Electrical Power and Energy Systems 115 (2020) 105432

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

A game-theoretic approach for power systems defense against dynamic cyber-attacks

Saqib Hasan*, Abhishek Dubey, Gabor Karsai, Xenofon Koutsoukos

Institute for Software Integrated Systems, Vanderbilt University, Nashville, TN 37212, USA

ARTICLE INFO

Keywords:
Cascading failures
Cyber-attack
Dynamic attack
Game theory
Resilience
Smart grid
Static attack

ABSTRACT

Technological advancements in today's electrical grids give rise to new vulnerabilities and increase the potential attack surface for cyber-attacks that can severely affect the resilience of the grid. Cyber-attacks are increasing both in number as well as sophistication and these attacks can be strategically organized in chronological order (dynamic attacks), where they can be instantiated at different time instants. The chronological order of attacks enables us to uncover those attack combinations that can cause severe system damage but this concept remained unexplored due to the lack of dynamic attack models. Motivated by the idea, we consider a game-theoretic approach to design a new attacker-defender model for power systems. Here, the attacker can strategically identify the chronological order in which the critical substations and their protection assemblies can be attacked in order to maximize the overall system damage. However, the defender can intelligently identify the critical substations to protect such that the system damage can be minimized. We apply the developed algorithms to the IEEE-39 and 57 bus systems with finite attacker/defender budgets. Our results show the effectiveness of these models in improving the system resilience under dynamic attacks.

1. Introduction

Recent studies by the National Electric Research Council (NERC) documented that malicious attacks on power grids are much more devastating than the destruction caused by natural calamities [1] and can be instigated through cyber penetration [2] or physical obstruction [3] resulting in large blackouts. Today, power system resilience considering cyber-security has gained significant attention [4] as cyber-attacks are increasing both in number as well as sophistication and are considered as one of the major obstacles towards the reliable system operations [5–8]. For instance, due to the technological transformation of the traditional power grids into smart grids, power systems employ a large number of sophisticated and autonomous components such as protection devices, phasor measurement units (PMUs), remote terminal units (RTUs), etc. This increases the potential attack surface by giving rise to new vulnerabilities [9].

The attackers take advantage of such cyber components and gain access to the network by compromising the firewall and can launch catastrophic attacks, compromising system reliability [10], such as the recent Ukraine 2015 cyber-attack [11]. What makes the problem worse is the fact that most operators follow the guidelines from NERC [12] applying the $N-2$ reliability criterion [13], since analysis of higher order contingencies is computationally hard [14,15]. However, a cyber-attack is not limited to only two component failures.

Given such challenges, it is crucial to not only analyze a power system topology for reliability issues but it is also important to analyze the effect of cyber-attacks. In principle this can be approached by considering static attacks, where the devices are affected simultaneously or by dynamically sequenced attacks, which as shown in this paper, can cause significantly higher damage as compared to their static counterparts. Therefore, methods to study dynamic attack are important.

Several frameworks and attack models have been developed to study security vulnerabilities [16–26]. A man-in-the middle attack and modeling of cyber-physical switching attacks are presented in [16,17]. Several data integrity attack studies the effect of manipulating control messages, measurement data in [19,20]. A special type of false data injection attack, i.e., load redistribution (LR) attack is presented in [21,22]. The effect of cyber-attack on the voltage stability of support devices is provided in [23]. The work in [24] considers cyber-failures in protection assemblies and provides a platform to obtain new cascading traces. A real-time cyber-physical system testbed that provides mitigation strategies against attacks is discussed in [25]. An attack strategy using Self-Organising Maps (SOM) that is supposed to identify better

* Corresponding author.
E-mail addresses: saqib.hasan@vanderbilt.edu (S. Hasan), abhishek.dubey@vanderbilt.edu (A. Dubey), gabor.karsai@vanderbilt.edu (G. Karsai), xenofon.koutsoukos@vanderbilt.edu (X. Koutsoukos).

https://doi.org/10.1016/j.ijepes.2019.105432
Received 24 January 2019; Received in revised form 4 June 2019; Accepted 22 July 2019

attacks than load ranking and other clustering algorithms is developed in [26]. Additionally, a number of game-theory based studies have been done. For example, an efficient algorithm to solve the defender-attacker-defender problem for system protection is discussed in [27]. In [28], the authors formulate the problem as a minmax non-cooperative game and solved it using a genetic algorithm. Moreover, the work in [29] formulates the coordinated attacks on power systems as a bi-level optimization problem. The authors in [30] consider coordinated multi-switch attacks that leads to cascading failures in a smart grid. In [31], the authors studied a joint substation-transmission line vulnerability and proposed a component inter-dependency graph based attack strategy. Based on false data injection attacks, a Markov security game for attacks on automatic generation control is formulated in [32] and a time synchronization based attack is presented in [33]. Further, in [34] the effect of false data injection attacks against state estimation in power grids are studied. Finally, the work in [35,36] studies the temporal features of attacks in power systems.

However, there are several limitations in these approaches. The frameworks in [16,17,25] do not consider a system-wide identification of critical components to compromise. Attack models and strategies referenced in [18–23,27–34] focus on simultaneous attacks on different aspects of the system such as opening of circuit breakers, false data injection attacks in monitoring components, etc. In summary, none of these approaches consider cyber-attacks from the perspective of time domain, which is a vital facet in cascading failures since the progression of such failures takes at least minutes [37] or, at times, hours [38]. An attacker can easily and realistically sequence these attacks in a stealthy manner such that the attack mimics the trace of a normal cascading failure that could easily misguide the system operators. Moreover, considering strategically timed cyber-attacks reveal new system vulnerabilities which can not be found using previous approaches and their identification can enhance the overall power system resilience. Further, the attack model in [36] is based on the constructed sequential attack graph (SAG) which can be computationally infeasible for large power networks and most of them do not provide any defense model.

In this paper, we consider a game-theoretic approach to design attacker-defender cyber-attack and -defense models for power systems, to identify the worst-case dynamic attack. This work proposes a much simpler approach that does not require the construction of complex SAG as required by [36]. Further, we do not choose attacks based on node degree or load which enables us to explore a wider attack area. The specific contributions are:

- A formal dynamic attack model is described, where the cost to attack any substation and its components is uniform. In this model, the attacker can strategically identify the critical substations and their components to attack at different time instants in order to maximize the system damage, while constrained by the attacker's budget.
- A formal dynamic defense model is described, where the protection cost of any substation is uniform. In this model, given a defense budget, a defender can strategically identify the most critical substations to prioritize and protect so as to minimize the overall system damage.
- Two efficient polynomial-time algorithms are introduced to identify both the worst-case dynamic attack and a defense strategy which minimizes overall system damage.

Our results (shown using IEEE 39 and 57 bus examples) demonstrate that the approach captures the worst-case dynamic attacks on the power system networks and effectively uses the dynamic defense model to minimize the overall system damage. It also proves the effectiveness and efficiency of our algorithms. Moreover, the attack algorithm is able to maximize the system damage for both static and random attacks.

The remainder of this paper is organized as follows. The system model along with a motivating example is discussed in Section 2. Sections 3 and 4 give a detailed formal description of the static attack and defense models. The dynamic attack and defense models along with their algorithms are formally presented in Sections 5 and 6. Results are discussed in Section 7 followed by the conclusions in Section 8.

Table 1
List of commonly used symbols.

Symbol	Description
General Symbols	
$\mathcal{G}_{\mathcal{P}}$	power system model
S	set of substations
P	set of protection assemblies in a power system
S^i	i^{th} substation in S
$F(S^i)$	function that returns the set of protection assemblies in substation S^i
B_S	substation attack budget
B_P	protection assemblies attack budget
B_D	substation defense budget
Z	set of loads in $\mathcal{G}_{\mathcal{P}}$
Z_j	j^{th} load in Z
I_j	current flowing through j^{th} load in Z
Z_T	total power system load
Static Attack and Defense Model	
$S^{'}, S^{''}$	set of substations selected from S for static attack
$P^{'}, P^{''}$	set of protection assemblies selected from $P^{'}$ for static attack
$A_{P^{'}}$	static attack on substations $S^{'}$ and protection assemblies $P^{'}$
D_S	set of protected substations
Dynamic Attack and Defense Model	
k	time instant in $\{1, \ldots, T\}$
$S^{'}(k)$	set of substations selected from S for dynamic attack
$P^{'}(k)$	set of protection assemblies selected from P for dynamic attack
$A_{P^{'}}(k)$	dynamic attack on substations $S^{'}(k)$ and protection assemblies $P^{'}(k)$
$x(k)$	state of the system at k^{th} time instant
$H(k)$	attack history of the system $\mathcal{G}_{\mathcal{P}}$
$G(H(k))$	function representing the power system state under the presence of attack history $H(k)$ at time step k
$g(H(k))$	function representing nominal system state with no attack history
D_S	set of protected substations

2. System model and motivating example

In this section, we first present our abstract system model. Next, we use a motivating example to demonstrate that the dynamic attacks are much more catastrophic than the static attacks in a power system network. Finally, we present the assumptions that are made while designing our attack and defense models. For a list of symbols that are commonly used in this paper, see Table 1.

2.1. System model

We consider a power system $\mathcal{G}_{\mathcal{P}}$, where U is a set of buses, G is a set of generators, R is a set of transmission lines, Z is a set of loads, and P is a set of protection assemblies. The protection assemblies P consists of distance relay, over-current relay, and circuit breakers. The power system $\mathcal{G}_{\mathcal{P}}$ is divided into substations. Each substation has its own monitoring and control units referred to as RTUs. Let $S = \{S^i\}_{i=1}^{m}$ be the set of substations. Each substation consists of a set of protection assemblies from P. We define $F(S^i)$ as a function that returns the set of protection assemblies in a substation S^i. Clearly, the union of all the protection assemblies in every substation represents the set of P in the

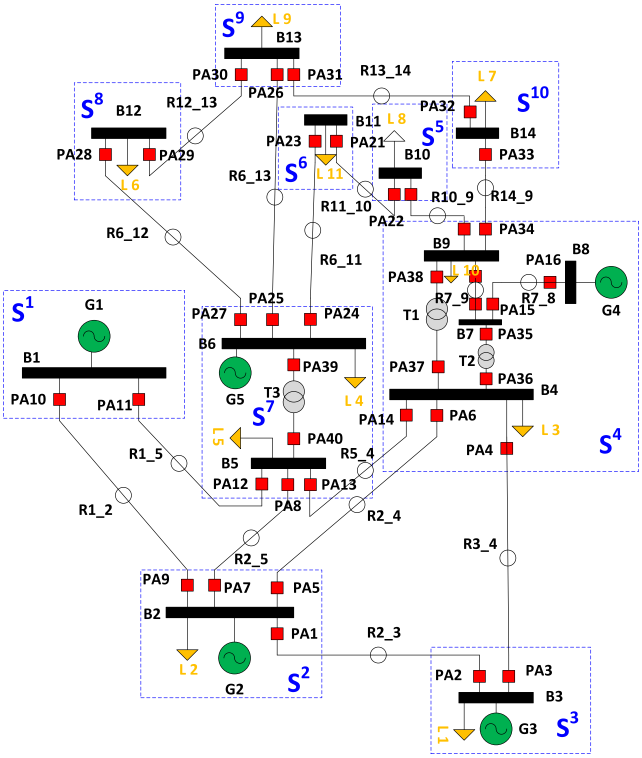


Fig. 1. IEEE-14 Bus System [39].

power network, that is, $\bigcup_{i=1}^{m} F(S^i) = P$.

