energies

MDPI

Article

Deep Reinforcement Learning-Driven Mitigation of Adverse Effects of Cyber-Attacks on Electric Vehicle Charging Station

Manoj Basnet [1] and Mohd. Hasan Ali [2,*]

[1] FedEx Services, Collierville, TN 38017, USA; manoj.basnet@fedex.com
[2] Department of Electrical and Computer Engineering, The University of Memphis, Memphis, TN 38152, USA
* Correspondence: mhali@memphis.edu

Abstract: An electric vehicle charging station (EVCS) infrastructure is the backbone of transportation electrification; however, the EVCS has various vulnerabilities in software, hardware, supply chain, and incumbent legacy technologies such as network, communication, and control. These standalone or networked EVCSs open up large attack surfaces for local or state-funded adversaries. The state-of-the-art approaches are not agile and intelligent enough to defend against and mitigate advanced persistent threats (APT). We propose data-driven model-free digital clones based on multiple independent agents deep reinforcement learning (IADRL) that uses the Twin Delayed Deep Deterministic Policy Gradient (TD3) to efficiently learn the control policy to mitigate the cyberattacks on the controllers of EVCS. Also, the proposed digital clones trained with TD3 are compared against the benchmark Deep Deterministic Policy Gradient (DDPG) agent. The attack model considers the APT designed to malfunction the duty cycles of the EVCS controllers with Type-I low-frequency attacks and Type-II constant attacks. The proposed model restores the EVCS operation under threat incidence in any/all controllers by correcting the control signals generated by the legacy controllers. Our experiments verify the superior control policies and actions of TD3-based clones compared to the DDPG-based clones. Also, the TD3-based controller clones solve the problem of incremental bias, suboptimal policy, and hyperparameter sensitivity of the benchmark DDPG-based digital clones, enforcing the efficient mitigation of the impact of cyberattacks on EVCS controllers.

Keywords: cyberattack; deep reinforcement learning (DRL); electric vehicle charging station; mitigation

Citation: Basnet, M.; Ali, M.H. Deep Reinforcement Learning-Driven Mitigation of Adverse Effects of Cyber-Attacks on Electric Vehicle Charging Station. *Energies* **2023**, *16*, 7296. https://doi.org/10.3390/en16217296

Academic Editors: José Matas, Jorge El Mariachet and Sen Tan

Received: 1 October 2023
Revised: 20 October 2023
Accepted: 23 October 2023
Published: 27 October 2023

1. Introduction

According to the second-quarterly (Q2) data of 2021 from the Alternative Fuels Data Center, the United States hosts 128,474 public and private electric vehicle charging station (EVCS) ports in 50,054 different station locations [1]. In 2021 alone, charging stations increased by more than 55% in the United States. This upsurge is anticipated to grow further along with the announcement of the Bipartisan Infrastructure law to build out the nationwide electric vehicle network in April 2021 [2]. In February of 2022, USDOT and USDOE announced $5 billion over five years for the new National Electric Vehicle Infrastructure (NEVI) program. In February of 2022, USDOT and USDOE announced $5 billion over five years for the new National Electric Vehicle Infrastructure (NEVI) program under the Bipartisan law to create a network of electric vehicle (EV) charging stations and designated alternative fuel corridors on the interstate highway [3].

In contrast with the broad interest and investment in transportation electrification and EVCS deployment, the cyber-physical security hygiene of EVCS standalone/network is often slow-paced, poorly defined, and understudied [4–7]. The internet-facing elements of EVCS are primarily designed for communications and controls with other internet of things (IoTs) and stakeholders such as EV, EV operators, grid, Supervisory Control and Data Acquisition (SCADA), EVCS owners, and push the air-gapped critical physical

infrastructures to the internet [8]. It could potentially open up large attack vectors for the interconnected systems of the EVCS. We present a brief review of the cyberattack detections and mitigations efforts in EVCS from the next paragraph that lays the foundation towards automated cyberattacks mitigation in EVCS.

The works of [9–13] assessed the impacts of cyber-enabled physical attacks on EVCS infrastructures ranging from disruption, damage, hijack, and so on. On that note, researchers have worked on numerous detection methods implementing different computational intelligence algorithms, including machine learning and deep learning [7,9,13–15]. Reference [7] designed and engineered a deep learning-powered [deep neural network (DNN), long short-term memory (LSTM)] network intrusion detection system that could detect DDoS attacks for the EVCS network based on network fingerprint with nearly 99% accuracy. Similarly, ref. [9] developed a stacked LSTM-based host intrusion detection system solely based on a local electrical fingerprint that could detect stealthy 5G-borne distributed denial of service (DDoS) and false data injection (FDI) attacks targeting the legacy controllers of EVCS with nearly 100% accuracy. Furthermore, several deep learning-based ransomware detection engines have been proposed, tested, and evaluated that can share the information in a cloud-based or distributed ransomware detection framework for EVCS [10]. Recently, generative AI models were deployed to train the detectors to tackle the cyberattack data scarcity problem. Our previous work successfully achieved more than 99% performance metrics in detecting the DDoS attacks on EVCS infrastructure [13].

De et al. designed a control-oriented model-based static detector (deviation in battery cell voltage) and a dynamic detector (using system dynamics) algorithm to detect denial of charging attacks and overcharging attacks on plug-in electric vehicle (PEV) battery packs [14]. The threshold-based static and filter-based dynamic detection techniques have the least flexibility toward advanced persistent threats (APT) and evolving zero-day attacks.

There have been attempts to address the mitigation of cyberattacks on smart grid paradigms [15–23]. However, no works have exactly addressed the mitigation, defense, and correction for the cyber-physical attacks on EV charging infrastructures. Girdhar et al. used the Spoofing, Tampering, Repudiation, Information Disclosure, Denial of service, Elevation of privilege (STRIDE) method for threat modeling and a weighted attack defense tree for vulnerability assessment in the ultra-fast charging (XFC) station [15]. The Hidden Markov Model (HMM)-based detection and prediction system for a multi-step attack scenario was proposed. The proposed defense strategy optimizes the objective function to minimize the defense cost added by the cost of reducing the vulnerability index. As a means of defense/mitigation, the authors recommended isolating and taking the compromised EVCS off the interconnections and intercommunication. The traditional isolation-based protection approach fails miserably in the smart grid due to the availability constraints of electricity and few reserved physical backups. On this note, Mousavian et al. implemented mixed-integer linear programming (MILP) that jointly optimizes security risk and equipment availability in grid-connected EVCS systems [16]. Still, their model aimed to isolate a subset of compromised and likely compromised EVCS, ensuring minimal attack propagation risk with a satisfactory level of equipment available for supply demand.

Acharya et al. derived the optimal cyber insurance premium for public EVCS to deal with the financial loss incurred by cyberattacks [17]. However, the cyber insurance does not address the mitigation of attack impacts. Reference [18] proposed the proportional-integral (PI) controller-based mitigation approach for the FDI attack on microgrids. This method is based on the reference tracking application. A feed-forward neural network produces the reference voltage required for the PI controller, and the PI controller injects the signal to nullify the FDI. The problem with the method is that the neural networks optimized under the microgrid's normal operating conditions may produce unreliable reference signals under adversarial conditions such as manipulated inputs. Recurrent neural networks better deal with reference tracking problems than regular feed-forward networks. The proposed model imposes additional hardware requirements and is not efficient enough to deal with non-linear and periodic FDI attacks. Reference [19] implemented the DRL-based approach

for mitigating oscillations of unstable smart inverters in distributed energy resources (DER) caused by cyberattacks. The adversary who gained system access can reconfigure the control settings of the smart grid to disrupt the distribution grid operations. To mitigate the impact, the authors trained the actor-critic-based proximal policy optimization (PPO) DRL to develop the optimal control policy to reconfigure the control settings of uncompromised DERs. However, this article has not presented the DRL efficacy of mitigation methods. Reference [20] proposed the concept of an Autonomous Response Controller that uses the hierarchical risk correlation tree to model the paths of an attacker and measures the financial risk at cyber physical systems (CPS) assets. Recently, deep learning powered mitigations engine are evolving with improved adaptation to system dynamics and control actions [24].

Based on the above discussions, it appears that state-of-the-art algorithms have progressed well for attack detection and prediction in EVCS, aided by cutting-edge computational intelligence at in-network and standalone levels. However, the current state-of-the-art lacks a proactive vision for developing embedded intelligence that could defend/correct the attacks on the EVCS controllers.

Above all, there is an imminent need to develop data driven distributed intelligence to proactively and independently defend the critical process controllers under the threat incidence. It motivates us to design, implement, and test local, independent agent DRL-based cyber defense clones (software agents) that could detect and mitigate controller-targeted APT in the EVCS charging process.

To fill the gap in cyber-physical defense research at EVCS, we propose novel, independent multiple-agent RL-based clones that oversee the critical functionality of all the controllers in the system and corrects and defend them under the detection of threat incidence as well as an anomaly. The proposed software agents operate solely based on the local data at EVCS to be purely air-gapped. Without shutting down the process, they can take over all the infected and frozen controllers under the worst cyberattack, such as ransomware and/or APT. In addition, the proposed TD3-based clones mitigation approach is designed, verified and compared against the benchmark Deep Deterministic Policy Gradient (DDPG) clones. For the verification of proposed algorithms, the PV-powered, off-the-grid standalone EVCS prototype with a battery energy storage (BES) and an EV with the corresponding control circuitry of MPPT controller, PI controller-based BES controller, and EV controller are designed. The APT attacks (Type-I: low-frequency attack, Type-II: constant magnitude attack) are engineered and launched on the duty cycle of the controllers. Since the scope of the paper is defense/mitigation, a threshold-based detection engine is used for simplicity. We summarize the contribution of this paper as follows:

- Proposed novel data-driven controller clones with TD3-based algorithm that could correct or take over the legacy controllers under APT detection.
- Agents can learn and adapt control policies online, accommodating changes in EVCS dynamics or configurations, a feature lacking in traditional legacy controllers.
- The proposed agents successfully restore the regular operation under the APT attacks and system anomaly on the legacy EVCS controllers.
- Finally, the proposed digital clones with TD3 outperforms the benchmark DDPG-based clones in terms of stability, convergence, and mitigation performance.

