Algorithmic Attack Synthesis using Hybrid Dynamics of Power Grid Critical Infrastructures

Abstract—**Automated vulnerability assessment and exploit generation for computing systems have been explored for decades. However, these approaches are incomplete in assessing industrial control systems, where networks of computing devices and physical processes interact for safety-critical missions. We present[1] an attack synthesis algorithm against such cyber-physical electricity grids. The algorithm explores both discrete network configurations and continuous dynamics of the plant's embedded control system, to search for attack strategies that evade detection with conventional monitors. The algorithm enabling this exploration is rooted in recent developments in the *hybrid system* verification research: it effectively approximates the behavior of the system for a set of possible attacks by computing sensitivity of the system's response to variations in the attack parameters. For parts of the attack space, the proposed algorithm can infer whether or not there exists a feasible attack that avoids triggering protection measures such as relays and steady-state monitors. The algorithm can take into account constraints on the attack space such as the power system topology and the set of controllers across the plant that can be compromised without detection. With a proof-of-concept prototype, we demonstrate the synthesis of transient attacks in several typical electricity grids and analyze the robustness of the synthesized attacks to perturbations in the network parameters.**

I. Introduction

Trustworthy operation of the nation's critical infrastructure like the electricity grid require effective cyber security and power system protections simultaneously. Ideas from cyber security research have been extensively deployed to keep adversaries out of the critical plants and control systems. However, cyber security solutions alone are inadequate for safe-guarding cyber-physical systems where software is used to monitor and control networks of complex physical processes.

Recent security incidents presage this new class of vulnerabilities. One well-known example is the Stuxnet worm, which targeted Siemens industrial software used to control nuclear fuel processing plants. The worm exploited several complicated cyber attack vectors, including four Windows zero-day vulnerabilities and logic controller exploits. It ultimately sabotaged and destroyed an Iranian facility by introducing malicious control inputs to actuators controlling uranium centrifuges. Understanding the scale and sophistication of this attack has led to mandatory governmental regulations embodied in the North American Electric Reliability Corporation's Critical Infrastructure Protection (NERC-CIP) regulations [29] that is now widely adopted in the industry.

The most recent CIP (v5 - enforceable on April 1, 2016) has significantly increased strict cyber security requirements in order to prevent adversarial power grid incidents by terrorists and targeted nation-state intruders. One of the major CIP standards for electricity grid protection is that the grid's real-time operation should always comply with "N-1" contingency resilience requirements. That is, given a power system with N components (for example transmission lines), the system must be able to tolerate any single component failure such as an overflown line outage possibly caused by malicious control inputs. Non-compliant utilities are required to increase their system redundancy to improve tolerance, otherwise they risk financial penalties imposed by the government. To implement the requirements, power utilities across the nation have been deploying various automation solutions such as protection relays to detect incidents, e.g., line current overflows, and clear the faults through opening of circuit breakers. The utilities have developed regular procedures such as contingency analyses to periodically validate their system resilience against failures. The existing contingency analysis algorithms that are used in practice nowadays leverage power system to perform *steady-state* analysis of potential incidents and their potential consequences. Those solution completely ignore non-steady-state or transient dynamics of the system. Thus far, these protections are largely believed to provide safety against the accidental component failures and malicious attacks such as Stuxnet that target the steady-state system dynamics.