2.2. Motivating example

To demonstrate the concept of static and dynamic attack, let us consider the IEEE 14 bus system [39] as shown in Fig. 1. The system is divided into substations as shown in Fig. 1 that are represented by blue dashed rectangles labeled as S^n, where $n \in \mathbb{N}$. The protection assemblies within the substations are labeled as *PAn*. Further, the transmission lines labeled as 'Rn_m' can be isolated by manipulating the protection assemblies *PAn* associated with substations S^n. The scenario described below has been simulated using the OpenDSS simulation tool and the line overload events were derived from observing the simulation results.

First, consider the static attack scenario where the protection assemblies associated with the transmission lines 'R6_13' and 'R7_8' (as shown in Fig. 1) are manipulated simultaneously to isolate them from the power network. This leads to removal of lines 'R9_14', 'R6_12', 'R9_10', 'R12_13' and loads 'L 5, L9, L4, and L7' as shown in Fig. 1 from the power network due to subsequent system overloading. Next, in case of dynamic attack, at first only transmission line 'R6_13' is isolated initially. As a result, cascading failure occurs that causes removal of lines 'R12_13', 'R9_14' and 'R6_12' due to transmission line overloading as shown in Fig. 1. The overloaded transmission lines are isolated. At this time another attack is executed, i.e., transmission line 'R7_8' is isolated. This results in further outages of lines 'R10_11', and 'R9_10' in the subsequent cascading stage. Post dynamic attack, the system lost a total of five loads namely; 'L 5, L8, L9, L4, and L 7' as opposed to 'L 5, L9, L4, and L7' in the static attack scenario. Considering the same components are attacked in both the scenarios with a difference in the attack execution time, dynamic attack caused a higher system damage as compared to its static counterpart (the state of the system is different when attacks are executed at different instants in time that causes dynamic attack to cause higher system damage) and provides the motivation to the problem. In this paper, we simulated the motivating example using Opendss on a standard IEEE-14 bus system and the idea of dynamic attacks causing higher system damage is well supported by various research works such as the one discussed in [36].

2.3. Attacker/defender model assumptions

In the following sections, we will describe the game-theoretic formulation of our attack and defense models in detail. Note, that the approach considers the attack-defense models in both static and dynamic scenarios as a two-player game where attacker and defender tries to maximize their own objectives, i.e., maximize/minimize power system damage. Further, the attacker and defender are both resources bounded. This is achieved by placing budget constraints on each of the participants. From the game-theory perspective, here budget constraints reflect the strategic choices that an attacker/defender can make whereas, maximizing individual objectives reflect the optimization problem that each player needs to solve. We do not consider the attacker and defender pay-offs from the game-theoretical approach since there are no gains that are obtained by an attacker/defender other than maximizing their own objectives which are already considered during problem formulation. Furthermore, we use a cyber-physical model in our approach where the physical system is simulated using Opendss and all the substations are assumed to be on an IP network that can be attacked and forms the cyber layer of the model. Once the attacker gain access to a substation by compromising the network, he can then manipulate its protection assemblies to cause system damage.

In addition, please also note that, our focus in this paper is not to design stealthy attack vectors but rather to provide a framework that includes theoretical analysis of the impact of various cyber-attack scenarios, i.e., static and dynamic attacks, without analyzing attack mechanisms (attack mechanisms have been analyzed in previous research works [16,17,19,20]). The paper also suggest that both static and dynamic attacks can be realized in practice (for instance, an attacker can gain access into a substation by compromising network firewall. He can then control various components of the power systems and manipulate them to cause system damage, e.g., disconnecting transmission lines) and dynamic attacks can cause significantly higher damage than their static counter parts. Further, according to [34] it is proved that bad data injection attacks are possible while being undetected. Moreover, these attacks can also be realized by injecting abnormal control signals or manipulate sensor data measurements that can result in unnecessary actuation of the switching devices. Our approach considers these possibilities abstractly, meaning if we slightly modify the relay settings of a protection assembly it will still result in opening of circuit breaker without being detected as a bad data injection attack. In this paper, our approach only focuses on the consequence of these attacks instead of focusing on how actually the attacks are realized in practice.

Before we dive into the details of dynamic attack and defense models, it is important to understand the problem from the static attack perspective. Therefore, we will first explain the static attack and defense models in detail to give the reader a better understanding about the problem in general. The evaluation of static attack and defense models are demonstrated in [40] and we build our dynamic attack and defense models on this foundation. Next, we discuss our proposed attack and defense models in detail.

3. Static attack model

In this section, we introduce the static attack model for the power system network, and then we provide an efficient algorithm to identify the worst-case static attack. Please note that throughout the paper, a worst-case attack is defined as an attack that maximizes the power system damage. We have further described the power system damage in detail in Section 3.1. For a list of symbols and methods used in the algorithms described in Sections 3–6, see Tables 2 and 3.

Table 2
List of Additional Symbols used in Attack and Defense Algorithms.

Symbol	Description
L_w	worst-case static attack load loss or system damage
L_{prev}	previous worst-case load loss or system damage
P_t	set of protection assemblies in a power system
$\hat{P}$	set of protection assemblies causing maximum system damage in the i^{th} iteration
$\hat{P}_t$	new set of protection assemblies that needs to be manipulated in i^{th} iteration to identify maximum system damage
L_H	list of system damage history
L_m	minimum system damage while protecting a substation
L_w^d	worst-case dynamic attack load loss
a_k^d	worst-case dynamic attack vectors
$a_k^{'}, a^d$	initialization list of attack vector
P^d	temporary set of protection assemblies required to generate contingencies
C	two dimensional list of contingencies
$L_s, L_s^{'}$	worst-case static and dynamic attack in i^{th} iteration respectively
P^*	set of protection assemblies causing i^{th} worst-case dynamic attack
a^*	dynamic attack vector causing i^{th} worst-case dynamic attack
a_{temp}^d	list of temporary attack vector representing time instants of previously attacked protection assemblies
a_k^*	list of attack vector for a set of contingency C in iteration k
P_{a_k}	temporary set of protection assemblies representing contingencies in iteration k
P_C	temporary list of protection assemblies representing contingencies C
$a_C, a_C^{'}$	temporary list of attack vectors
L_C	system damage corresponding to contingency C
P_t, P_i	temporary list of protection assemblies representing contingencies C

Table 3
List of methods.

Method Name	Use
Gen_Contin$(S, \hat{P}_a)$	Returns the set of contingencies based on the protection assemblies in S, and $\hat{P}_a$
Simulate_Model$(\mathcal{G}_\mathcal{P})$	Simulates the nominal state of the power system model $\mathcal{G}_\mathcal{P}$
Isolate_Branches$(\mathcal{G}_\mathcal{P}, p)$	Removes branch(es) from the power system model $\mathcal{G}_\mathcal{P}$ associated with the attacked protection assemblies p
Simulate_Contin$(\mathcal{G}_\mathcal{P}, p, k)$	Simulates the power system model $\mathcal{G}_\mathcal{P}$ with branch(es) removal at specific time instants k
Get_Branches$(\mathcal{G}_\mathcal{P}, p)$	Returns the overloaded branches in the power system model $\mathcal{G}_\mathcal{P}$ post attack
Get_Loads$(\mathcal{G}_\mathcal{P}, p, k)$	Returns the load names l that are disconnected in the power system model $\mathcal{G}_\mathcal{P}$ post attack
Get_Damage$(\mathcal{G}_\mathcal{P}, l)$	Returns the overall damage in the power system model $\mathcal{G}_\mathcal{P}$ post attack
Obtain_Subs(S, p)	Returns the substation(s) corresponding to the attacked protection assemblies p in the power system model $\mathcal{G}_\mathcal{P}$

3.1. Worst-case static attack

The objective of a malevolent attacker is to maximize the load loss and destabilize the power network. To achieve this, first the attacker may gain access to a subset of substations $S' \subseteq S$ where the attacker is resource bounded, i.e., the attacker can compromise at most B_S substations. Now, the adversary can identify the protection assemblies $P' \subseteq F(S')$ to manipulate them in order to isolate the transmission lines from the power network where the protection assemblies belong to the selected substations S'. The attacker is again resource bounded and can attack at most B_P protection assemblies. Note that budget on protection assemblies can be favourable for an attacker in the following ways:

- A naive attacker may select a large B_P and probably attack all the protection assemblies within the compromised substations, whereas, a strategic attacker may favor a small B_P as it would enable the attacker to remain undetected for a considerably longer period of time that could provide the attacker with an opportunity to potentially cause more system damage.
- Transmission lines are rated to carry a maximum amount of power and are isolated from the rest of the system in case of limit violations. This action often results in cascading failures causing severe load loss. Manipulating all the protection assemblies of a substation to disconnect power lines may reduce the overall system load. Hence, this may not lead to severe cascading failures causing higher load loss.

Please note that, in our attack model, we do not consider attack on generators since they are better protected as compared to other components such as transmission lines in the power systems. In addition, our approach is motivated by several previous realistic cyber-attacks such as the Ukraine 2015 attack [11], where the attackers attacked only the substations and transmission lines by opening circuit breakers to cause power system load loss. Another reason to consider only substations and transmission lines via protection assemblies as attack targets is that the attacker can remain hidden for a longer time by executing these attacks intelligently and as a result portray them as normal cascading failures without being detected easily. Similar assumption holds true for the dynamic attack model.