2. Cybersecurity Issues in EVCS

Our prior work [9] designed the PV-powered off-the-grid standalone EVCS prototype comprised of PV, BES, and EV with an associated control strategy. Figure 1 depicts the SCADA system communicating with three isolated field controllers: PV, BES, and EV. The EVCS architecture, control circuitry, system formulation, and component modeling are available in [9]. These field controllers are responsible for the reliable and safe operation of EVCS and hold exploitable technical vulnerabilities. Using social engineering and/or reverse engineering, the adversary can poison the control signals reaching the physical controllers at EVCS either at the network level of the SCADA or at the physical infrastructure

layer. On that note, the threat actors with domain expertise can launch vicious APT attacks on these legacy controllers. To deal with these APT, reinforcement learning can be the reasonable control paradigm. Figure 1 depicts the working mechanisms of an individual DRL-based controller agent and is valid for all controller agents: PV, BES, and EV. Each controller is equipped with the detection engine and the controller agent. The detection engine continuously monitors the control signals and detects the cyberattack, if any, on the deployed controller based on the heuristics of the signal integrity. Whenever the detection engine is triggered, the RL-based control agent takes over the control process bypassing the traditional controllers; maintains regular operation until the attack event is cleared. The detailed functionality and deployment of these agents with states, rewards, and actions information will be discussed in Section 4.

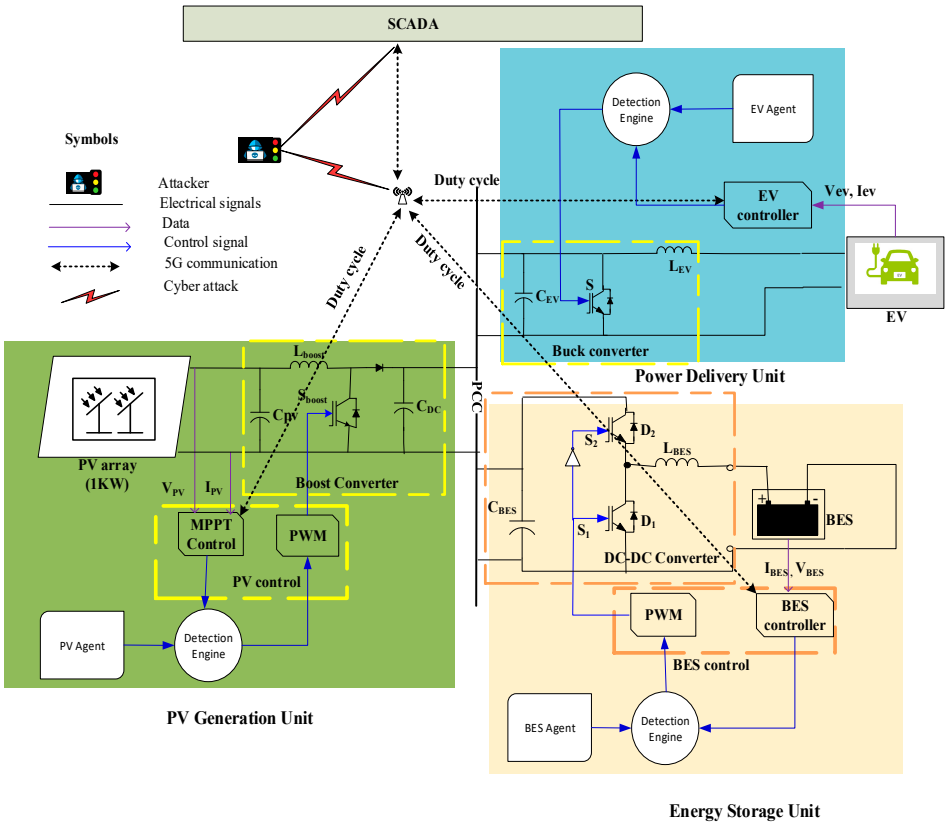

Figure 1. Proposed Detection and Defense based on DRL.

3. Attack Modeling and Detection

3.1. Attack Modelling

The attacker's primary goal is to disrupt, damage, or freeze the critical controllers of EVCS. The attacker is assumed to poison/manipulate the critical parameters with sophisticated attacking tools and domain expertise. Most legacy controllers generate a critical control signal, i.e., the duty cycle that controls the switching of the high-frequency transistor switches. It is assumed that the attacker decides the number of controllers $(N_c \in \mathbb{R})$ to exploit, the attack duration $(T_a \in \mathbb{R})$ and Types of the attack $S_a = \{(A_t, E_a)\}$ once it exploited the critical control signals $\mathcal{C} = \{C_1, C_2, .., C_n\}$ from controllers $N_1, N_2, .., N_n$. The attacker chooses the set of exploited resources ζ from another set $\mathcal{M} = \{N_c, T_a, S_a, \mathcal{C}\}$ in such a way as to minimize the critical functionality CF of the process as in (1). The attack Type S_a can be a tuple of attack time $A_t = \{sim, diff\}$ and engineered attack Types $E_a = \{\tau_1, \tau_2\}$ where *sim* and *diff* refer to the attack that can be launched simultaneously and at different times, respectively, with attack Types τ_1 and τ_2 defined in (2) and (3).

$$\underset{\zeta \subset \mathcal{M}}{\text{argmin}} CF \tag{1}$$

$$\tau_1 = \mathrm{T}(\alpha) \tag{2}$$

$$\tau_2 = c \tag{3}$$

where T is some random function parameterized by parameters α and c is some scalar constant. The function T is envisioned to generate the statistical randomness in the attack. With critical control signals of the controllers $\mathcal{C} \in [low\_limit, upper\_limit]$, it is wise for a stealthy attacker to design a similar kind of pseudorandom attack that intersects with the range of $\mathcal{C}$. Pseudorandom number *PRN* (*low_limit, up_limit, rep*) fluctuates between lower and upper bound, and repeating *rep* times serve the purpose. Similarly, c is the average of the upper and lower limit of the $\mathcal{C}$. After finding the sets of optimal ζ, Finally, the attacker algebraically combines the attack signal E_a with the critical parameter set $\mathcal{C}$ as per (4).

$$attack\_signal = \mathcal{C} \pm E_a \text{ subjected to } \zeta \tag{4}$$

We pragmatically chose the ζ and CF of the attack for this case as in Table 1 after the repeated experimentation. Both Type I and Type II attacks are carefully engineered APT attacks with domain expertise. The Type I attack imposes the low-frequency attack on the duty cycles, while the Type II attack imposes the constant duty cycle attack.

Table 1. Parameters for attack modeling.

Parameters	Value
N_c	3
T_a	2 s
S_a	$\{sim, diff\} \times \{\tau_1, \tau_2\}$
$\mathcal{C}$	$\{\mathfrak{D}_{PV}, \mathfrak{D}_{BES}, \mathfrak{D}_{EV}\}$
CF	Observed normalcy or stability of the process variables such as power, bus voltage, BES, and EV voltages and currents
τ_1	PRN (0, 1, 10)
τ_2	0.5
$Type\ I\ attack = \mathcal{C} \pm E_a$ s.t. $\zeta(.\tau_1)$	$\{\mathfrak{D}_{PV} + \tau_1, \mathfrak{D}_{BES} - \tau_1, \mathfrak{D}_{EV} - \tau_1\}$
$Type\ II\ attack = \mathcal{C} \pm E_a$ s.t. $\zeta(.\tau_2)$	$\{\mathfrak{D}_{PV} - \tau_2, \mathfrak{D}_{BES} - \tau_2, \mathfrak{D}_{EV} - \tau_2\}$

3.2. Detection Technique

The threshold-based detection engine is a good choice for simplicity and speed as it makes the online decision after implementing rule-based logic. This detection engine is founded on point anomaly detection that continuously oversees whether the control signal and controlled signal are within the predefined threshold range. If these signals fall outside the predefined thresholds, the engine decides it as an anomaly or attack. The detection logic at each controller can be defined as follows.

If $\left(Up\ perThreshold < \mathfrak{D}_{(.)} < Lower\ Threshold\right)$ && $\left(Up\ perThreshold < CF_{(.)} < Lower\ Threshold\right)$
  Continue with Legacy controllers, i.e., duty = $\mathfrak{D}_{(.)}$;
Else
  Correct the duty cycle with TD3 agent, i.e., duty = $\mathfrak{D}_{(corrected)}$;
End

The complete information of thresholds for respective controllers are presented in Sections 4.3–4.5, along with the agent design.

4. Proposed Mitigation Approach

The data-driven digital clones for the physical controllers employing a DRL-based TD3 algorithms are trained and deployed in each critical controller that controls a dynamic system's critical functionality. Moreover, the clones employing TD3 agents are compared with the benchmark DDPG algorithm. Upon the threat incidence or operational anomaly,

the rule-based or DL-based detection engine deploys the corrected control signal from the clones. It takes over the legacy controllers until the threat has been eliminated. The RL-based autonomous defense agent is employed for each controller whose primary purpose is to generate the corrected control signal upon incidences of cyberattacks and system anomalies. These controllers are designed for the mere to extreme adversarial setups such as APT or malware that could freeze/control the legacy controllers.

4.1. Twin Delayed Deep Deterministic Policy Gradient (TD3)

Actor-critic RL learns both value function (as in value-based RL) and policy (as in policy-based RL) and is proven to have better convergence properties, ability to learn stochastic policy, and efficacy in hyperdimensional or continuous action space. The function approximation error in actor-critic RL leads to overestimated value estimates and suboptimal policies [25]. TD3 is the off-policy actor-critic RL designed for continuous action space to minimize the impact of overestimation bias on both actor-critic networks by implementing three tasks. The first one is clipped double—Q Learning, where TD3 uses a minimum of two Q-values to form the target. The second one is the delayed policy updates of the target network. And the third one is the target policy smoothing, where TD3 adds noise to the target action so that the target policy could not exploit Q function error by smoothing out Q along the gradient of action. Algorithm 1 is the TD3 algorithm implemented for the controller agents.