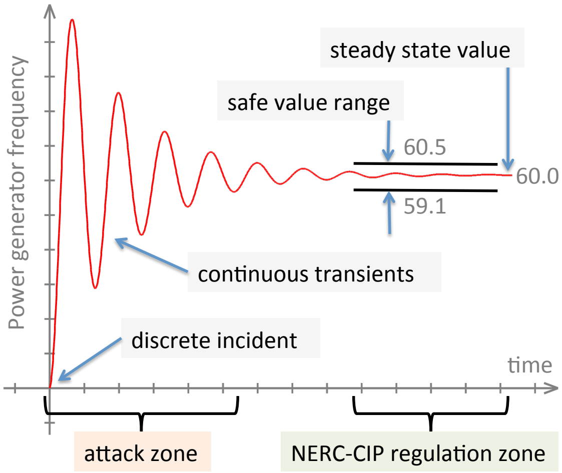


Fig. 1: Transient Dynamic Attacks vs. Enforced Regulations

We show that NERC-CIP cyber security requirements can be vulnerable against a new class of attacks, and hence inadequate for protecting the national electricity grid from dynamic transient exploits. Not only do we show the existence of transient attacks in systems that are compliant with NERC-CIP regulations, we present an algorithmic approach for effectively synthesizing such attacks from reasonably available information about power networks. These synthesized attacks against CIP compliant power networks can cause global system-level collapse or instability. As attackers are becoming more concerted and sophisticated, transient attacks should be taken into consideration for the design of future protection mechanisms and our analysis suggests conceptual methods for security evaluation of power networks.

The synthesis procedure leverages *hybrid models* of the electricity grid that incorporate not only the continuous, nonlinear dynamics of the electrical quantities like currents, volt-

[1] This is a Regular research paper.

ages, power, and phase angles, but also discrete dynamics of the changes in the network topology brought about by opening and closing of relays and changes in power consumption (loads). The synthesized attacks consist of a sequence of malicious control actions that ensure that all possible behaviors of the system remain undetected by the conventional steady-state monitors, but ultimately destabilize the system.

Our goal is to automatically synthesize such destabilizing attack over a specified bounded time horizon such that the deployed CIP compliant protection mechanisms are not triggered. Owing to the capabilities of the adversary, it turns out that the space of possible attacks can be naturally parameterized. The adversary can switch a few compromised relays (opening/closing power lines) or she can inject power at a few of the compromised buses. These actions can be stitched together to construct more complex attack sequences. Nevertheless, there are only a finite (and in fact small) number of network topologies that the adversary can realize. Although the set of possible power injection signals are infinite, they can also be parameterized via a family of curves or proportionalintegralderivative (PID) control signals.

With a parameterized attack space, the proposed algorithm leverages state-of-the-art formal verification algorithms to effectively search this space for successful attacks. The key to this approach is to be able to quickly eliminate parts of the attack space that *cannot* produce successful attacks—either because they are detected or because they are unsuccessful in making the system unstable. The algorithm eliminates sets of unsuccessful attacks by first simulating one potential attack with specific parameter values and checking many other similar potential attacks, by computing on-the-fly the sensitivity of the system's behavior to changes in the attack parameters. If all attacks with a single topology are eliminated, then the algorithm proceeds to check attacks that string together several topology switches with different power injections in each.

Contributions. The contributions of this paper are as follows: *i)* we design and demonstrate the first transient dynamics attack against critical electricity grid infrastructures, and show that the current governmental regulatory protections are rendered insufficient in practice; *ii)* we introduce an automatic attack synthesis framework that implements reachability-based synthesis to generate transient attacks for a given network while meeting realistic constraints on the attack space; and *iii)* we implemented a proof-of-concept prototype to demonstrate the feasibility of transient attacks synthesis against NERC CIP-compliant topologies.

The remainder of this paper is organized as follows: Section II describes and justifies the threat model used in the paper and gives an overview of the proposed approach. Section III describes the hybrid models of power network, protections and attacks. Section IV presents the attack synthesis problem formulation. Section V describes our results and demonstration of transient attacks. Section VI shows a case study transient attack against a power system that is NERC CIP N-1 compliant. Section VII reviews the past related work. Section VIII concludes the paper.

II. Overview

In this section, we justify the attack model and the architecture used throughout this paper based on the current state of the electricity grid, its operations, and its protection mechanisms.

Threat model. Like most of the past work on power grid attacks [22], we assume that the attackers know the configuration of target power network. We also assume that the attackers have access to necessary controllable actuation points to manipulate few system parameters such as a power generator governor valve (input mechanical power to the generator) and circuit breakers.