Next, the attack on a set of substations S' and protection assemblies P' is denoted by $A_{P'}$. Let Z_j denote the j^{th} load in the power network $\mathcal{G}_\mathcal{P}$. The current flowing through each load Z_j is given by I_j, where $j = 1$ to n. Power system damage, i.e., load loss is defined as the ratio of the sum of all the loads that are disconnected as a consequence of attackers actions from the entire power system model and the total power system load under nominal condition. Please note that similar definition holds for the dynamic attack case with the addition of time parameter. Now, we compute the damage function as below:

$$J(A_{P'}) = \frac{\sum_{j=1}^{n} Z_j}{Z_T} \times 100,\ \forall\ I_j = 0 \tag{1}$$

where Z_T represents the total system load. Hence, the attacker will try to maximize this damage function which is formally defined as follows.

Problem 1 (*Worst-Case Static Attack*). Given a power system network $\mathcal{G}_\mathcal{P}$, a substation budget B_S, and a protection assembly budget B_P, find a worst-case static attack $A_{P'}$ that maximizes the damage in the power system network. Formally,

$$\underset{S'}{\operatorname{argmax}} \max_{P' \subseteq F(S')} J(A_{P'}) \tag{2}$$

$$\begin{gathered} |S'| \leqslant B_S \\ \forall\ S', S'' \in S\colon S' \cap S'' = \varnothing \end{gathered} \tag{3}$$

$$\begin{gathered} |P'| \leqslant B_P \\ \forall\ P', P'' \in P\colon P' \cap P'' = \varnothing \end{gathered} \tag{4}$$

$$B_S \leqslant B_P \tag{5}$$

where S', S'' represents the substations that are selected to be attacked, however no substation is attacked twice. Note that, similar assumption is true for protection assemblies.

3.2. Algorithm for finding worst-case static attack

Now, we describe Algorithm 1, i.e., Get_WSA($\mathcal{G}_{\mathcal{P}}$, B_P, S) that illustrates the computation of worst-case static attack in detail. It is based on iteratively identifying attacks on substations and protection assemblies that maximize the system damage depending upon the budget constraints, i.e., B_S and B_P. The budget constraints also help in identifying combination of substations and protection assemblies that cause maximum system damage but can not cause such damage when attacked individually. Our algorithm takes this into consideration and utilizes the budget constraints to identify such combinations effectively.

The algorithm takes as inputs the power system model $\mathcal{G}_{\mathcal{P}}$, protection assembly's budget B_P, and power system substation information S. Further, it identifies the worst-case static attack by identifying a set of critical substations to compromise S', the protection assemblies to manipulate P' and the damage caused by the attack L_w. Please note that substation(s) in our model that causes the highest system damage when attacked either individually or in conjunction with other substations depending upon the attackers budget is/are identified as critical substation(s) using our proposed algorithms. Also, please note that similar analogy is true for dynamic attack scenario and the algorithm to find critical substation(s) for dynamic scenario is defined in the later section.

Algorithm 1. Algorithm for Finding Worst-Case Static Attack: Get_WSA($\mathcal{G}_{\mathcal{P}}$, B_P, S)

```
Input: $\mathcal{G}_{\mathcal{P}}$, $B_P$, $S$
Initialize: $L_w \leftarrow 0$, $P' \leftarrow \varnothing$, $S' \leftarrow \varnothing$, $L_{prev} \leftarrow 0$
$P_t \leftarrow F(S)$
$\widehat{P}, J(A_{P'}) \leftarrow$ Get_Static_Attack($\mathcal{G}_{\mathcal{P}}$, $P_t$)
$L_w \leftarrow J(A_{P'})$, $P' \leftarrow \widehat{P}$
for k = 2, ..., B_P do
    $\widehat{P}_t \leftarrow$ Get_Contin($S$, $\widehat{P}$)
    $\widehat{P}, J(A_{P'}) \leftarrow$ Get_Static_Attack($\mathcal{G}_{\mathcal{P}}$, $\widehat{P}_t$)
    if $J(A_{P'}) > L_w$ then
        $L_w \leftarrow J(A_{P'})$, $P' \leftarrow \widehat{P}$
    end if
    if $(L_{prev} - L_w) \leqslant \varepsilon$ then
        break
    else
        $L_{prev} \leftarrow L_w$
    end if
end for
$S' \leftarrow$ Obtain_subs($S$, $P'$)
return $S'$, $P'$, $L_w$
```

As a first step, the algorithm identifies the maximum damage causing protection assemblies that can be manipulated from the entire set of protection assemblies using the method Get_Static_Attack($\mathcal{G}_{\mathcal{P}}$, P_t). The set of all protection assemblies can be obtained by using the function $F(S)$. depending upon the attacker budget B_P, for each iteration, a new set of protection assemblies that needs to be attacked in order to isolate power lines are obtained using Get_Contin(S, $\widehat{P}$). For instance, if an attacker has attacked a protection assembly $\widehat{P}$ from the set of substations S then in the next iteration, Get_Contin(S, $\widehat{P}$) uses the $\widehat{P}$ to return a new set of protection assemblies that can be attacked such that the attacker can choose only one new protection assembly from the total number of protection assemblies P in S that has not been previously attacked.

Similarly, in each iteration the algorithm selects the protection assemblies P' to manipulate from the attackable set of protection assemblies that are part of the selected S' in order to isolate transmission lines from the power network. Here, the function Get_Static_Attack($\mathcal{G}_{\mathcal{P}}$, $\widehat{P}_t$) identifies the protection assemblies that cause maximum damage and updates the solution if the damage L_p caused by the selected protection assemblies is greater than the worst-case static damage L_w, where $\widehat{P}_t$ represents the set of protection assemblies that are available for the attack. The function Get_Static_Attack($\mathcal{G}_{\mathcal{P}}$, $\widehat{P}_t$) is similar to Algorithm 4, however, it does not consider the time for scheduling the attacks. The algorithm terminates if no further improvement in system damage is observed. At the end, the substations S' that should be compromised in order to maximize system damage corresponding to the attacked protection assemblies are identified through direct mapping using the method Obtain_subs(S, P'). The worst-case running time of Algorithm 1 is non-exponential and is given by $O(|P| \times |B_P|)$.

4. Static defense model

In this section, first we provide the formulation of the defender model to improve the power system resilience by minimizing the damage/load loss. Then, we provide an efficient algorithm for identifying the critical substations to be protected in order to minimize the system damage considering the static attack model. Here, based on the substations and their components i.e. protection assemblies targeted by the attack, a set of critical substations to be protected is identified.

4.1. Defender's problem

The primary goal of a defender is to improve the power system resilience by protecting the critical substations in order to minimize the possible load loss when an attack is launched. To achieve this, the defender can protect a subset of substations D_S from the total number of substations S, i.e., $D_S \subseteq S$. Note that, the defender is resource bounded to protect critical substations and the budget for protecting substations can benefit the defender in two ways:

- The defender can prioritize and protect up to B_D substations due to financial budget constraints because it is impossible to protect and upgrade all the substations simultaneously.
- A strategic attacker would aim at maximizing the system damage by attacking the most critical substations. Hence, this model can provide important insight into which substations can be prioritized for the upgrade and protected first against the malicious adversarial attack.

Next, the defender strategically utilizes the defense budget to minimize the damage function $J(A_{P'})$ and the problem is formally described below.

Problem 2 (*Defender's Problem*). Given a power system network $\mathcal{G}_{\mathcal{P}}$, a defense budget B_D, a substation budget B_S, a protection assembly budget B_P, find a defense strategy to minimize the system load loss. Formally,

$$\underset{D_S}{\operatorname{argmin}} \max_{S' \subseteq S \setminus D_S} \max_{P' \subseteq F(S')} J(A_{P'}) \tag{6}$$

$$|D_S| \leqslant B_D \tag{7}$$

$$\begin{gathered} |S'| \leqslant B_S \\ \forall\, S', S'' \in S\colon S' \cap S'' = \varnothing \end{gathered} \tag{8}$$

$$\begin{gathered} |P'| \leqslant B_P \\ \forall\, P', P'' \in P\colon P' \cap P'' = \varnothing \end{gathered} \tag{9}$$

$$B_S \leqslant B_P \tag{10}$$

4.2. Algorithm for finding the critical substations to protect

Now, we describe Algorithm 2 that identifies the defense strategy against the worst-case static attack in detail. It starts with an empty set and strategically identifies the critical substations to protect one by one such that when an attacker launches an attack the overall system damage can be minimized. The algorithm takes the same inputs as Algorithm 1 with the defense budget B_D as an additional input. It then identifies the critical substations D_S to prioritize and protect to minimize the system damage when a static attack is launched.

Algorithm 2. Algorithm to Find Critical Substations to Protect: Get_Static_Defense($\mathcal{G}_{\mathcal{P}}$, B_P, B_D, S)

```
Input: $\mathcal{G}_{\mathcal{P}}, B_P, B_D, S$
Initialize: $s' \leftarrow \emptyset, D_S \leftarrow \emptyset, D_S^t \leftarrow \emptyset, L_m \leftarrow 100, L_{Prev} \leftarrow 100, L_H \leftarrow \emptyset$
$S', P', L_w \leftarrow$ Get_WSA($\mathcal{G}_{\mathcal{P}}, B_P, S$)
$L_H \cup L_{Prev}$
for $i = 1, ..., B_D$ do
    $L_m \leftarrow 100, flag \leftarrow 0$
    if $D_S^t \neq \emptyset$then
        $S', L_{Prev} \leftarrow$ Get_WSA1($\mathcal{G}_{\mathcal{P}}, B_P, S, D_S^t, \emptyset$)
        $L_H \cup L_{Prev}$
    end if
    for all $s \in S'$ do
        $L_s \leftarrow$ Get_WSA2($\mathcal{G}_{\mathcal{P}}, B_P, S, D_S^t, s$)
        if $L_s < L_m$ then
            $L_m \leftarrow L_s, s' \leftarrow s, flag \leftarrow 1$
        end if
    end for
    $D_S \leftarrow D_S \cup s', D_S^t \leftarrow D_S^t \cup s'$
    if $L_m > min(L_H)$ AND $flag = =1$ then
        $D_S \leftarrow D_S \backslash s'$
    else
        $D_S \leftarrow D_S^t$
    end if
end for
return $D_S$
```

First, the worst-case static attack is identified using Get_WSA($\mathcal{G}_{\mathcal{P}}$, B_P, S) that is illustrated as Algorithm 1. Next, for the first iteration when there are no critical substations in D_S^t to protect, we use the critical substations S' identified from the worst-case attack to identify the first substation to protect. D_S^t represents the intermediate solution set for substations to be protected in order to obtain a better solution. We iteratively protect each substation s in S' and evaluate the overall system damage post static attack using Get_WSA2($\mathcal{G}_{\mathcal{P}}$, B_P, S, D_S^t, s). The computed system damage in each iteration is used to select the substation to protect, i.e., $D_S \leftarrow D_S \cup D_S'$, $D_S^t \leftarrow D_S^t \cup s'$, where s' is the substation that is to be protected and is obtained in the i^{th} iteration. Note that, the function Get_WSA2($\mathcal{G}_{\mathcal{P}}$, B_P, S, D_S^t, s) is the same as in Algorithm 1, however, here the worst-case static attack is computed by eliminating the protected substations D_S^t and the substation s from the attackable list of substations, i.e., $S \backslash (D_S^t \cup s)$. Further, if the computed damage L_s' is smaller than the maximum damage $\hat{L}_w$, the solution is updated.

