of cyber attacks in power systems," *International Journal of Electrical Power & Energy Systems*, vol. 125, p. 106516, 2021.
- [16] W. Wang and Z. Lu, "Cyber security in the smart grid: Survey and challenges," *Computer networks*, vol. 57, no. 5, pp. 1344–1371, 2013.
- [17] W. Wang, F. Harrou, B. Bouyeddou, S.-M. Senouci, and Y. Sun, "A stacked deep learning approach to cyber-attacks detection in industrial systems: application to power system and gas pipeline systems," *Cluster Computing*, vol. 25, no. 1, pp. 561–578, 2022.
- [18] Y. Xu, "A review of cyber security risks of power systems: from static to dynamic false data attacks," *Protection and Control of Modern Power Systems*, vol. 5, no. 1, pp. 1–12, 2020.
- [19] Z. El Mrabet, N. Kaabouch, H. El Ghazi, and H. El Ghazi, "Cybersecurity in smart grid: Survey and challenges," *Computers & Electrical Engineering*, vol. 67, pp. 469–482, 2018.
- [20] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Trans. Smart Grid*, vol. 8, no. 4, pp. 1630–1638, 2016.
- [21] K. Lai, M. Illindala, and K. Subramaniam, "A tri-level optimization model to mitigate coordinated attacks on electric power systems in a cyber-physical environment," *Applied energy*, vol. 235, pp. 204–218, 2019.
- [22] A. Selim, J. Zhao, F. Ding, F. Miao, and S.-Y. Park, "Deep Reinforcement Learning for Distribution System Cyber Attack Defense with DERs," in *2023 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, IEEE, 2023.
- [23] J. Qi, A. Hahn, X. Lu, J. Wang, and C.-C. Liu, "Cybersecurity for distributed energy resources and smart inverters," *IET Cyber-Physical Systems: Theory & Applications*, vol. 1, no. 1, pp. 28–39, 2016.
- [24] U.S. Department of Energy, "Cybersecurity Considerations for Distributed Energy Resources on the U.S. Electric Grid," October 2022. Accessed: 2023-11-07.
- [25] A. Gopstein, N. Hastings, L. Feldman, R. Agarwal, and N. Bartol, "Distributed energy resource security: potential guidelines and research topics," Tech. Rep. NIST TN 2182, National Institute of Standards and Technology (U.S.), Gaithersburg, MD, 2021.
- [26] S. D. Roy and S. Debbarma, "Detection and mitigation of cyber-attacks on age systems of low inertia power grid," *IEEE Systems Journal*, vol. 14, no. 2, pp. 2023–2031, 2019.
- [27] P. Srikantha and D. Kundur, "A DER attack-mitigation differential game for smart grid security analysis," *IEEE Trans. Smart Grid*, vol. 7, no. 3, pp. 1476–1485, 2015.
- [28] P. S. Sarker, M. F. Rafy, A. K. Srivastava, and R. Singh, "Cyber Anomaly-Aware Distributed Voltage Control with Active Power Curtailment and DERs," *IEEE Trans. Industry Applications*, 2023.
- [29] D. Jafarigiv, K. Sheshyekani, M. Kassouf, Y. Seyedi, H. Karimi, and J. Mahseredjian, "Countering FDI Attacks on DERs Coordinated Control System Using FMI-Compatible Cosimulation," *IEEE Trans. Smart Grid*, vol. 12, no. 2, pp. 1640–1650, 2021.
- [30] T. Sauter and M. Lobashov, "End-to-end communication architecture for smart grids," *IEEE Trans. Industrial Electronics*, vol. 58, no. 4, pp. 1218–1228, 2011.
- [31] Electric Power Research Institute (EPRI), "Open distribution system simulator (openss)." <https://www.epri.com/pages/sa/openss>, 2023. Accessed: 2023-11-07.
- [32] RTDS Technologies, "Rscad fx real-time simulation software package." <https://knowledge.rtds.com/hc/en-us/articles/360046352893-RSCAD-FX-Real-Time-Simulation-Software-Package>, 2023. Accessed: 2023-11-07.
- [33] S. C. Dhulipala, R. V. A. Monteiro, R. F. da Silva Teixeira, C. Ruben, A. S. Bretas, and G. C. Guimarães, "Distributed model-predictive control strategy for distribution network volt/var control: A smart-building-based approach," *IEEE Trans. Ind. Appl.*, vol. 55, no. 6, pp. 7041–7051, 2019.
- [34] D. Cao, W. Hu, X. Xu, Q. Wu, Q. Huang, Z. Chen, and F. Blaabjerg, "Deep reinforcement learning based approach for optimal power flow of distribution networks embedded with renewable energy and storage devices," *Journal of Modern Power Systems and Clean Energy*, vol. 9, no. 5, pp. 1101–1110, 2021.
- [35] Y. Shuai, J. Fang, X. Ai, Y. Tang, J. Wen, and H. He, "Stochastic optimization of economic dispatch for microgrid based on approximate dynamic programming," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2440–2452, 2018.
- [36] Y. Zhou, B. Zhang, C. Xu, T. Lan, R. Diao, D. Shi, Z. Wang, and W.-J. Lee, "A data-driven method for fast ac optimal power flow solutions via deep reinforcement learning," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 6, pp. 1128–1139, 2020.
- [37] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*, pp. 1861–1870, PMLR, 2018.
- [38] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, et al., "Soft actor-critic algorithms and applications," *arXiv preprint arXiv:1812.05905*, 2018.
- [39] Pranz, "PyTorch implementation of Soft Actor-Critic." <https://github.com/pranz24/pytorch-soft-actor-critic>, 2023. Accessed: 17/11/2023.
- [40] A. Selim, J. Zhao, X. Zhang, and F. Ding, "Deep reinforcement learning for distribution system restoration using distributed energy resources and tie-switches," in *2022 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1–5, IEEE, 2022.
- [41] S. Diamond and S. Boyd, "Cvxpy: A python-embedded modeling language for convex optimization," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2909–2913, 2016.
- [42] S. Monger, R. Vega, H. Krishnaswami, et al., "Simulation of smart functionalities of photovoltaic inverters by interfacing openss and matlab," in *2015 IEEE 16th Workshop on Control and Modeling for Power Electronics (COMPEL)*, pp. 1–6, IEEE, 2015.