A. Electricity Grid Security

Electricity power grid consists of electric power generators, loads such as factories, where the power gets consumed, and the transmission lines to transfer the generated power to the consumption sites (loads). As an interconnected network, the power grid connects a variety of electric generators together with a host of users across a large geographical area. Generally, the power system transient dynamics originate from its power generators (synchronous machines). The generation AC power flows across the grid based on the Kirchhoff laws—power system equations. Any change to the system parameters or its network topology will change the equations and the power flow values will update accordingly. The changes to the grid are performed through installed actuators, and are normally used by the power operators to control and ensure the safe operation of the grid. Examples of typical actuators on the power system include *(i)* circuit breakers that are physical devices controlled by computer-enabled relays, and could connect/disconnect a power system asset such as a generator to/from the rest of the grid interconnect; *(ii)* generator governor steam valve controller that defines how much mechanical torque should be fed to the generator and hence power to the rest of the grid. Hence in case of load (power consumption) increase, the valve should open further to compensate for the consumed power, and maintain the load-generation balance; *(iii)* the DC current value controller that defines how much current should flow through the field windings within a generator. Higher field currents increase the generator's output voltage, and hence this actuator could be used for power grid's global voltage control and prevent so called a wide-area voltage collapse. In a real-world power system there are many actuators deployed, and they could be maliciously accessed remotely through cyber attacks, and leveraged to change the power system parameters to cause a large-scale blackout.

For reliability purposes, redundant paths and lines are provided so that power can be routed from any power plant to any customers, through a variety of routes that is resilient against failures such as a line outage. A SCADA network is usually used to monitor and control the power system and devices in a geographical area. The SCADA network is connected to the physical power system components through distributed sensors and actuators. The daily operation of the power grid follows typical closed-loop monitoring and control paradigm. Sensors send noisy measurements to the SCADA network, where the state estimation servers process the received measurement data points. The state estimation's objective is to filter out noise and calculate precise estimates of the various power system variables, i.e., power bus voltages. The power operators look at these estimates and decide on proper control actions that are sent to the actuators. For instance, if the calculated power system estimates indicate that there is more power consumption than generation in the grid, the operator increases the generation set-points on one (or more) of the generators to maintain the load-generation balance. Traditional power grids often have protection measures deployed to maintain reliable system operations against accidental and natural events and failures such as a broken line.

Due to the close and constant interaction of the physical power system with the SCADA network, the electricity grid is referred to as a *cyber-physical* infrastructure. The emerging advanced SCADA equipment improves cyber interconnections among various points (e.g., actuators) of the system from remote sites. Although this simplifies the grid's monitoring and control significantly, improving SCADA technologies also vastly inflates the grid's attack surface. The adversaries launch the attack through cyber-side penetration to gain sufficient control on the cyber assets such as actuators like power system relays that are often used to (dis)connect a transmission line. The final and more important step of the attack is to leverage the compromised actuators to cause a physical damage to the power system. For instance, disconnecting a transmission line to a large neighborhood may result in a temporal power outage in the area. There have been several successful security attacks against the electricity grid within the United States [1]. In fact, the number of reported major security attacks against the critical infrastructures has increased from 9 incidents in 2009 to 257 incidents in 2013, 28X within 4 years.

Due to the rising number of attacks, both the industry and government have taken initiatives to protect the electricity grid on both the cyber and the physical sides. Initial efforts involved incorporation of traditional IT cyber security mechanisms into SCADA networks. The direct adoption did not always result in suitable outcomes such as unacceptable packet transfer latencies for control inputs, which often have strict end-to-end time requirements. Later endeavors included more domain-specific mechanism deployments such as Digital Bond's Snort signatures for power control network DNP3 protocol. Following the cyber-side security protection improvements, however, there have been little progress on power-side solutions against non-accidental incidents and malicious adversaries that could lead to mis-operation or instability.