Additionally, for each next iteration, if the protected substations set D_S^t is non-empty then a new set of critical substations are identified using worst-case static attack function, i.e., Get_WSA1($\mathcal{G}_{\mathcal{P}}$, B_P, S, D_S^t, $\emptyset$). This function is also same as Algorithm 1, however, the protected substations D_S^t are removed from the attackable list of substations while executing the worst-case static attack on the power system model $\mathcal{G}_{\mathcal{P}}$. It ensures that once the substations are protected, the attacker can only launch the static attack on the remaining substations depending on the attack budget. The obtained attack can further be utilized to identify the substation to protect considering the defense budget constraints. In the algorithm L_H keeps a track of all the previous load losses obtained after protecting the substations in D_S^t and updates the final solution D_S depending upon the comparison of the obtained damage with the previous system damages. This ensures a better protection mechanism that provides an effective solution. The worst-case run time of Algorithm 2 is non-exponential and is given by $O(|S| \times |B_D| \times |P| \times |B_P|)$.

5. Dynamic attack model

Now we first formulate the dynamic attack model then we provide an efficient algorithm for identifying the worst-case dynamic attack that maximizes the system damage.

5.1. Worst-case dynamic attack

The objective of the malicious attacker is to destabilize the power system by maximizing the load loss. In order to achieve this, first the attacker can gain access to a subset of substations $S'(k) \subseteq S$ at different time instants k, where $k \in \{1, ..., T\}$. The attacker is resource bounded and can compromise up to B_S substations. Next, the adversary can identify the protection assemblies $P'(k) \subseteq F(S'(k))$ to manipulate within the selected substations in order to disconnect transmission lines from the power system network at different time instants k. Here, the attacker is again resource bounded, i.e., it can manipulate at most B_P protection assemblies. Note that the budget on protection assemblies is favourable to an attacker based on the reasons described in Section 3. Finally, the dynamic attack on a set of substations S' and protection assemblies P' at time step k is denoted by $A_{P'}(k)$. We compute the dynamic attack damage function as below:

$$J\left(A_{P'}(k), x(k)\right) = \frac{\sum_{j=1}^{n} Z_j(k)}{Z_T} \times 100, \forall\ I_j(k) = 0 \tag{11}$$

where $k \in \{1, ..., T\}$, $x(k)$, and $A_{P'}(k)$ represents the time step, system state, and the attack at time step k respectively. Hence, the attacker will try to maximize this damage function which is formally defined as follows.

Problem 3 (*Worst-Case Dynamic Attack***).** Given a power system network $\mathcal{G}_{\mathcal{P}}$, a substation budget B_S, and a protection assembly budget B_P, find a worst-case dynamic attack $A_{P'}(k)$ that maximizes the system damage. Formally,

$$\underset{\{S'(k)\}_{k=1}^{T}}{\text{argmax}} \quad \underset{(\{P'(k) \subseteq F(S'(k))\}_{k=1}^{T})}{\max} \sum_{k=1}^{T} J\left(A_{P'}(k), x(k)\right) \tag{12}$$

$$x(k) = \begin{cases} G(H(k)), & \text{if} \quad H(k) = \{A_{P'}(i)\}_{i=1}^{k-1} \\ g(H(k)), & \text{if} \quad H(k) = \emptyset \end{cases} \tag{13}$$

$$\sum_{k=1}^{T} \left| S'(k) \right| \leqslant B_S$$
$$\forall\ k, k' \in \{1, ...,T\}: S'(k) \cap S'(k') = \emptyset, k \neq k' \tag{14}$$

$$\sum_{k=1}^{T} \left| P'(k) \right| \leqslant B_P$$
$$\forall\ k, k' \in \{1, ...,T\}: P'(k) \cap P'(k') = \emptyset, k \neq k' \tag{15}$$

$$B_S \leqslant B_P \tag{16}$$

where $x(k)$ represents the state of the system at time step k, $H(k)$ represents the attack history of the system, $G(H(k))$, denote a function that returns the system state given an attack history $H(k)$, and $g(H(k))$ denote a function that returns normal system state with no attack history. Note that, $S'(k)$, $S'(k')$ represents the substations that are selected to be attacked, however once a substation is attacked at time step k it does not need to be attacked again at time step k'. Note that, a similar assumption holds for protection assemblies.

5.2. Algorithm for finding worst-case dynamic attack

This section first describes the main algorithm for finding the worst-case dynamic attack and then describes its subroutine in detail.

5.2.1. Get_WDA($\mathcal{G}_{\mathcal{P}}$, B_P, S, a_k)

The worst case dynamic attack, i.e., Get_WDA($\mathcal{G}_{\mathcal{P}}$, B_P, S, a_k) is illustrated as Algorithm 3. It is based on iteratively identifying the attacks that maximizes system damage at specific instants in time depending upon the budget constraints, i.e., B_S and B_P. Here, S denotes power system substation configuration, a_k denotes the possible attack time vector, L_w^d represents the worst-case dynamic damage and a_k^d represents the identified time vector at which the attack needs to be executed.

Algorithm 3. Algorithm for Finding Worst-Case Dynamic Attack: Get_WDA($\mathcal{G}_{\mathcal{P}}$, B_P, S, a_k)

```
**Input:** $\mathcal{G}_{\mathcal{P}}$, $B_P$, $S$, $a_k$
**Initialize:** $L_w^d \leftarrow 0$, $P'(k) \leftarrow \varnothing$, $S'(k) \leftarrow \varnothing$, $a_k^d \leftarrow \varnothing$, $a_k' \leftarrow 0$
$S'$, $P'$, $L_w \leftarrow$ Get_WSA($\mathcal{G}_{\mathcal{P}}$, $S$, $B_P$)
$S'(k) \leftarrow S'$, $P'(k) \leftarrow P'$, $L_w^d \leftarrow L_w$
**for all** p ∈ P′ **do**
    $a_k^d \leftarrow a_k^d \cup a_k'$
**end for**
**for all** p ∈ P′ **do**
    $P^d \leftarrow \varnothing$, $a^d \leftarrow a_k'$, $a_{temp}^d \leftarrow a^d$
    $P^d \leftarrow P^d \cup p$
    **for** $\mathrm{i} = 1, \ldots, (|\mathrm{P}'|)$ **do**
        $C \leftarrow$ Gen_Contin($P'$, $P^d$)
        $P^*$, $J(A_{P'}(k))$, $a^* \leftarrow$ Get_Dynamic_Attack($\mathcal{G}_{\mathcal{P}}$, $C$, $a_{temp}^d$, $a_k$)
        $P^d \leftarrow P^*$, $a_{temp}^d \leftarrow a^*$
        **if** $J(A_{P'}(k)) \geqslant L_w^d$ **then**
            $L_w^d \leftarrow J(A_{P'}(k))$, $P'(k) \leftarrow P^*$, $a_k^d \leftarrow a^*$
        **end if**
    **end for**
**end for**
$S'(k) \leftarrow$ Obtain_subs($S'$, $P'(k)$)
**return** $S'(k)$, $P'(k)$, $L_w^d$, $a_k^d$
```

First, we use Get_WSA($\mathcal{G}_{\mathcal{P}}$, S, B_P) to identify the worst-case static attack using Algorithm 1 described in Section 3. Here, we identify the maximum damage causing attack that provides the substations to compromise S', and the protection assemblies P' within the substations to manipulate in order to isolate the transmission lines from the power network assuming the attacks take place at the same time. The set of P' is iteratively used to generate a new set of contingencies C using Gen_Contin(P', P^d). The contingencies C are used by Get_Dynamic_Attack($\mathcal{G}_{\mathcal{P}}$, C, a_{temp}^d, a_k) (illustrated as Algorithm 4) which returns the maximum damage $J(A_{P'}(k))$ causing attack consisting of substations and associated protection assemblies P^* and the attack time vector a^*. In each iteration one attack is intelligently identified along with its time instant vector a_{temp}^d and added to the solution. Note that during the contingency generation process, P^* is utilized in such a way that the search space remain much smaller than the exhaustive search but still effective. Further, in each iteration, if the maximum damage $J(A_{P'}(k))$ obtained from Get_Dynamic_Attack($\mathcal{G}_{\mathcal{P}}$, C, a_{temp}^d, a_k) is larger than the worst-case dynamic damage L_w^d then the solution is updated. At the end, the method Obtain_subs(S', $P'(k)$) is used to obtain the direct mapping of the substations to be attacked. This is possible because the corresponding protection assemblies belong to the respective substations. This process reduces algorithm run time and provides an effective solution.

5.2.2. Get_Dynamic_Attack($\mathcal{G}_{\mathcal{P}}$, C, a_{temp}^d, a_k)

Now, we explain the subroutine Get_Dynamic_Attack($\mathcal{G}_{\mathcal{P}}$, C, a_{temp}^d, a_k) which is illustrated as Algorithm 4. Given a set of contingencies, Algorithm 4 identifies the protection assemblies one-by-one and the best sequence in which the attack can be executed to maximize the power system damage. Here, a_{temp}^d represents the attack time vector of set of contingencies in C. Note that, the attack vector a_{temp}^d of any contingency $C(i, j)$ represents the time instants of the previously attacked protection assemblies in $C(i, j)$. $C(i, j)$ represents a two-dimensional list of contingencies, i.e., protection assemblies that needs to be manipulated in order to isolate the transmission lines from the rest of the power network. Since protection assemblies are identified one-by-one and added to the solution, the maximum damage causing protection assembly that needs to be identified in any iteration will have an empty time instant ([]) in $C(i, j)$ before the algorithm is executed. Further, for any iteration in Algorithm 3, Algorithm 4 computes the maximum damage causing attack identifying the set of protection assemblies P^* to manipulate within the identified substations S', the system damage $J(A_{P'}(k))$ caused by the attack, and the time instants a^* at which the attacks need to be executed. Please note that, a^* here represent a set of positive integers that denote the instant at which the attack is executed.