Algorithm 1: TD3 Algorithm.

Each proposed standalone control agent for EVCS follows the strict training protocol as follows.
Initialize critic networks $Q = \left[Q_{\theta_1}, Q_{\theta_2}\right]$ and actor-network π_ϕ with random parameters θ_1, θ_2 and ϕ
Initialize target networks $\theta_1' \longleftarrow \theta_1, \theta_2' \longleftarrow \theta_2, \phi' \longleftarrow \phi$
Initialize replay buffer $\mathcal{B}$
for t = 1 to T do
Select action with exploration noise $a \sim \pi_\phi(s) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma)$
Store transition tuple $\langle s, a, r, s', d\rangle$ into $\mathcal{B}$ where d is the signal to indicate s' is the terminal state.
If s' is the terminal state, reset environment state.
Else randomly sample mini batch of N transitions $\langle s, a, r, s', d\rangle$ from $\mathcal{B}$
Compute the target actions and compute targets:

$$a'(s') = clip\Big(\mu_{\phi'}(s') + clip(\epsilon, -c, c), a_{low}, a_{high}\Big), \epsilon \sim \mathcal{N}(0,1)$$

$$y(r, s', d) = r + \gamma(1-d)\min_{i=1,2} Q_{\theta' i}(s', a'(s'))$$

Update critics Q-function by using one step of gradient descent:

$$\theta_i \longleftarrow \text{argmin}_{\theta_i} \nabla_{\theta_i} \frac{1}{|\mathcal{B}|} \sum_{\langle s,a,r,s',d\rangle \in \mathcal{B}} \left(Q_{\theta_i}(s,a) - y(r,s',d)\right)^2$$

If t mod *policy_delay*, **then**
Update ϕ by the deterministic policy gradient:

$$\nabla_\phi J(\phi) = \frac{1}{|\mathcal{B}|} \sum \nabla_a Q_{\theta i}(s,a)\,|_{a=\mu_\phi(s)} \nabla_\Phi \pi_\phi(s)$$

Update target networks:

$$\theta_i' \longleftarrow \tau\theta_i + (1-\tau)\theta_i'$$

$$\phi' \longleftarrow \tau\phi + (1-\tau)\phi'$$

End If
End for.

4.2. Graphical Representation of TD3 Algorithm

TD3 uses twin critic networks (critic 1 and critic 2) inspired from DRL with clipped Double Q-Learning, where it takes the smallest Q-value of two critic networks to remove the overestimation bias in $Q_{\theta i}(s, a)$. The concept of target networks was introduced to stabilize the agent training. The target network provides a stable objective and greater coverage of the training data as DNN requires multiple gradient updates to converge [25]. Without the fixed target, the accumulated residual errors after each update produce divergent values when paired with a policy maximizing the value estimate. Therefore, TD3 uses delayed

updates of actor-network (policy update) compared to critic network (value update), resulting in more stable training.

The replay buffer stores the history of agent experience and randomly fetches the data in mini batches to update actor and critic networks. There are six neural networks in TD3: two critic networks, two target networks for two critics, an actor-network, and a corresponding target network for the actor. Figure 2 summarizes the graphical abstract of a TD3 agent.

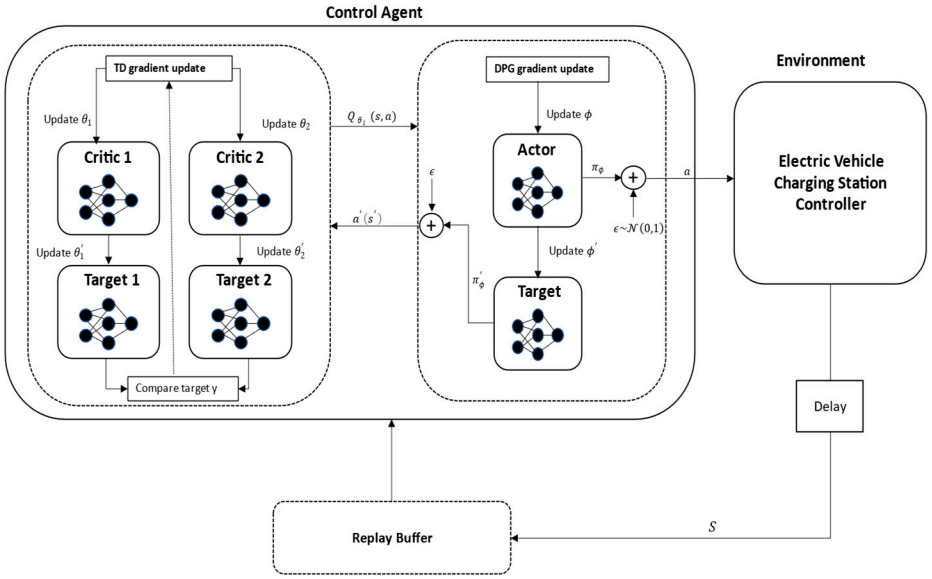

Figure 2. Graphical representation of a TD3 agent.

4.3. PV Agent

The design goal of the PV agent is to take over the infected MPPT controller and implement the optimized control policy to generate the duty cycle $\mathfrak{D}_{PV}(t)$ needed by the boost converter to have the least impact on the system. A PV agent continuously monitors the error $e_P(t)$and integrated errors $e_{int_P}(t)$between the PV output power $P_{PV}(t)$and reference power $P_{ref}(t)$ as in (5) and (6). The objective of the PV agent is to find the optimal policy for the duty cycle that correctly transforms observation space to action space by maximizing the cumulative scalar reward.

The output or action of the PV agent is the duty cycle with a linear quadratic regulator (LQR) as the instantaneous reward or cost function $r_{PV}(t)$ as in (7). The $\alpha = 0.01$ and $\beta = 1$ on $r(t)$ represent the negative penalty terms imposed on error and action, respectively. T_S is the sampling time and is same for each agent and is set to 0.1 s meaning the agent samples and updates the errors every 0.1 s throughout the operation.

$$e_P(t) = P_{ref}(t) - P_{PV}(t) \tag{5}$$

$$e_{int\_P}(t) = \sum\nolimits_{T_s} e_P(t) \tag{6}$$

$$r_{PV}(t) = \alpha e_P(t)^2 + \beta \mathfrak{D}_{PV}(t)^2 \tag{7}$$

The rule/threshold-based detection engine derived pragmatically for PV agent will determine the attack event if observed power falls beyond the range (1020, 1045) Watts and the duty of MPPT falls beyond the range (0.200, 0.201).

4.4. BES Agent

The design goal of the BES agent is to generate the corrected duty cycle $\mathfrak{D}_{BES}(t)$ for the buck-boost converter under the threat incidence. Similar to the PV agent, the BES agent

observes the states of the error $e_V(t)$and integrated errors $e_{int\_V}(t)$ between the desired reference bus voltage $V_{bus_ref}(t)$and the bus voltage $V_{bus}(t)$ as in (8) and (9). The optimal control policy that maps the observation space to action space is found by minimizing the expected value of the cost function $r_{BES}(t)$, which is the linear quadratic regulator function. The $\alpha = 0.01$ and $\beta = 1$ on $r_{BES}(t)$ represent the negative penalty terms imposed on error and action, respectively as in (10).

$$e_V(t) = V_{bus\_ref}(t) - V_{bus}(t) \tag{8}$$

$$e_{int\_V} = \sum_{T_s} e_V(t) \tag{9}$$

$$r_{BES}(t) = \alpha e_V(t)^2 + \beta \mathfrak{D}_{BES}(t)^2 \tag{10}$$

The rule/threshold-based detection engine derived pragmatically for the BES agent will determine the attack event if observed power falls beyond the range (1020, 1045) and the PI controller's duty falls beyond the range (0.7, 0.71).

4.5. EV Agent

The design goal of the EV agent is to generate the corrected duty cycle $\mathfrak{D}_{EV}(t)$ for a buck converter if the legacy EV charger got infected. Similar to the previous agent, the EV agent observes the states of the error $e_{VEV}(t)$and integrated errors $e_{int\_VEV}(t)$ between the desired reference battery voltage $V_{batt\_ref}(t)$ and the bus voltage $V_{batt}(t)$ as in (11) and (12). The optimal control policy that maps the observation space to action space is found by minimizing the expected value of the cost function $r_{EV}(t)$, which is the linear quadratic regulator function. The $\alpha = 0.01$ and $\beta = 1$ on $r_{EV}(t)$ represent the negative penalty terms imposed on error and action, respectively as in (13).

$$e_{VEV}(t) = V_{batt\_ref}(t) - V_{batt}(t) \tag{11}$$

$$e_{int\_VEV} = \sum_{T_s} e_{VEV}(t) \tag{12}$$

$$r_{EV}(t) = \alpha e_{VEV}(t)^2 + \beta \mathfrak{D}_{EV}(t)^2 \tag{13}$$

The rule/threshold-based detection engine derived pragmatically for the EV agent will determine the attack event if observed power falls beyond the range (1020, 1045) and the PI controller's duty falls beyond the operating range (0.54, 0.55).Table 2 summarizes the observations, reward, and action information of multiple TD3 agents.

Table 2. Summary of multiple independent agents.

Agents	Observations ($\mathcal{S}$)	Reward ($\mathcal{R}$)	Action ($\mathcal{A}$)
PV Agent	$\{e_P, e_{int\_P}\}$	$\{r_{PV}\}$	$\{\mathfrak{D}_{PV}\}$
BES Agent	$\{e_V, e_{int\_V}\}$	$\{r_{BES}\}$	$\{\mathfrak{D}_{BES}\}$
EV Agent	$\{e_{VEV}, e_{int\_VEV}\}$	$\{r_{EV}\}$	$\{\mathfrak{D}_{EV}\}$

5. Benchmark Deep Deterministic Policy Gradient (DDPG)

Deep Q Network (DQN) is a proven RL method capable of solving complex problems on par with human-level performance, as proven in Atari video games. However, DQN only solves the problem with high dimensional observation space and low dimensional discrete action space. For the continuous control problem that requires an iterative optimization process at every step to find the action that maximizes the action-value function, DQN cannot be applied straightforwardly. The DDPG is a model-free, online, off-policy actor-critic algorithm that can learn the optimal policies to maximize the expected cumulative

rewards in high dimensional continuous action spaces as in Algorithm 2. While training, a DDPG agent updates the critic and actor parameters at each time step. It stores past experiences in the circular experience buffer, and the agent updates the critic and actor parameters using mini batches of experiences selected randomly from the buffer. After that, in each time step, the action chosen by the policy with a stochastic noise model is perturbed.