NERC-CIP contingency requirements. Consequently, NERC developed 412-page CIP v5 requirements, and Federal energy regulation commission (FERC) approved the requirements on November 22, 2013. The NERC website says "[the requirements] represent significant progress in mitigating cyber risks to the bulk power system". One of the key requirements is the $N-1$ *contingency* compliance. NERC defines a power system contingency as an unexpected failure or outage of a system component, such as a generator, transmission line, circuit breaker, switch or other electrical element [29]. NERC CIP N-1 contingency requirement mandates that grid must operate safely given any single contingency. To enforce the requirements, power utilities run contingency analysis algorithms on the grid's current topology and state every few minutes to ensure that it can tolerate any single contingency. All of the deployed real-world algorithms make use of steady-state system models such as (rarely) non-linear AC power flow models or (almost always) linearized approximate DC models for speed up. None of these models consider the transient dynamics of the system. We answer the following question in this paper: is it possible that an electricity grid is reported NERC CIP-compliant after state-of-the-art contingency analyses, while a (possibly sophisticated) attack is still destabilizes it?

B. *Overview of Synthesized Attacks*

We show that the NERC-CIP compliant systems are vulnerable against a new class of attacks that leverage the transient dynamics of the power system. These attacks can be launched with the threat models discussed above. The attack's core idea is to manipulate the system in such a way that its fine-grained (fast) transient dynamics exceeds the safety zone, while the system's steady-state converged values resides within NERC CIP requirement limits. Our algorithm achieves its objectives through a controlled violation of the *synchrony condition* [4] that refers to the condition when both the AC current frequency and phase for all generators within the electricity grid are aligned. Malicious loss of synchrony in the grid can lead to blackouts. The frequency of a generator is directly related to the speed (angular velocity) of its internal rotor that converts its input mechanical power (e.g., by steam turbines) to electrical power. The generated electrical power is then injected to the grid. The input mechanical power magnitude can be adjusted by the generator's governor valve controller. Consequently, a malicious governor valve controller could manipulate the rotor speed, and hence the system frequency that could possibly perturb the grid's synchrony condition.

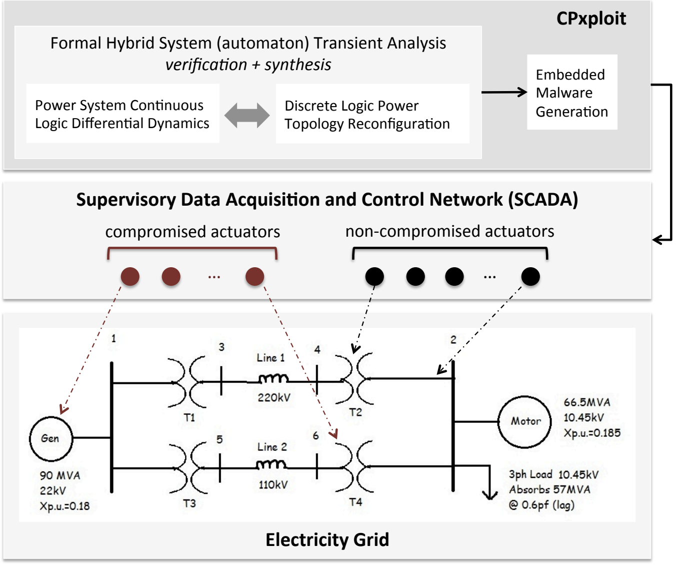

CPxploit
Formal Hybrid System (automaton) Transient Analysis
verification + synthesis
Power System Continuous Logic Differential Dynamics
Discrete Logic Power Topology Reconfiguration
Embedded Malware Generation
executable distribution
Supervisory Data Acquisition and Control Network (SCADA)
compromised actuators
non-compromised actuators
1
2
3
4
5
6
Line 1
220kV
Line 2
110kV
T1
T2
T3
T4
Gen
90 MVA
22kV
Xp.u.=0.18
Motor
66.5MVA
10.45kV
Xp.u.=0.185
3ph Load 10.45kV
Absorbs 57MVA
@ 0.6pf (lag)
Electricity Grid

Fig. 2: High-Level architecture for an attack.

Figure 2 shows the high-level architecture of an attack and how its components are logically interconnected. CIP requirements are based on the widely-used simplifying assumption that the system transients die (fade) quickly and it is focused on detecting anomalies in the steady-state values of the system. Normal transients are caused by abrupt discrete incidents, such as a transmission line outage, and do indeed meet this assumption. As we show, maliciously triggered transients, however, can be amplified significantly if one is followed by another in a quick succession. The synthesis algorithm explores the space of such transient attacks and (if possible) finds one that cannot be detected by CIP requirements that only consider the steady state values.