Algorithm 4. Algorithm for Finding Dynamic Attack: Get_Dynamic_Attack($\mathcal{G}_{\mathcal{P}}$, C, a_{temp}^d, a_k)

```
**Input:** $\mathcal{G}_{\mathcal{P}}$, $C$, $a_{temp}^d$, $a_k$
**Initialize:** $J(A_{P'}(k)) \leftarrow 0$, $P^* \leftarrow \varnothing$, $a^* \leftarrow \varnothing$, $a_k^* \leftarrow \varnothing$, $P_{a_k} \leftarrow \varnothing$
**for** $\mathrm{i} = 1, \ldots, |\mathrm{C}|$ **do**
    Simulate_Model($\mathcal{G}_{\mathcal{P}}$)
    **for** $\mathrm{k} = 1, \ldots, |\mathrm{a_k}|$ **do**
        $P_{C(i,j)} \leftarrow \varnothing$, $k_c \leftarrow 0$, $a_{C(i,j)} \leftarrow \varnothing$
        **for** $\mathrm{j} = 1, \ldots, |\mathrm{a_{temp}^d}|$ **do**
            **if** $a_{temp}^d(j) = 0$ **then**
                Isolate_Branches($\mathcal{G}_{\mathcal{P}}$, $C(i, j)$)
                $a_{C(i,j)} \leftarrow a_{C(i,j)} \cup a_{temp}^d(j)$
                $P_{C(i,j)} \leftarrow P_{C(i,j)} \cup C(i, j)$
            **end if**
        **end for**
        Simulate_Contin($\mathcal{G}_{\mathcal{P}}$, $P_{C(i,j)}$, $a_{C(i,j)}$)
        $e \leftarrow 1$
        **while** $\mathrm{e} = 1$ **do**
            $e \leftarrow 0$, $k_c \leftarrow k_c + 1$
            $c \leftarrow$ Get_Branches($\mathcal{G}_{\mathcal{P}}$, $C(i, j)$)
            **if** $|\mathrm{c}| \neq 0$ **then**
                **for** $\mathrm{y} = 1, \ldots, |\mathrm{c}|$ **do**
                    Isolate_Branches($\mathcal{G}_{\mathcal{P}}$, $c(y)$)
                **end for**
                $e \leftarrow 1$
            **end if**
            **for** $\mathrm{j} = 1, \ldots, |\mathrm{a_{temp}^d}|$ **do**
                **if** $\mathrm{k_c} = \mathrm{a_{temp}^d(j)}$ **then**
                    Isolate_Branches($\mathcal{G}_{\mathcal{P}}$, $C(i, j)$)
                    $P_{C(i,j)} \leftarrow P_{C(i,j)} \cup C(i, j)$
                    $a_{C(i,j)} \leftarrow a_{C(i,j)} \cup a_{temp}^d(j)$
                **end if**
            **end for**
            **if** $\mathrm{k_c} = \mathrm{a_k(k)}$ **then**
                Isolate_Branches($\mathcal{G}_{\mathcal{P}}$, $C(i, |C(i)| - 1)$)
                $P_{a_k} \leftarrow C(i, |C(i)| - 1)$, $a_k^* \leftarrow k_c$
            **end if**
            Simulate_Contin($\mathcal{G}_{\mathcal{P}}$, $P_{C(i,j)} \cup P_{a_k}$, $a_{C(i,j)} \cup a_k^*$)
            $L_l \leftarrow$ Get_Loads($\mathcal{G}_{\mathcal{P}}$, $P_{C(i,j)} \cup P_{a_k}$, $a_{C(i,j)} \cup a_k^*$)
            $L_C \leftarrow$ Get_Damage($\mathcal{G}_{\mathcal{P}}$, $L_l$)
        **end while**
        **if** $\mathrm{L_C} > \mathrm{J(A_{P'}(k))}$ **then**
            $J(A_{P'}(k)) \leftarrow L_C$, $P_t \leftarrow P_{C(i,j)}$, $P_i \leftarrow P_{a_k}$
            $a_C' \leftarrow a_k^*$, $a_C \leftarrow a_{C(i,j)}$
            end if
            Simulate_Model($\mathcal{G}_{\mathcal{P}}$)
        end for
    end for
    $P^* \leftarrow P_t$, $a^* \leftarrow a_C$
    $P^* \leftarrow P^* \cup P_i$, $a^* \leftarrow a^* \cup a'_C$
    return $P^*$, $J(A_{P'}(k))$, $a^*$
```

Further, for each contingency, the algorithm first simulates the power system in its nominal state, i.e., without any attack. Then, depending upon a contingency $C(i, j)$ and the attack vector a^d_{temp}, all the transmission lines associated with $C(i, j)$ are removed from the power network for which the time instants are '0', i.e., initial attack. The power system $\mathcal{G}_{\mathcal{P}}$ is then simulated with the initial attack and is further evaluated for the secondary effects of this attack, i.e., additional system overloads. If there are any overloaded transmission lines they are identified and removed from the power network. Additionally, if there are any other attacks in $C(i, j)$ that are available to be executed using the attack vector a^d_{temp} at any other time instants they are also identified and executed. Next, the algorithm uses the time instant vector a_k to manipulate the protection assembly with empty time instant to isolate the associated transmission line such that it maximizes the system damage. The power system model is then simulated with the contingencies $(P_{C(i,j)} \cup P_{a_k})$ and its associated attack vector $(a_{C(i,j)} \cup a^*_k)$. Next, the amount of system damage caused by the attack is computed for every contingency set in C. If the computed load L_C is larger than the maximum damage $J(A_{P'}(k))$ in any iteration, the solution is updated. Note that, after evaluating each contingency set in C, the power system model is set back to its nominal state.

6. Dynamic defense model

Next, we first formulate the defender model and then we provide an efficient algorithm for identifying the critical substations to be protected to minimize the system damage.

6.1. Defender's problem

The objective of the defender is to improve the power system resilience by minimizing the possible load loss. In order to achieve this, the defender can protect a subset of substations D_S from the total number of substations S in the power system network, i.e., $D_S \subseteq S$. Further, due to financial budget constraints, the defender is resource bounded and can prioritize and protect at most B_D substations. Note that the defense budget on substations can support the defender in the similar way as discussed in Section 4. Next, the defender strategically utilizes the defense budget to minimize the damage function $J(A_{P'}(k), x(k))$ and the problem is formally described below.

Problem 4 (*Defender's Problem*). Given a power system network $\mathcal{G}_{\mathcal{P}}$, a defense budget B_D, a substation budget B_S, a protection assembly budget B_P, find a defense strategy to minimize the damage/load loss when an attacker launches a dynamic attack at different time instants k. Formally,

$$\underset{D_S}{\text{argmin}} \quad \max_{\{(S'(k) \subseteq S \setminus D_S)} \quad \max_{(P'(k) \subseteq F(S'(k)))\}_{k=1}^{T}} \sum_{k=1}^{T} J\left(A_{P'}(k), x(k)\right) \tag{17}$$

$$x(k) = \begin{cases} G(H(k)), & \text{if} \quad H(k) = \{A_{P'}(i)\}_{i=1}^{k-1} \\ g(H(k)), & \text{if} \quad H(k) = \varnothing \end{cases} \tag{18}$$

$$|D_S| \leqslant B_D \tag{19}$$

$$\sum_{k=1}^{T} \left| S'(k) \right| \leqslant B_S$$
$$\forall\ k,\ k' \in \{1, \ldots\ldots, T\}\text{: } S'(k) \cap S'(k') = \varnothing,\ k \neq k' \tag{20}$$

$$\sum_{k=1}^{T} \left| P'(k) \right| \leqslant B_P$$
$$\forall\ k,\ k' \in \{1, \ldots\ldots, T\}\text{: } P'(k) \cap P'(k') = \varnothing,\ k \neq k' \tag{21}$$

$$B_S \leqslant B_P \tag{22}$$

where $x(k)$ represents the state of the system at time step k and $H(k)$ represents the attack history of the system.

6.2. Algorithm for finding the critical substations to protect

Now, we describe Algorithm 5 that identifies the defense strategy against the worst-case dynamic attack in detail. Algorithm 5 starts with an empty set and intelligently identifies the critical substations to protect one by one such that when an attack is launched the overall system damage can be minimized. The algorithm takes the same inputs as Algorithm 3 with the defense budget B_D as an additional input and identifies the critical substations D_S to protect.