Algorithm 2: DDPG Algorithm.

Initialize critic networks $Q(s, a|\theta^Q)$ and actor-network $\mu(s|\phi^\mu)$ with random parameters θ and ϕ
Initialize target networks Q'andμ' with $\theta' \longleftarrow \theta, \phi' \longleftarrow \phi$
Initialize replay buffer $\mathcal{B}$
for t = 1 to T, do
Select action with exploration noise $a \sim \pi_\phi(s) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma)$ and execute a in the EVCS environment.
Store transition tuple $\langle s, a, r, s', d\rangle$ into $\mathcal{B}$ where d is the signal to indicate s' is the terminal state.
If s' is the terminal state, reset environment state.
Else randomly sample mini batch of N transitions $\langle s, a, r, s', d\rangle$ from $\mathcal{B}$
Compute the target:
$y(r, s', d) = r + \gamma(1 - d)\mathrm{Q}'(s', \mu'(s'))$
Update critic Q-function by using one step of gradient descent:
$\theta \longleftarrow \mathrm{argmin}_\theta \nabla_{\theta_i} \frac{1}{|\mathcal{B}|} \sum_{\langle s,a,r,s',d\rangle \in \mathcal{B}} \left(Q_{\theta_i}(s, a) - y(r, s', d)\right)^2$
Update the actor policy ϕ by the deterministic policy gradient:
$\nabla_\phi J(\phi) = \frac{1}{|\mathcal{B}|} \sum \nabla_a Q_{\theta i}(s, a)\,|_{a=\mu_\phi(s)} \nabla_\Phi \pi_\phi(s)$
Update target networks:
$\theta\prime \longleftarrow \tau\theta + (1 - \tau)\theta\prime$
$\phi' \longleftarrow \tau\phi + (1 - \tau)\phi'$
End for

6. Experimental Setups for the TD3-Based Method

The TD3-based agents were built with specific neural architectures for critics and actor neural networks with similar architecture for the target neural network. Then, the layerwise actors' and critics' neural networks with their targets were properly engineered and parameterized with desired activation functions and appropriate initial weights and biases. Finally, the hyperparameters were carefully selected to train the agents optimally after the series of training up to 500 episodes. Similarly, all the hyperparameters and neural architectures were kept the same in the DDPG agents for accurate comparison.

6.1. Configurations of TD3 Critic Networks

A TD3 critic estimates the optimal Q-value based on the observations and actions received by the parameterized DNN. Figure 3 depicts the structure of a single critic network we have created. Before concatenating those features, the state and action information goes through some local neural network transformations. After concatenation, it goes through another set of neural networks to produce the Q-value function. The network that takes state information has three fully connected hidden layers with respective hidden units of 64, 32, 16, and the ReLU activation layer between them. After the concatenation, the transformed state and action info pass-through two fully connected hidden layers with 64 and 32 hidden units, respectively, with ReLU layer in between to produce the Q-value.

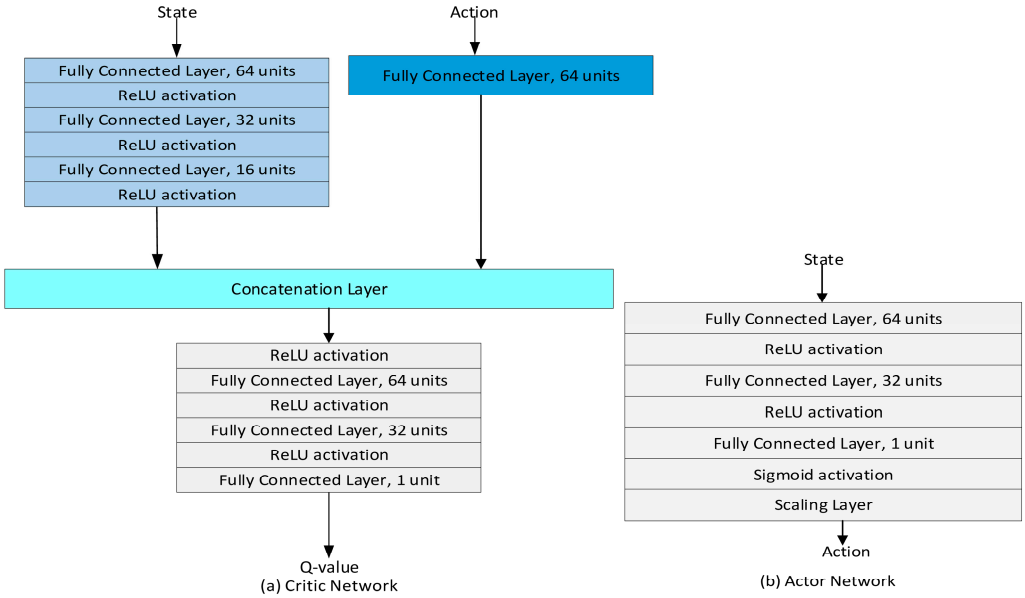

Figure 3. Structure of proposed (**a**) Critic-Network (**b**) Actor-Network.

6.2. Configurations of TD3 Actor Networks

The actor-network in the TD3 agent decided which action to take based on the observations, i.e., policy mapping. We have created a DNN with three fully connected hidden layers with respective hidden units of 64, 32, and units equal to the number of actions, i.e., 1 in our case with relu layers in between. In addition, a sigmoid layer was added since the output of the action ranges from 0 to 1 for duty cycle in our case. Finally, the scaling layer scaled the output from the sigmoid layer with a scale of 1 and a bias of 0.5. The scale was set to the range of action signal, and the bias was set to within half a range. We then created the actor representation using specified neural networks and options as in Figure 4. Table 3 presents the options of actor-network, critic network as well as training of agent. Table 4 presents the hyperparameters setting to administer the training.

Table 3. Actor-Critic Network parameters.

	Optimizer	**Learning Rate**	**Gradient Threshold**	**L2 Regularization Factor**
Critic	Adam	0.001	1	0.0001
Actor	Adam	0.001	1	0.00001

Table 4. Training parameters setting.

Discount factor	0.99
Experience buffer length	10^6
Mini-batch size	128
Number of steps to look ahead	10
Target smooth factor	5×10^{-3}
Target update frequency	2
Exploration variance	0.01
Target policy smooth variance	0.2

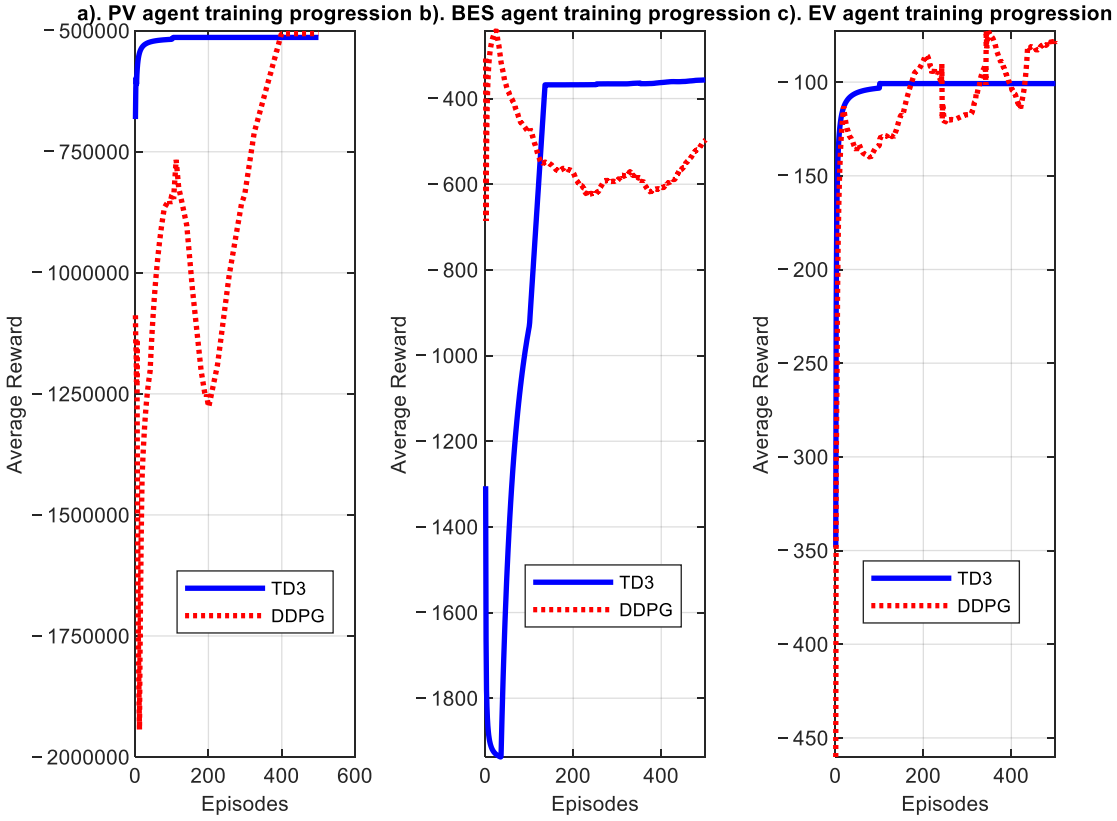

Figure 4. Training performance of the DDPG and TD3 agents in terms of average rewards.

6.3. Training an Agent

The agent trained by randomly selecting mini batches of size 128 with a discount factor of 0.99 towards the long-term reward from the replay buffer or experience buffer with a maximum capacity of 1×10^6. The target critics and actors were time-delayed clones of the critics and actor network with a smoothing factor of 0.005 that updated every two agent steps during training. The agent used a Gaussian action noise model with a specified noise variance and decay rate to explore the action space during training. The agent also smoothed the target policy updates using the specified Gaussian noise model. Each training consisted of 500 episodes, with each episode consisting of nearly 170 steps. The agent training was terminated when the agent receives an average cumulative reward of more than 800 over 100 consecutive episodes.