The algorithm searches for attacks over all possible sequence of network topology changes that the adversary can bring about. For each possible sequence, the algorithm explore the space of possible attacks that can drive the system to an *Unsafe* state that violates the synchrony condition, and avoids the states *Detected* that can be detected by the conventional (CIP) monitors. This exploration is performed by over-approximating the systems behavior (reachable states) under a set of attack injections. This computation of reachable states pushes the state-of-the-art in formal verification for hybrid models (see Section VII for an overview). A new class of verification algorithms [8], [11] compute reachable sets of the system by simulating individual behaviors and computing the local sensitivity of that execution to small perturbations to the states and inputs to the system. In this paper, we extend this approach to analyze the sensitivity with respect to the attack parameters. Since the algorithm is based on over-approximations, it is *sound* in the sense that it can infer that certain range of parameter values and certain toplogies *cannot* give rise to a successful attack. For example, for several networks, we show that CIP regulations effectively protect against attacks on the system with no discrete topological modifications. This enables the algorithm to move the search quickly to a different part of the attack space or a different (possibly longer) sequence of topology switches.

Finally, we comment on the advantages for synthesizing attacks. Generally, one could use simulation-based strategy to search for a an attack. However, simulation-based approaches alone are incomplete; they cannot guarantee absence of an attack even for small (but compact) sets of system parameters. Our formal approach using sensitivity computation enables us to eliminate (possibly large) sets of potential attacks as infeasible and therefore helps focus the search on interesting parts of the attack parameter space. Our choice was also influenced by the finding that CIP compliance and its protection measures significantly reduces the attack search space. It makes it significantly harder to realize an attack such that it can achieve its adversarial objectives without triggering CIP's mandatory protection measures. Therefore, the proposed exploration for a feasible attack recipe includes the so called corner cases that are not often hit by simulation methods quickly in practice.

III. Transient Attacks on Power Networks

We describe typical dynamical models for monitoring and control of power networks, and then we proceed to develop the core technique for synthesizing transient attacks.

A. Power Network Dynamics

A power network consists of several buses. Each bus serves as an electrical interconnection point for generators, loads, and transmission lines to other buses. A network with three buses is illustrated as Figure 5a. The relationship among the currents and voltages in an N-bus network is given by the Kirchoff's law

$$\vec{I} = Y \cdot \vec{V},$$

where $\vec{I} = [I_1, \ldots, I_N]$ and $\vec{V} = [I_1, \ldots, I_N]$ are complex vectors corresponding to the currents the voltages in the buses, and Y is the $N \times N$ matrix of complex numbers called the *Nodal admittance matrix* and it captures the electrical parameters of the network. The *admittance* (reciprocal of the *impedance*) between the i^{th} and the k^{th} bus is the complex number $Y_{ik} = G_{ik} + jB_{ik}$. Its real part G_{ik} is called the *conductance* and its imaginary part B_{ik} is called the *susceptance*. The second relationship is given by the power flow equation: $S_i = V_i \cdot I_i^*$ for each bus i. By rewriting $\vec{I}$ as $Y.\vec{V}$ and comparing the real power P_i and the *reactive* or imaginary part of power Q_i at each bus, we get

$$P_i = \sum_{k=1}^{N} |V_i||V_k|(G_{ik}\cos(\delta_i - \delta_k) + B_{ik}\sin(\delta_i - \delta_k)), \quad (1)$$

$$Q_i = \sum_{k=1}^{N} |V_i||V_k|(G_{ik}\sin(\delta_i - \delta_k) - B_{ik}\cos(\delta_i - \delta_k)), \quad (2)$$

where δ_i is the phase angle of the complex voltage V_i. For an N-bus network this gives $2N$ equations with $4N$ unknowns, namely, $|V_i|$, P_i, Q_i, and δ_i for each bus i. If we fix $2N$ of these unknowns (e.g., based on real-time sensor measurements), we can solve these equations and obtain the *steady state* values of power, reactive power, the voltage and the phase angle.