Algorithm 5. Algorithm for Finding Critical Substations to Protect: Get_Dynamic_Defense($\mathcal{G}_{\mathcal{P}}$, B_P, B_D, S, a_k)

```
Input: $\mathcal{G}_{\mathcal{P}}$, $B_P$, $B_D$, $S$, $a_k$
Initialize: $s' \leftarrow \varnothing$, $D_S \leftarrow \varnothing$, $L_m \leftarrow 100$, $L_{Prev} \leftarrow 100$, $L_H \leftarrow \varnothing$
$S'(k)$, $P'(k)$, $L^d_w$, $a^d_k \leftarrow$ Get_WDA($\mathcal{G}_{\mathcal{P}}$, $B_P$, $S$, $a_k$)
$L_H \cup L_{Prev}$
for i = 1, ..., $\mathrm{B_D}$ do
    $L_m \leftarrow 100$, $flag \leftarrow 0$
    if $D^t_S \neq \varnothing$ then
        $\hat{S}'(k)$, $L_{Prev} \leftarrow$ Get_WDA1($\mathcal{G}_{\mathcal{P}}$, $B_P$, $S$, $a_k$, $D^t_S$, $\varnothing$)
        $L_H \cup L_{Prev}$
    end if
    for all $\mathrm{s} \in \mathrm{S'(k)}$ do
        $L'_s \leftarrow$ Get_WDA2($\mathcal{G}_{\mathcal{P}}$, $B_P$, $S$, $a_k$, $D^t_S$, $s$)
        if $L'_s < L_m$ then
            $L_m \leftarrow L'_s$, $s' \leftarrow s$, $flag \leftarrow 1$
        end if
    end for
    $D_S \leftarrow D_S \cup s'$, $D^t_S \leftarrow D^t_S \cup s'$
    if $L_m > min(L_H)$ AND $flag = =1$ then
        $D_S \leftarrow D_S \setminus s'$
    else
        $D_S \leftarrow D^t_S$
    end if
end for
return $D_S$
```

First, the worst-case dynamic attack is identified by using Get_WDA($\mathcal{G}_{\mathcal{P}}$, B_P, S, a_k) which is illustrated as Algorithm 3. Next, if there are no critical substations in D_S, We use the critical substations $S'(k)$ identified from the worst-case dynamic attack to identify the first substation to protect. We iteratively protect each substation in $S'(k)$ and evaluate the overall system damage post dynamic attack using Get_WDA2($\mathcal{G}_{\mathcal{P}}$, B_P, S, a_k, D^t_S, s). The computed system damage in each iteration is used to select the substation to protect, i.e., $D_S \leftarrow D_S \cup s'$. A track of intermediate solution $D^t_S \leftarrow D^t_S \cup s'$ is kept in order to obtain a better solution. Note that, the function Get_WDA2($\mathcal{G}_{\mathcal{P}}$, B_P, S, a_k, D^t_S, s) is same as Algorithm 3, however, here the worst-case dynamic attack is computed by eliminating the protected substations D^t_S and the substation s from the attackable list of substations, i.e., $S \setminus (D^t_S \cup s)$. Further, if the computed damage L'_s is smaller than the maximum damage $\hat{L}_w$, the solution is updated.

Additionally, for next each iteration, if the protected substations set D^t_S is non-empty then a new set of critical substations are identified using the worst-case dynamic attack function, i.e., Get_WDA1($\mathcal{G}_{\mathcal{P}}$, B_P, S, a_k, D^t_S, $\varnothing$). This function is also same as Algorithm 3, however, the protected substations D^t_S are removed from the attackable set of substations while executing the worst-case dynamic attack on the power system model $\mathcal{G}_{\mathcal{P}}$. It ensures that once the substations are protected, the attacker can only launch the dynamic attack on the remaining substations based on the attack budget. The obtained attack can further be utilized to identify the substation to protect considering the defense budget constraints. In addition, inside the algorithm L_H keeps a track of all the previous load losses obtained after protecting the substations in D_S^t and updates the final solution D_S depending upon the comparison of the obtained damage with the previous system damages. This ensures a better protection mechanism that provides an effective solution.

7. Evaluation

In this section, we evaluate our approach using two IEEE standard systems: the 39 and the 57 bus systems. We used a modified version of the steady state simulator discussed in [15] to perform the analysis. First, we discuss how randomly chosen attacks can be optimized using our dynamic attack model and then we show the optimization of the worst-case static attacks using the dynamic attack model. Next, we present the dynamic defense results that show the reduction in the overall system damage/load loss. Please note that, we have segregated each power system model into substations and its protection assemblies similar to the one shown in Fig. 1 and used Eqs. (1) and (11) for corresponding attack models to compute the load loss via simulating the standard IEEE system models using Opendss. Further, we evaluate the performance of our algorithm's execution time for the dynamic attack and defense algorithms in comparison with the naive exhaustive search algorithm. Finally, we discuss the optimality of our approach as compared to the exhaustive search algorithms.

7.1. Optimizing random attacks

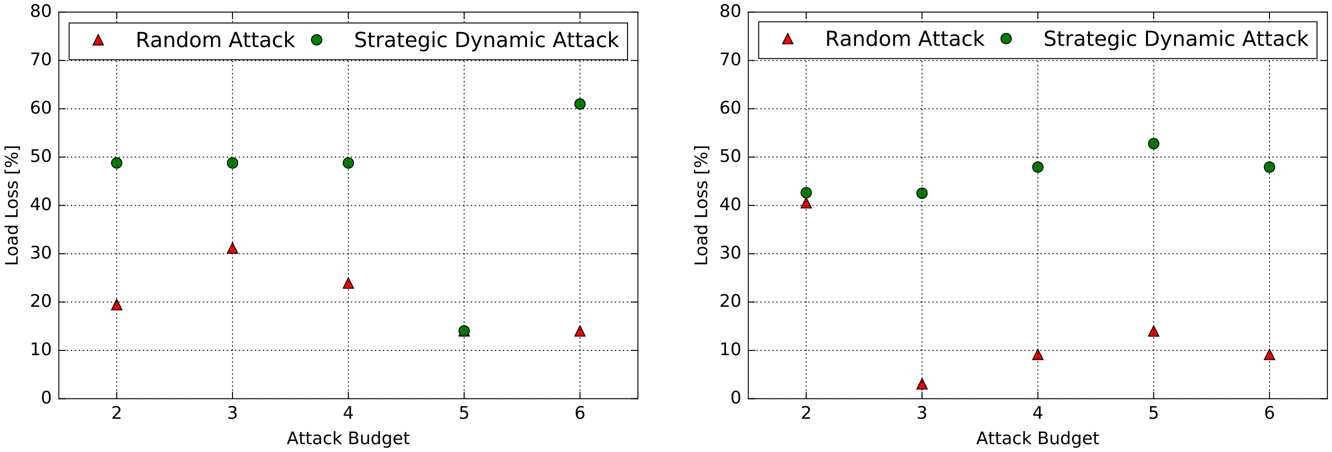
Fig. 2 shows the optimization of the random attacks using the dynamic attack model discussed in Section 5. In our approach, random attacks are those static attacks that are randomly selected by an attacker but are not worst-case static attacks. We show that our dynamic attack algorithm is able to maximize system damage on these randomly selected attacks. Here, depending upon the attack budget (up to 6), we randomly picked the components to attack from the power system model. Then, we used these attacks as inputs to our dynamic attack algorithm to obtain a strategic sequence in which the attacks can be executed so as to maximize the system damage. We performed our evaluation on the IEEE 39 and 57 bus systems and the results are shown in Fig. 2. The x-axis represents the attack budget (B_S/B_P) whereas the y-axis represents the system damage, i.e., load loss. Red, green color markers represent the random and strategic dynamic attacks respectively.

For both standard IEEE systems, we can clearly see from Fig. 2a and b that our dynamic attack algorithms described in this paper are able to strategically identify the specific instants (or sequences) at which different attacks can be executed and maximize the system damage for a randomly identified set of components to attack. From Fig. 2a, for an attack budget of 6 in IEEE-39 bus system the random attack caused a load loss of 14.03%, however, the same attack when executed at different instants in time, i.e., dynamic attack resulted in a total load loss of 60.99%. The dynamic attack on the same components caused a 334.71% higher load loss than the static attack. For the same attack budget in the IEEE-57 bus system the random attack caused a load loss of 9.16%, whereas, the dynamic attack resulted in a load loss of 47.93% as shown in Fig. 2b. This dynamic attack load loss is 423.25% higher than the random attack.

7.2. Optimizing static attacks

To demonstrate the optimization of the static attacks, We perform the analysis on the same standard IEEE systems. In this approach first, we identified the worst-case static attack and then we use this attack to identify the worst-case dynamic attack in order to further maximize the system damage. Fig. 3 shows the results for the optimization of the worst-case static attack using our dynamic attack model and algorithm. The x-axis represents the attack budget (B_S/B_P), whereas, the y-axis represents the system damage. Red, green colored markers represent the worst-case static and dynamic attacks respectively. Here, we consider an attack budget of up to 6 components/substations.

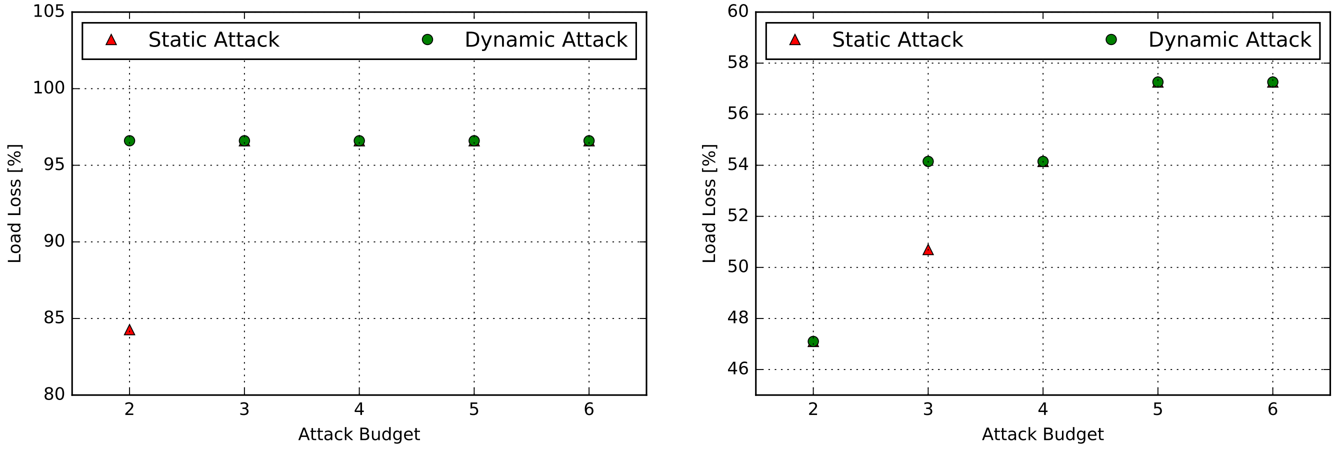
From Fig. 3a and b it is clear that the dynamic attack causes higher damage with various attack budgets. As shown in Fig. 3a, for an attack budget of 2 in IEEE-39 bus system the worst-case static attack caused a load loss of 84.27%, however, the optimized worst-case dynamic attack resulted in a load loss of 96.60%. Here, the dynamic attack on the same components caused a 14.63% higher load loss. Similarly, for the IEEE-59 bus system in Fig. 3b, the worst-case static attack caused a load loss of 50.70%, whereas, the optimized worst-case dynamic attack resulted in a load loss of 54.15% for an attack budget of 3. The dynamic attack caused a higher load loss by 6.80%. Note that the worst-case static attacks are already identified as the attacks that cause maximum damage, however our dynamic attack algorithms are still able to optimize them for obtaining even higher system damage if there is a possibility for optimization. The dynamic attack algorithm results from Fig. 3 clearly show that the dynamic attacks on the same components that are identified from the static attack scenario when scheduled and executed strategically resulted in a higher system damage. Note that, in Fig. 3,



(a) IEEE-39 bus system

(b) IEEE-57 bus system

Fig. 2. Random Attacks Vs Dynamic Attacks: Load loss as a function of various attack budgets for different standard IEEE systems.