6.4. Computational Performance Comparison of DDPG and TD3

The proposed digital clones with DRL agents: the PV agent, BES agent, and EV agent, trained individually as the agents should learn to act independently employing both DDPG and TD3 algorithms. The motive behind designing the independent agents was that they should be able to work with legacy controllers (in case only a few controllers got infected) and other trained RL agents (all legacy controllers got infected). We train the agents as configured in Sections 4.3–4.5 independently for both DDPG and TD3 algorithms.

All the computations and simulations were performed in MATLAB 2022 b and Simulink 10.6 model version 5.6 in Dell XPS 15 7590 machine with i7-9750H CPU @2.6 GHz and 16 GB RAM. Each agent took approximately 6.68 h of training for the 500 episodes under DDPG and TD3. The TD3 training progress in terms of average rewards is shown in Figure 5 with stabilized reward within 99 episodes for PV agent, 136 episodes for BES agent, and 101 episodes for EV agent. However, the clones trained with DDPG exhibit poor convergence stability due to their higher sensitivity towards the hyperparameter settings though they were trained with the same hyperparameters and observation spaces as in

TD3. The optimal episodes for training DDPG agents are 398 for PV agent, 22 for BES agent, and 348 for EV agent after analyzing the rewards; Q-values as in Figures 5–7. The incremental bias and suboptimal policy seen in DDPG training was due to the overestimation of Q-values as they updated the Q-value as in DQN as evident in episode rewards of Figure 5 and the estimated Q-values in Figure 6. That is the reason the DDPG has myriads of suboptimal overshoots till the end of training progression. Therefore, the near optimal policy under DDPG training can be found in 398 epochs for PV agent, 22 episodes for BES, and 351 episodes for EV agent under the horizon of 500 episodes.

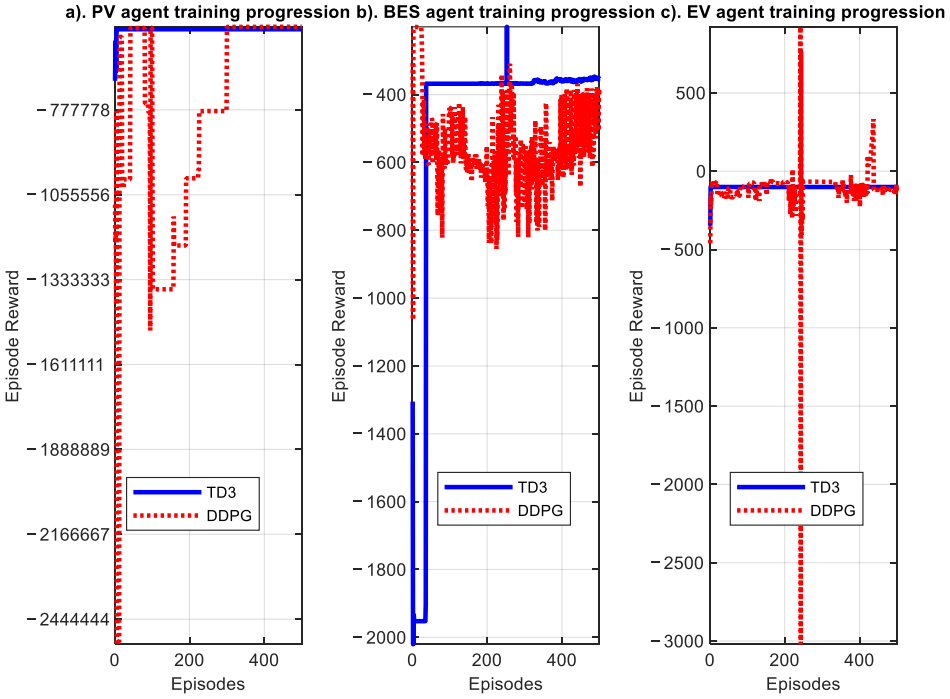

Figure 5. Training performance of the DDPG and TD3 agents in terms of episode rewards.

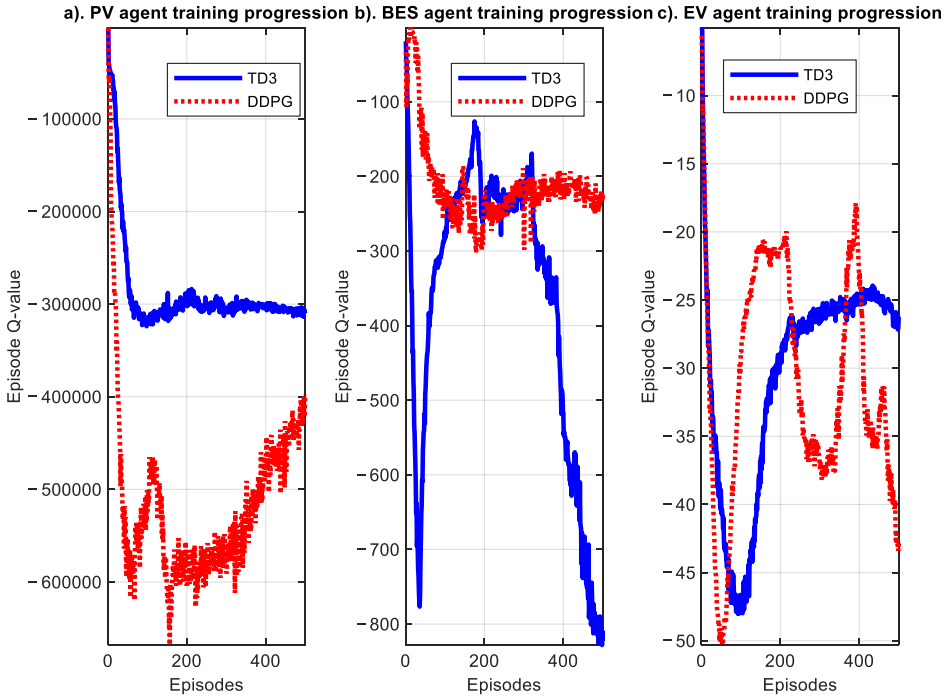

Figure 6. Training performance of the DDPG and TD3 agents in terms of episode Q-values.

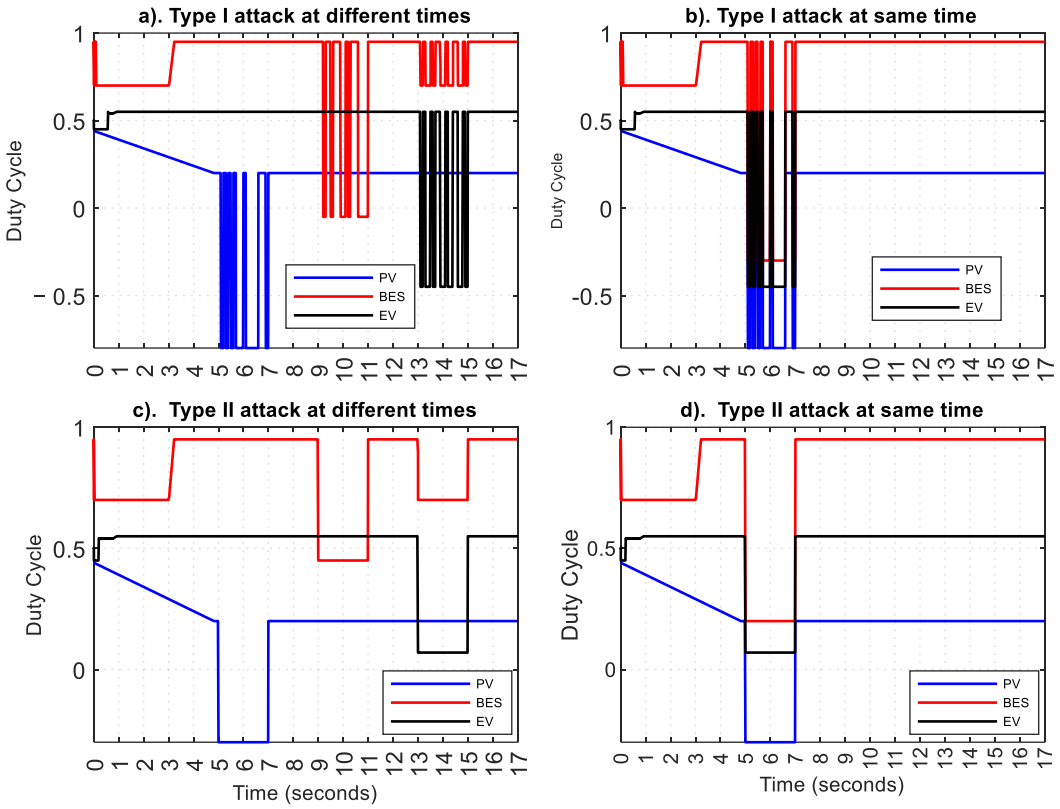

Figure 7. Impacts of the Type I and Type II attacks on duty cycles of different controllers.

7. Simulation Results and Discussion

7.1. Type I and Type II Attacks

Figure 7 summarizes the Type I and Type II attacks launched at different controllers at the same time and at different times. The Type I attack is the low-frequency attack at the duty cycles of the controllers, while Type II is the constant attack. The BES duty cycle was found to be more vulnerable to both kinds of attacks than the duty cycles of other controllers. The Type I attack has an irreversible impact on the BES controller as opposed to the Type II attack on the BES controller and both attacks on other controllers.

7.2. Type I Attack on Different Times and Mitigation Analysis

The Type I attack was launched in three different controllers PV controller at 5–7 s, BES controller from 9–11 s, and EV controller from 13–15 s, as shown in Figure 8. Tables 5–10 present the corresponding statistics of important electrical parameters. The Type I attack impacted all the critical electrical parameters. It forced the power to have approx. 2.99k times the normal range, 7.5k times the normal interquartile range (IQR), and median less than 18.4 Watt to the median at regular operation. The proposed mitigation restored the power with approximate errors of 0.002 watts in the median, 0.0001 watts in IQR, and −2.44 watts in range with the one at normal operations, as evident in Table 5.