(a) IEEE-39 bus system

(b) IEEE-57 bus system

Fig. 3. Static Attacks Vs Dynamic Attacks: Load loss as a function of various attack budgets for different standard IEEE systems.

the static and dynamic attack load loss becomes equal for some attack budgets because there is no additional load loss possible within the system. Also note that, for some attack budgets the difference in the load loss between the static attack and the dynamic attack can remain very small because the additional loads that gets disconnected during the dynamic attack maybe smaller in magnitude as compared to the total load loss. However, if the additional load loss is larger in magnitude, then this difference can be significantly larger as shown by attack budget 2, 3 in Fig. 3a and b respectively.

In order to further demonstrate optimization process in detail, we have shown the exact cascade progression using IEEE-39 bus system for one of the static and dynamic attack scenarios presented in Table 4 that can easily answer the question of ‘how dynamic attacks can have higher impact?’. Here, for both the attack scenarios, we consider the same substations and its components to attack, but the only difference is the attack time. First, for the static attack scenario with an attack budget of 2, Table 4 shows that both the attacks are launched at the same time [0, 0] ([0, 0] indicates simultaneous attack or static attack). As a result of the static attack the transmission lines associated with the attacked protection assemblies are isolated. This resulted in a sequence of cascading failures as shown by the ‘Stage 1 Outages’ through ‘Stage 4 Outages’ in Table 4 and the total system load loss was observed to be

Table 4
Scenario representing the maximization of system damage using dynamic attack model.

Static Attack		Dynamic Attack	
Initial Attack	Attack time vector: [0, 0] Substations compromised: [S^{13}, S^{24}] Protection assemblies attacked: [PA10, PA16] Transmission lines Isolated due to the attacked protection assemblies: [‘R16_19’, ‘R2_3’] Load loss: ‘0%’	Initial Attack	Attack time vector: [0] Substations compromised: [S^{24}] Protection assemblies attacked: [PA16] Transmission lines Isolated due to the attacked protection assemblies: [‘R2_3’] Load loss: ‘0%’
Stage 1 Outages	Isolation of transmission lines due to the secondary effect of the outages from the initial attack: [‘R2_25, R25_26, R18_17, R27_26’], Load loss: ‘0%’	Stage 1 Outages	Isolation of transmission lines due to the secondary effect of the outages from the initial attack: [‘R2_25, R18_17’], Load loss: ‘0%’
Stage 2 Outages	Isolation of transmission lines due to the secondary effect of the outages from the stage 1: [‘R6_5, R14_15, R14_13, R10_13, R26_28, R21_22] Load loss: ‘35.48%’	Additional Attack	Attack time vector: [1] Substations compromised: [S^{13}] Protection assemblies attacked: [PA10] Transmission lines Isolated due to the attacked protection assemblies: [‘R16_19’] Load loss: ‘0%’
Stage 3 Outages	Isolation of transmission lines due to the secondary effect of the outages from the stage 2: [‘R8_7, R6_7, R10_11, R6_11] Load loss: ‘64.80%’	Stage 2 Outages	Isolation of transmission lines due to the secondary effect of the outages from the stage 1: [‘R6_5’], Load loss: ‘0%’
Stage 4 Outages	Isolation of transmission lines due to the secondary effect of the outages from the stage 3: [‘R9_39, R8_9’], Load loss: ‘84.27%’	Stage 3 Outages	Isolation of transmission lines due to the secondary effect of the outages from the stage 2: [‘R8_7, R6_7, R4_14, R14_13, R10_13’], Load loss: ‘7.25%’
		Stage 4 Outages	Isolation of transmission lines due to the secondary effect of the outages from the stage 3: [‘R9_39, R8_9, R21_22, R24_23’], Load loss: ‘56.42%’
		Stage 5 Outages	Isolation of transmission lines due to the secondary effect of the outages from the stage 3: [‘R25_26, R17_27, R27_26, , R16_17, R26_28, R28_29, R26_29, R16_21’], Load loss: ‘96.60%’

84.27%.

Now, we consider the same substations and protection assemblies for the dynamic attack scenario. Here, the initial attack takes place at time instant 0 that initiated a cascading event causing subsequent failures (Stage 1 Outages in Table 4). At time instant 1, another attack was launched that further weakened the system causing Outages through Stage 2 to Stage 5 resulting in a significant damage to the system. The overall system load loss was observed to be 96.60% (Stage 2 and Stage 5 Outages in Table 4) which is considerably higher than the static attack. Note that the specific time at which these attacks can be executed are computed using the algorithms described in Section 5.

7.3. Minimizing system damage using dynamic defense

Now, we demonstrate that the power systems damage can be significantly reduced by intelligently prioritizing and protecting critical system substations while considering limited defense budget. We evaluate our defense model and algorithm using the standard IEEE-39 and 57 bus systems. Fig. 4 shows the load losses in the power system at different attack budgets when a dynamic attack is launched after the critical substations are intelligently identified and protected depending upon the defense budget. In each figure, the x-axis represents the defense budget and the y-axis represent the total system damage. Red, green, blue, and yellow colored markers represents the attack budgets 2, 3, 4 and 5 respectively. Further, the respective color markers at the defense budget 0 represent the total system damage without any defense.

Now, from Fig. 4, we can clearly see that by intelligently selecting and protecting the critical substations of the power network, the system damage can be significantly reduced for IEEE-39 bus system (Fig. 4a) and 57 bus system (Fig. 4b) when a dynamic attack is launched. In Fig. 4a, for an attack and a defense budget of 2, the load loss is reduced from 96.60% to 84.27%, that is, a total of 12.76% reduction in load loss. Moreover, for the same attack budget and a defense budget of 18, a total of 88.58% reduction in load loss is observed. For other attack budgets, as the defense budget increases we can see significant improvement in the reduction of total system load loss.

7.4. Performance of the dynamic attack and defense algorithms

Finally, we compare the execution time of our dynamic attack and defense algorithms with the naive exhaustive search algorithms. We use the same standard IEEE systems to perform our analysis. Fig. 5 shows the dynamic attack and defense execution time with respect to the exhaustive search. In each figure, the x-axis represents either the attack budget (B_S/B_P) or the defense budget and the y-axis represents the time taken by the algorithm to identify the attack or defense. The details of the markers are shown in the legend box of Fig. 5.

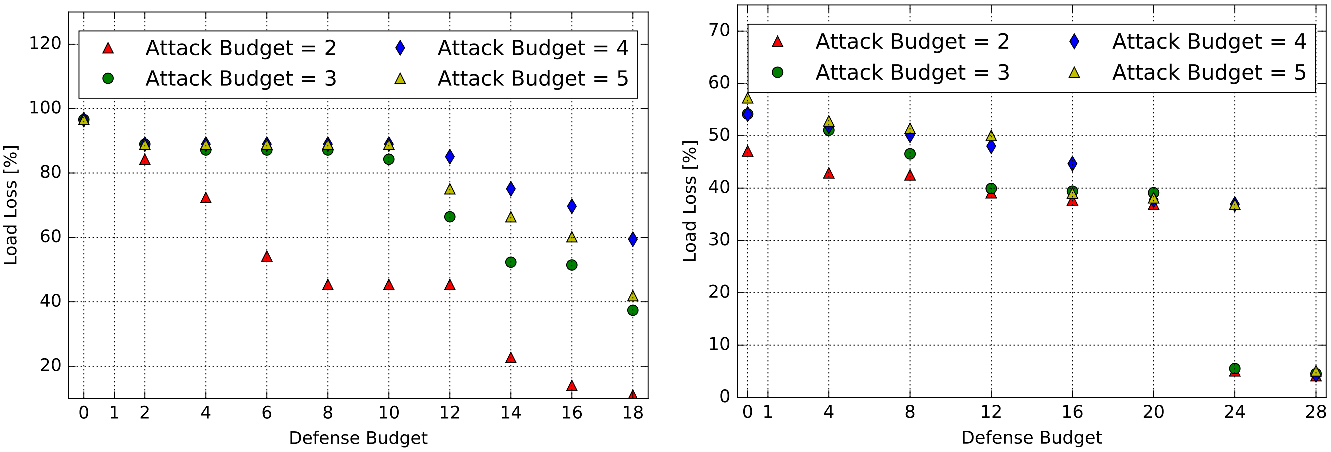
Here, from Fig. 5a, we can clearly see that the time taken to identify the dynamic attack for IEEE-39, 57 bus system increases very slightly with increase in the attack budget. However, the time taken to identify the attack using the exhaustive search algorithm is observed to be exponential even at smaller attack budgets. The exhaustive search execution time in Fig. 5a represents the time taken to identify the maximum damage causing static attack. Moreover, the exhaustive search's execution time for identifying the maximum damage causing dynamic attack will be much larger than the time taken to identify the static attack. Similarly, it is clear from Fig. 5b that the time taken to identify the defense increases slowly with the increase in the defense budget. Further, we know that dynamic defense via exhaustive analysis will take much longer than the exhaustive attack since it will have to first identify the attack and then identify the defense. Hence, if we compare only against the attack time, it still shows that the developed approach is much faster than the exhaustive search. Therefore, as demonstrated in Fig. 5, our algorithms prove to be far more efficient than the naive exhaustive search.

7.5. Optimality of our proposed algorithms

Several research works in the past [14,15] have theoretically and experimentally proved that identifying critical contingencies including cyber-physical attacks require extensive computational resources due to the search space explosion. Therefore, optimal solution via exhaustive search remains infeasible. However, we performed simulations and compared our proposed approach against exhaustive search algorithms for those scenarios that were computationally feasible. The results obtained via simulations demonstrated that our proposed heuristic algorithms are able to obtain exact solutions, i.e., optimal solutions that match the exhaustive search solutions in most scenarios. Further, in some scenarios the solutions obtained via our proposed approach differ by an average of only 0.355% when compared against the exhaustive search algorithms. This demonstrates that the developed algorithms are able to obtain optimal solutions and require significantly less computational time and effort.

8. Conclusions and future work

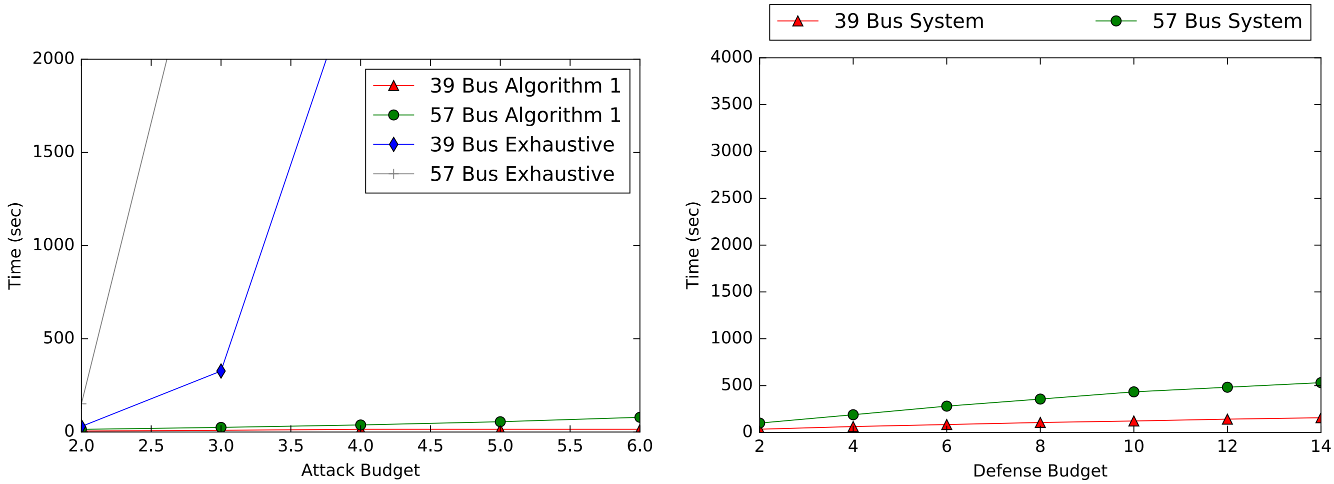
In this paper, we have described the static and dynamic cyber-attack and defense models for electrical power systems using game-theoretic approach. From the attacker's perspective, we provided an efficient and effective algorithm that is able to strategically identify the dynamic attacks that maximizes the system damage by considering both random



(a) IEEE-39 bus system

(b) IEEE-57 bus system

Fig. 4. Dynamic Defense: Load loss as a function of various defense budgets for different standard IEEE systems.



(a) Execution time of attack analysis

(b) Execution time of defense analysis

Fig. 5. Analysis execution time for attack and defense for different standard IEEE systems.

attacks as well as worst-case static attacks. We also provided an efficient algorithm from defenders perspective that identifies the critical substations to protect in order to minimize the overall system damage. Our results shows that, under budget constraints, intelligently selecting the substations to prioritize and protect can significantly improve the power system resilience. In addition, these algorithms are efficient and perform significantly better than the exhaustive search even with the complex dynamic attack and defense models.

As part of the future work, the attacker-defender models can be easily extended to consider randomness, i.e., a success probability can be associated with an attack and a defense that can give us more insight to improve the power system resilience under probabilistic scenarios. Further, under unknown circumstances where the defender has no idea whether an attacker follows a static attack model or a dynamic attack model, a defense strategy that could improve the overall power system resilience irrespective of the attack model can be an interesting direction to explore.

Declaration of Competing Interest

The authors have no conflict of interest to declare.

Acknowledgment

This work is funded in part by the NSF under the award number CNS-1329803 and the NSF FORCES project under the award number CNS-1238959. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF or FORCES. The authors would like to thank Amin Ghafouri and Charles Hartsell for their help and discussions related to the work presented here.

References

[1] Council NR, et al. Terrorism and the electric power delivery system. National Academies Press; 2012.
[2] Hawrylak PJ, Haney M, Papa M, Hale J. Using hybrid attack graphs to model cyber-physical attacks in the smart grid. 2012 5th international symposium on resilient control systems (ISRCS). IEEE; 2012. p. 161–4.
[3] David J. Double threat: us grid vulnerable on two fronts. CNBC; 2014. Retrieved from http://www.cnbc.com/id/101306145.
[4] Draffin Jr. CW. Cybersecurity white paper; 2016.
[5] A cyberattack on the U.S. power grid. https://www.cfr.org/report/cyberattack-us-power-grid/. Council on Foreign Relations; 2017.
[6] Rawat DB, Bajracharya C. Cyber security for smart grid systems: status, challenges and perspectives. 2015 SoutheastCon. IEEE; 2015. p. 1–6.
[7] Wang W, Lu Z. Cyber security in the smart grid: survey and challenges. Comput Netw 2013;57(5):1344–71.
[8] Gorman S. Electricity grid in us penetrated by spies. Wall Street J 2009;8.
[9] Q.T. Review, "Enabling modernization of the electric power system," U.S. Department of Energy; 2015.
[10] Sridhar S, Hahn A, Govindarasu M. Cyber–physical system security for the electric power grid. Proc IEEE 2012;100(1):210–24.
[11] Pultarova T. Cyber security-Ukraine grid hack is wake-up call for network operators [news briefing]. Eng Technol 2016.
[12] NERC S. Top-004–2: transmission operations. North American Electric Reliability Corporation; 2007.
[13] Glover JD, Sarma MS, Overbye T. Power system analysis & design. SI version. Cengage Learning; 2012.
[14] Eppstein MJ, Hines PD. A random chemistry algorithm for identifying collections of multiple contingencies that initiate cascading failure. IEEE Trans Power Syst 2012;27(3):1698–705.
[15] Hasan S, Ghafouri A, Dubey A, Karsai G, Koutsoukos X. Heuristics-based approach for identifying critical n-k contingencies in power systems. 2017 resilience week (RWS). IEEE; 2017. p. 191–7.
[16] Liu S, Mashayekh S, Kundur D, Zourntos T, Butler-Purry K. A framework for modeling cyber-physical switching attacks in smart grid. IEEE Trans Emerg Top Comput 2013.
[17] Yang Y, McLaughlin K, Littler T, Sezer S, Im EG, Yao Z, et al. Man-in-the-middle attack test-bed investigating cyber-security vulnerabilities in smart grid scada systems; 2012.
[18] Lin H, Alemzadeh H, Chen D, Kalbarczyk Z, Iyer RK. Safety-critical cyber-physical attacks: analysis, detection, and mitigation. Proceedings of the symposium and bootcamp on the science of security. ACM; 2016. p. 82–9.
[19] Sridhar S, Manimaran G. Data integrity attack and its impacts on voltage control loop in power grid. In: PES general meeting; 2011.
[20] Hao J, Piechocki RJ, Kaleshi D, Chin WH, Fan Z. Sparse malicious false data injection attacks and defense mechanisms in smart grids. IEEE Trans Industr Inf 2015.
[21] Yuan Y, Li Z, Ren K. Quantitative analysis of load redistribution attacks in power systems. IEEE Trans Parallel Distrib Syst 2012;23(9):1731–8.
[22] Turau V, Weyer C. Cascading failures caused by node overloading in complex networks. Joint workshop on cyber-physical security and resilience in smart grids (CPSR-SG). IEEE; 2016. p. 1–6.
[23] Chen B, Mashayekh S, Butler-Purry KL, Kundur D. Impact of cyber attacks on transient stability of smart grids with voltage support devices. Energy society general meeting (PES). 2013.
[24] Hasan S, Chhokra A, Dubey A, Mahadevan N, Karsai G, Jain R, et al. A simulation testbed for cascade analysis. 2017 power & energy society innovative smart grid technologies conference (ISGT). IEEE; 2017. p. 1–5.
[25] Poudel S, Ni Z, Malla N. Real-time cyber physical system testbed for power system security and control. Int J Electr Power Energy Syst 2017;90:124–33.
[26] Yan J, Zhu Y, He H, Sun Y. Multi-contingency cascading analysis of smart grid based on self-organizing map. IEEE Trans Inf Forensics Secur 2013;8(4):646–56.
[27] Yuan W, Zhao L, Zeng B. Optimal power grid protection through a defender–-attacker–defender model. Reliab Eng Syst Saf 2014;121:83–9.
[28] Hausken K, Levitin G. Minmax defense strategy for complex multi-state systems. Reliab Eng Syst Saf 2009;94(2):577–87.
[29] Xiang Y, Wang L, Liu N. Coordinated attacks on electric power systems in a cyber-physical environment. Electr Power Syst Res 2017;149:156–68.
[30] Liu S, Chen B, Zourntos T, Kundur D, Butler-Purry K. A coordinated multi-switch attack for cascading failures in smart grid. IEEE Trans Smart Grid 2014;5(3):1183–95.
[31] Zhu Y, Yan J, Tang Y, Sun YL, He H. Joint substation-transmission line vulnerability assessment against the smart grid. IEEE Trans Inf Forensics Secur 2015;10(5):1010–24.
[32] Law YW, Alpcan T, Palaniswami M. Security games for risk minimization in

automatic generation control. IEEE Trans Power Syst 2015;30(1):223–32.

[33] Zhang Z, Gong S, Dimitrovski AD, Li H. Time synchronization attack in smart grid: Impact and analysis. IEEE Trans Smart Grid 2013;4(1):87–98.

[34] Liu Y, Ning P, Reiter MK. False data injection attacks against state estimation in electric power grids. ACM Trans Inform Syst Secur (TISSEC) 2011;14(1):13.

[35] Yan J, Zhu Y, He H, Sun Y. Revealing temporal features of attacks against smart grid. 2013 IEEE PES innovative smart grid technologies (ISGT). IEEE; 2013. p. 1–6.

[36] Zhu Y, Yan J, Tang Y, Sun YL, He H. Resilience analysis of power grids under the sequential attack. IEEE Trans Inf Forensics Secur 2014;9(12):2340–54.

[37] Liscouski B, Elliot W. Final report on the august 14, 2003 blackout in the united states and canada: causes and recommendations. A report to US Department of Energy, vol. 40, no. 4; 2004.

[38] Pidd H. India blackouts leave 700 million without power. Guardian 2012;31.

[39] http://icseg.iti.illinois.edu/power-cases/, ICSEG.

[40] Hasan S, Ghafouri A, Dubey A, Karsai G, Koutsoukos X. Vulnerability analysis of power systems based on cyber-attack and defense models. 2018 IEEE Power & energy society innovative smart grid technologies conference (ISGT). IEEE; 2018. p. 1–5.