Table 5. PV power statistics in Watt during normal, attack and mitigation.

	Range	IQR	Median
Normal	[1043.59, 1044.60]	[1043.593, 1043.599]	1043.5996
Attack	[1768.23, 1255.32]	[998.97, 1043.71]	1025.1726
Mitigation	[1040.15, 1043.60]	[1043.594, 1043.5998]	1043.5969

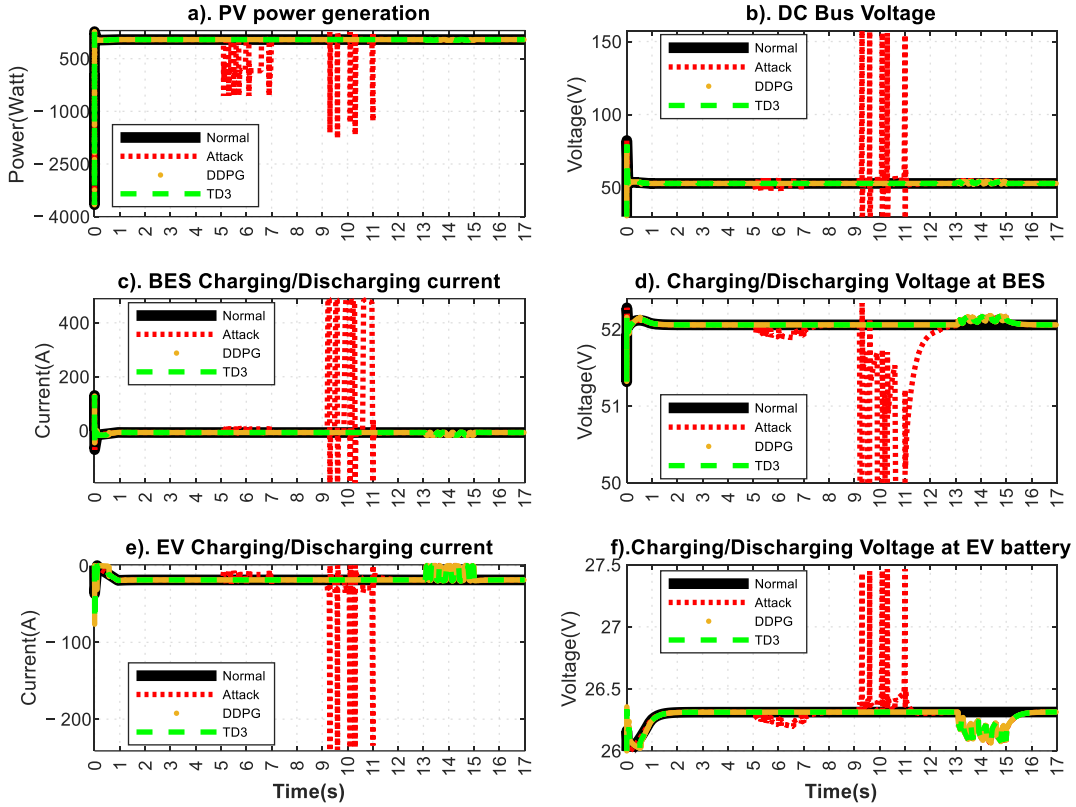

Figure 8. Impacts of Type I attack launched at PV controller from 5–7 s, BES controller from 9–11 s, and EV controller from 13–15 s and the mitigation performance during the attack.

Table 6. DC bus voltage statistics in Volts during normal, attack and mitigation.

	Range	IQR	Median
Normal	[52.7593, 52.761]	[52.7596, 52.7607]	52.7605
Attack	[−0.23051, 157.502]	[52.2102, 53.8464]	52.7553
Mitigation	[52.608, 53.1379]	[52.7596, 52.7608]	52.7605

Table 7. BES current statistics in Ampere during normal, attack and mitigation.

	Range	IQR	Median
Normal	[−6.947, −6.9435]	[−6.9435, −6.944]	−6.9452
Attack	[−193.294, 489.396]	[−8.816, −6.358]	−6.8273
Mitigation	[−9.143, −6.723]	[−6.9456, −6.944]	−6.945

Table 8. BES voltage statistics in Volts during normal, attack and mitigation.

	Range	IQR	Median
Normal	[52.0586, 52.0588]	[52.0586, 52.0588]	52.0587
Attack	[47.9982, 52.3418]	[51.5319, 52.1066]	51.9673
Mitigation	[52.0586, 52.0839]	[52.0586, 52.069]	52.0587

Table 9. EV current statistics in Ampere during normal, attack and mitigation.

	Range	IQR	Median
Normal	[−18.674, −18.668]	[−18.674, −18.669]	−18.6713
Attack	[−241.298, 5.26×10^{-5}]	[−19.531, −11.488]	−16.5862
Mitigation	[−18.9473, −14.887]	[−18.674, −18.671]	−18.6716

Table 10. EV voltage statistics in Volts during normal, attack and mitigation.

	Range	IQR	Median
Normal	[26.312, 26.313]	[26.312, 26.313]	26.3126
Attack	[26.056, 27.465]	[26.199, 26.344]	26.2524
Mitigation	[26.265, 26.313]	[26.293, 26.313]	26.3122

Similarly, the Type I attack has an inverse impact on bus voltage with a range elevation of approximately 158 V, IQR elevation of 1.63 V, and median reduction by 0.0052 V compared to the base operating conditions. The proposed mitigation can restore the bus voltage with approximate errors of 0 V in the median, 0.0001 V in IQR, and 0.5288 V in the range with the one at normal operations as per Table 6.

Also, as per Table 7, the Type I attack has an inverse impact on BES current with a range elevation of approximately 683 A, IQR elevation of 14 A, and median increment by 0.1 A compared to the base operating conditions. The proposed mitigation can restore the BES current with approximate errors of 0.0001 A in the median, 0.0013 A in IQR, and 2.4159 A in the range with the one at normal operations.

Likewise, the Type I attack has an inverse impact on BES voltage with a range elevation of approximately 4.3434 V, IQR elevation of 0.5747 V, and median decrement by 0.0914 V compared to the base operating conditions. The proposed mitigation can restore the BES current with approximate errors of 0.000 V in the median, 0.0102 V in IQR, and 0.0251 V in the range with the one at normal operations evident from Table 8.

Table 9 shows that the Type I attack has an inverse impact on EV current with a range elevation of approximately 241.2919 A, IQR elevation of 8.0385 A, and median increment by 2.0851 A compared to the base operating conditions. The proposed mitigation can restore the EV current with approximate errors of 3×10^{-4} V in the median, 0.0015 A in IQR, and 4.0534 A in the range with the one at normal operations.

Table 10 implies that the Type I attack has an inverse impact on EV voltage with the range elevation of approximately 1.4074 V, IQR elevation of 0.1438 V, and median decrement by 0.0602 V compared to the base operating conditions. The proposed mitigation can restore the BES current with approximate errors of $4.0000 \times 10^{-0.4}$ V in the median, 0.0193 V in IQR, and 0.0472 V in the range with the one at normal operations.

7.3. Type I Attack Simultaneously on All Controllers and Mitigation Analysis

The Type I attack was launched simultaneously in three different controllers at 5–7 s as shown in Figure 9. The Type II attack that launches at different times impacted all the critical electrical parameters.

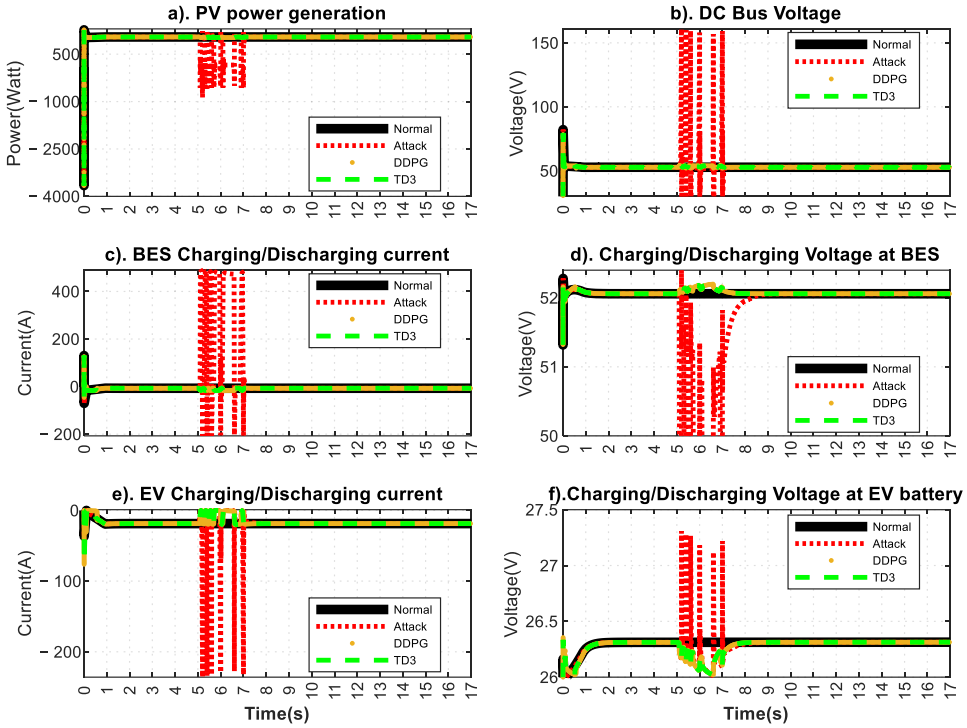

Figure 9. Impacts of Type I attack launched at PV controller from 5–7 s, BES controller from 5–7 s, and EV controller from 5–7 s and the mitigation performance during the attack.

7.4. Type II Attack on Different Times and Mitigation Analysis

The Type II attack was launched in three different controllers PV controller at 5–7 s, BES controller from 9–11 s, and EV controller from 13–15 s, as shown in Figure 10.

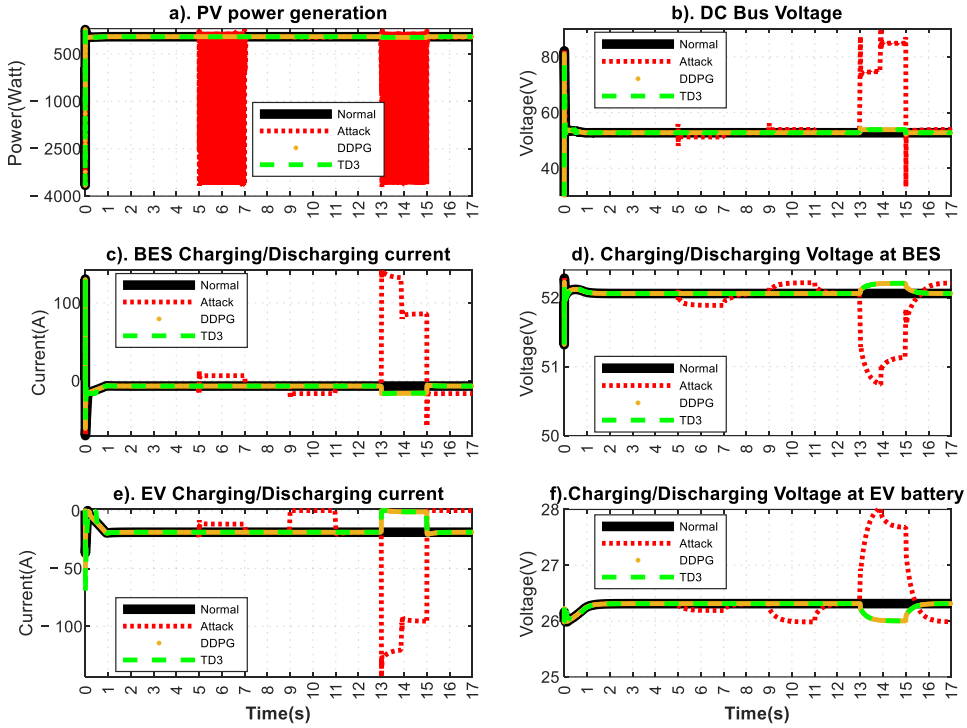

Figure 10. Impacts of Type II attack launched at PV controller from 5–7 s, BES controller from 9–11 s, and EV controller from 13–15 s and the mitigation performance during the attack.

7.5. Type II Attack Simultaneously on All Controllers and Mitigation Analysis

The Type II attack was launched simultaneously in three different controllers at 5–7 s as shown in Figure 11.

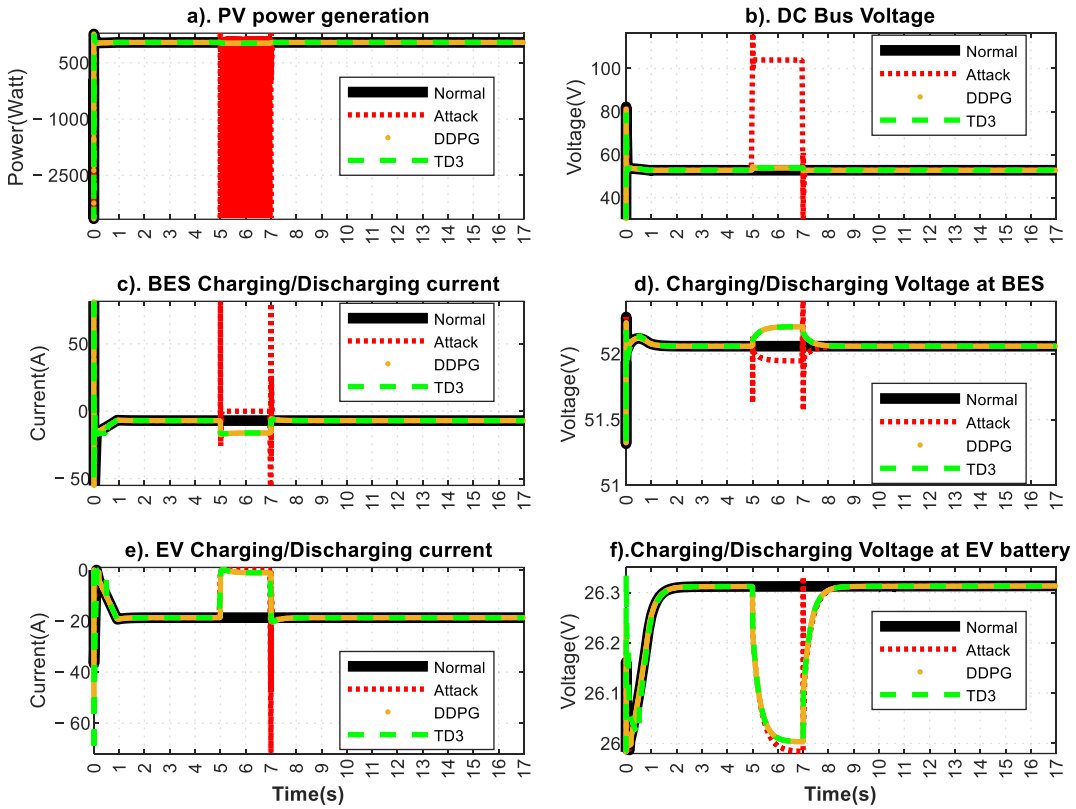

Figure 11. Impacts of Type II attack launched at PV controller from 5–7 s, BES controller from 5–7 s, and EV controller from 5–7 s and the mitigation performance during the attack.

7.6. Performance Comparison of Various Proposed Methods

Figure 12 depicts the control actions, i.e., duty cycles of the legacy controllers employed in the EVCS, trained clones with DDPG and TD3 algorithms. Under the normal operation, the legacy MPPT controller at PV stabilizes the duty cycle to 0.2 around 4.8 s. In contrast, the digital clone trained with DDPG and TD3 settles at a duty cycle of 0.4 and 0.495, respectively from the beginning as in Figure 12a. Figure 12b clearly shows the superior control action of TD3 duty cycle converged to 0.99 from the beginning as compared to DDPG BES clone converged to same after 4.2 s. The digital clone of EV controller trained with DDPG and TD3 has produce the same control action, i.e., duty cycle of 0.5. The legacy controllers are a manually tuned heuristic-based control that stabilizes the EVCS operation. However, it takes some time to stabilize the duty cycles. Contrastingly, the control actions taken by DDPG and TD3 are data driven as they optimize the control policy based on maximizing the expected long-term rewards. Therefore, clones are given freedom to choose the duty cycles that perfectly drive the normal operation of the EVCS.

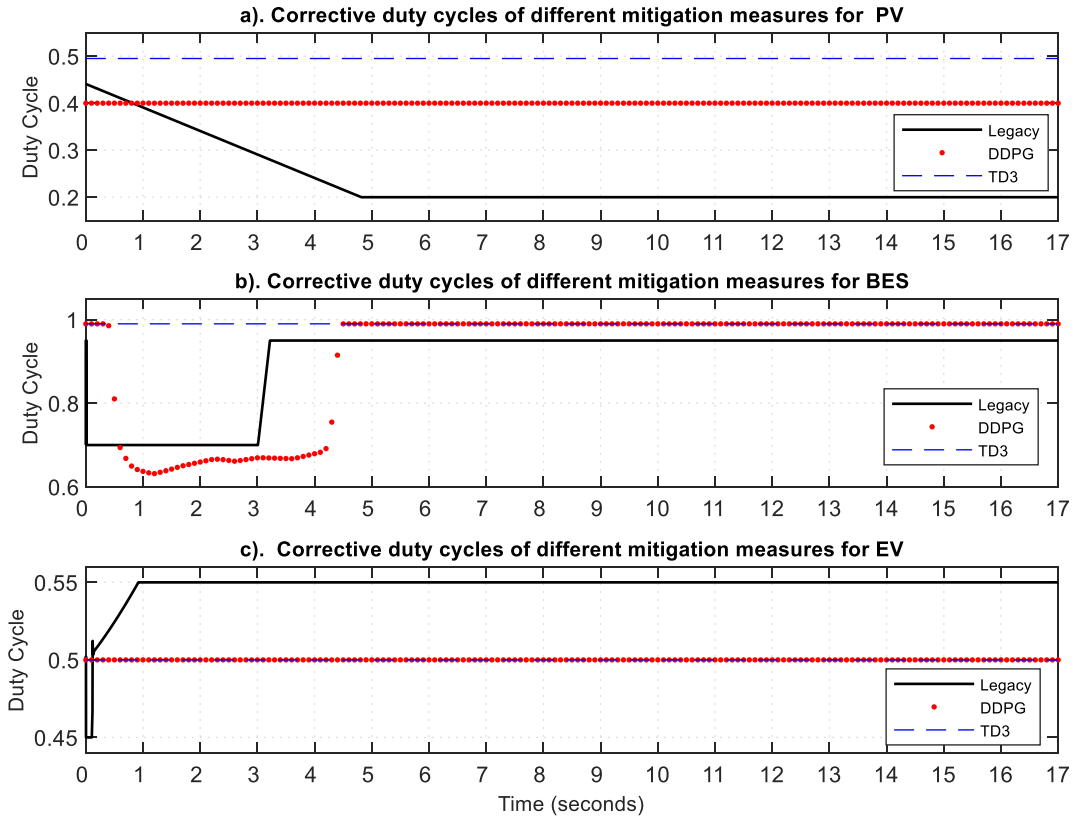

Figure 12. Control actions of Legacy, DDPG, and TD3 mitigations.

Table 11 presents the features of the proposed mitigation methods with respect to other related works. Our proposed air-gapped TD3-based mitigation has surpassed the various state-of-the-art methods in attack detection with online mitigation with embedded intelligence.

Table 11. Comparison between the proposed and the STATE-OF-THE-ART algorithms.

Solution	Attack Detection	Coordinated Attacks	Online Mitigation	Embedded Intelligence	Air Gapped
TD3 (our work)	√√	√√	√√	√√	√√
DDPG (Benchmark)	√√	√√	√√	X	√√
HIDS for EVCS [9]	√√	√√	X	√√	√√
NIDS for EVCS [7]	√√	√√	X	X	X
Weighted attack defense tree [15]	√√	√√	X	X	X

√√: Present; X: Absent.

8. Conclusions

This work devised a concept of data driven software clones implementing the state-of-the-art DRL (TD3) algorithm to mitigate the adverse effects of cyberattacks on the controllers of EVCS. The performance of the proposed TD3 based clones has been compared with that of the benchmark DDPG clones. The case studies uncover the following findings:

- The repetitive low-frequency attack (Type-I) on all controllers, at different times or simultaneously, has adverse impacts on critical functionalities of all controllers with the tendency to damage the EVCS with an upsurge/down surge in electrical signals. The agents successfully restore the EVCS operation by correcting the control signals of legacy controllers.

- The constant attack (Type-II) on controllers at different times or simultaneously tends to corrupt and damage the electrical components related to the legacy control actions. The proposed agents attempt to correct the control signals with the least error.
- The proposed TD3 based software clones are capable of taking over the legacy controllers under the APT attacks or even under the anomalous behavior.
- The TD3 based clones are found to be superior to DDPG based clones in terms of convergence, stability, hyperparameter sensitivity and mitigation actions.

Future research will focus on exploring other novel methods for detection and mitigation of cyberattacks on the EVCS controllers, and their performance will be compared with those in this work. Moreover, the new data driven control algorithm will be explored under the collaborative and adversarial setups to tackle the extreme cyber-physical attacks at EVCS. Also, 5G-based communication applications [9] in EVCS and related cybersecurity issues will be analyzed. Moreover, various game theoretic and dynamic programming approaches [26] along with powerful machine learning algorithms can be explored to develop optimal cyberattack-defense strategy in EVCS.

Author Contributions: Conceptualization, M.B. and M.H.A.; methodology, M.B.; software, M.B.; validation, M.B. and M.H.A.; formal analysis, M.B.; investigation, M.B.; resources, M.H.A.; data curation, M.B.; writing—original draft preparation, M.B.; writing—review and editing M.H.A.; visualization, M.B.; supervision, M.H.A.; project administration, M.H.A.; funding acquisition, M.H.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are pleased to acknowledge the financial support through Carnegie R1 Doctoral Fellowship from the Herff College of Engineering at the University of Memphis, USA, to complete this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Alternative Fuels Data Center: Electric Vehicle Charging Infrastructure Trends. Available online: https://afdc.energy.gov/fuels/electricity_infrastructure_trends.html (accessed on 22 February 2022).
2. The White House. FACT SHEET: Biden Administration Advances Electric Vehicle Charging Infrastructure. Available online: https://www.whitehouse.gov/briefing-room/statements-releases/2021/04/22/fact-sheet-biden-administration-advances-electric-vehicle-charging-infrastructure/ (accessed on 22 February 2022).
3. President Biden, USDOT and USDOE Announce $5 Billion over Five Years for National EV Charging Network, Made Possible by Bipartisan Infrastructure Law | FHWA. Available online: https://highways.dot.gov/newsroom/president-biden-usdot-and-usdoe-announce-5-billion-over-five-years-national-ev-charging (accessed on 22 February 2022).
4. Acharya, S.; Dvorkin, Y.; Pandžić, H.; Karri, R. Cybersecurity of Smart Electric Vehicle Charging: A Power Grid Perspective. *IEEE Access* **2020**, *8*, 214434–214453. [CrossRef]
5. Anderson, B.; Johnson, J. *Securing Vehicle Charging Infrastructure Against Cybersecurity Threats*; Sandia National Lab. (SNL-NM): Albuquerque, NM, USA, 2020. [CrossRef]
6. Park, Y.; Onar, O.C.; Ozpineci, B. Potential Cybersecurity Issues of Fast Charging Stations with Quantitative Severity Analysis. In Proceedings of the 2019 IEEE CyberPELS (CyberPELS), Knoxville, TN, USA, 29 April–1 May 2019; pp. 1–7. [CrossRef]
7. Basnet, M.; Ali, M.H. Deep Learning-based Intrusion Detection System for Electric Vehicle Charging Station. In Proceedings of the 2020 2nd International Conference on Smart Power Internet Energy Systems (SPIES), Bangkok, Thailand, 15–18 September 2020; pp. 408–413. [CrossRef]
8. Johnson, J. DER Cybersecurity Stakeholder Engagement, Standards Development, and EV Charger Penetration Testing. In Proceedings of the IEEE ECCE 2021, Vancouver, BC, Canada, 7 February 2021.
9. Basnet, M.; Ali, M.H. Exploring cybersecurity issues in 5G enabled electric vehicle charging station with deep learning. *IET Gener. Transm. Distrib.* **2021**, *15*, 3435–3449. [CrossRef]
10. Basnet, M.; Poudyal, S.; Ali, M.H.; Dasgupta, D. Ransomware Detection Using Deep Learning in the SCADA System of Electric Vehicle Charging Station. In Proceedings of the 2021 IEEE PES Innovative Smart Grid Technologies Conference—Latin America (ISGT Latin America), Lima, Peru, 15–17 September 2021; pp. 1–5. [CrossRef]

11. Gumrukcu, E.; Arsalan, A.; Muriithi, G.; Joglekar, C.; Aboulebdeh, A.; Zehir, M.A.; Papari, B.; Monti, A. Impact of Cyber-attacks on EV Charging Coordination: The Case of Single Point of Failure. In Proceedings of the 2022 4th Global Power, Energy and Communication Conference (GPECOM), Nevsehir, Turkey, 14–17 June 2022; pp. 506–511. [CrossRef]
12. Johnson, J.; Berg, T.; Anderson, B.; Wright, B. Review of Electric Vehicle Charger Cybersecurity Vulnerabilities, Potential Impacts, and Defenses. *Energies* **2022**, *15*, 3931. [CrossRef]
13. Basnet, M.; Ali, M.H. WCGAN-Based Cyber-Attacks Detection System in the EV Charging Infrastructure. In Proceedings of the 2022 4th International Conference on Smart Power & Internet Energy Systems (SPIES), Beijing, China, 9–12 December 2022; pp. 1761–1766. Available online: https://ieeexplore.ieee.org/abstract/document/10082342/ (accessed on 30 September 2023).
14. Dey, S.; Khanra, M. Cybersecurity of Plug-In Electric Vehicles: Cyberattack Detection During Charging. *IEEE Trans. Ind. Electron.* **2021**, *68*, 478–487. [CrossRef]
15. Girdhar, M.; Hong, J.; Lee, H.; Song, T. Hidden Markov Models based Anomaly Correlations for the Cyber-Physical Security of EV Charging Stations. *IEEE Trans. Smart Grid* **2021**, *13*, 3903–3914. [CrossRef]
16. Mousavian, S.; Erol-Kantarci, M.; Wu, L.; Ortmeyer, T. A Risk-Based Optimization Model for Electric Vehicle Infrastructure Response to Cyber Attacks. *IEEE Trans. Smart Grid* **2018**, *9*, 6160–6169. [CrossRef]
17. Acharya, S.; Mieth, R.; Konstantinou, C.; Karri, R.; Dvorkin, Y. Cyber Insurance Against Cyberattacks on Electric Vehicle Charging Stations. *IEEE Trans. Smart Grid* **2022**, *13*, 1529–1541. [CrossRef]
18. Habibi, M.R.; Baghaee, H.R.; Dragicevic, T.; Blaabjerg, F. False Data Injection Cyber-Attacks Mitigation in Parallel DC/DC Converters Based on Artificial Neural Networks. *IEEE Trans. Circuits Syst. II Express Briefs* **2021**, *68*, 717–721. [CrossRef]
19. Roberts, C.; Ngo, S.T.; Milesi, A.; Peisert, S.; Arnold, D.; Saha, S.; Scaglione, A.; Johnson, N.; Kocheturov, A.; Fradkin, D. Deep Reinforcement Learning for DER Cyber-Attack Mitigation. In Proceedings of the 2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Tempe, AZ, USA, 11–13 November 2020; pp. 1–7. [CrossRef]
20. Kholidy, H.A. Autonomous mitigation of cyber risks in the Cyber–Physical Systems. *Future Gener. Comput. Syst.* **2021**, *115*, 171–187. [CrossRef]
21. Singh, N.K.; Majeed, M.A.; Mahajan, V. Statistical machine learning defensive mechanism against cyber intrusion in smart grid cyber-physical network. *Comput. Secur.* **2022**, *123*, 102941. [CrossRef]
22. Huang, R.; Li, Y. Adversarial Attack Mitigation Strategy for Machine Learning-Based Network Attack Detection Model in Power System. *IEEE Trans. Smart Grid* **2023**, *14*, 2367–2376. [CrossRef]
23. Pradhan, N.R.; Singh, A.P.; Sudha, S.V.; Reddy, K.H.K.; Roy, D.S. Performance Evaluation and Cyberattack Mitigation in a Blockchain-Enabled Peer-to-Peer Energy Trading Framework. *Sensors* **2023**, *23*, 670. [CrossRef] [PubMed]
24. Basnet, M.; Ali, M.H. Deep-Learning-Powered Cyber-Attacks Mitigation Strategy in the EV Charging Infrastructure. In Proceedings of the 2023 IEEE Power & Energy Society General Meeting (PESGM), Orlando, FL, USA, 16–20 July 2023; pp. 1–5. [CrossRef]
25. Fujimoto, S.; van Hoof, H.; Meger, D. Addressing Function Approximation Error in Actor-Critic Methods. *arXiv* **2018**. Available online: http://arxiv.org/abs/1802.09477 (accessed on 3 March 2022).
26. Jay, D. Deception Technology Based Intrusion Protection and Detection Mechanism for Digital Substations: A Game Theoretical Approach. *IEEE Access* **2023**, *11*, 53301–53314. Available online: https://ieeexplore.ieee.org/abstract/document/10132467 (accessed on 14 October 2023). [CrossRef]