It is worth mentioning here that there are three different kinds of buses in a power system: *i)* a *generator bus* connects power generation equipment which fixes the real power P and the voltage $|V|$; *ii)* a *load bus* connects power consumers which fixes the real (P) and reactive power (Q); and *iii)* a *slack bus* connects to a *slack generator* that balances the active and reactive power in the system and it fixes the voltage $|V|$ and the phase angle θ at the bus. In a real power system with these three types of buses, each bus fixes two of the four unknowns, and therefore, we can solve Equation 1 and Equation 2 to obtain the steady state values of all the quantities.

In Section IV, we will consider power networks subject to changes in the input/output power, voltage, as well as in the network topology. When the topology of the network changes the admittance matrix Y changes, which in turn changes the steady state values of voltage, power, and phase of one or more buses as determined by Equations (1)-(2). Similarly, when the power at a load bus changes suddenly, then the steady state values at all the other buses may be affected. Between the time when the change occurs and the time when new steady state values are reached (if at all), the network is said to be in *transient state*. The thesis of this paper is that currently implemented NERC-CIP monitoring approaches leave power systems fundamentally vulnerable to attacks that exploit the transient dynamics of the system.

Finally, we describe the differential equations that govern the transient behavior of each generator. Each generator k has two state variables, namely the synchronous phase angle θ_k and the angular velocity ω_k. Given the bus quantities δ_k and V_k from the above, the dynamics of generator k is given by

$$\begin{aligned} \dot{\theta}_k &= \omega_k & (3) \\ \dot{\omega}_k &= C_1(P_M - C_2|V_k|\sin(\theta_k - \delta_k) - C_3\omega), & (4) \end{aligned}$$

where C_1, C_2, and C_3 are constant parameters of the generator and its circuitry; P_M is the input mechanical power to the generator (e.g., by a steam turbine). We denote the term

$$P_E := C_2|V_k|\sin(\theta_k - \delta_k)$$

as the output electrical power generated by the generator. Given a power system topology and steady state values of the bus voltage and phase angle, the output power is a function of the machine phase angle θ_k. The output current is derived by dividing the output power by the voltage $I_k = C_2\sin(\theta_k - \delta_k)$. Equation 3 merely says that the rate of change of the phase angle θ_k is the angular speed ω_k. Equation 4 says that the rate of change of angular speed depends (according to this nonlinear function) on the mechanical power input P_M, the phase angle θ_k, the angular velocity ω, and the steady state voltage $|V_k|$ and rotor angle δ_k of the bus it connects to.

To keep the power system stable, the phase angle has to satisfy the *synchrony condition*, i.e., keep $\theta_k \leq a_{crit}$, where $a_{crit} = \pi/2 + \delta_k$ is the *critical angle* [20]. If the generator is operating below the critical angle, it is dynamically stable, meaning it can tolerate the environment disturbance and maintain the power output. However, if the phase angle is above the critical angle, the generator's behavior goes out of synch with even little disturbances. We will further justify this statement in Section VI with experiments on a high-fidelity power system simulator. Throughout this paper, we study attackers that tempt to violate the synchrony condition and drive the phase angle larger than the critical angle without violating the steady-state NERC-CIP conditions.

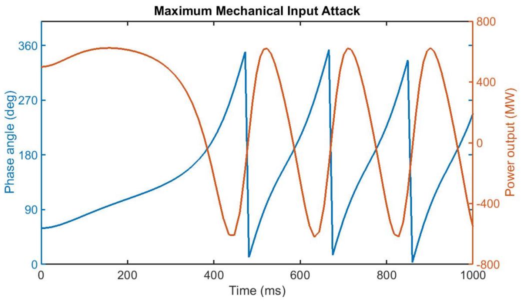


Fig. 3: Attack with maximum input mechanical power. The lines capture the trajectories of phase angle and output power with constant input $P_M = 7$ p.u.

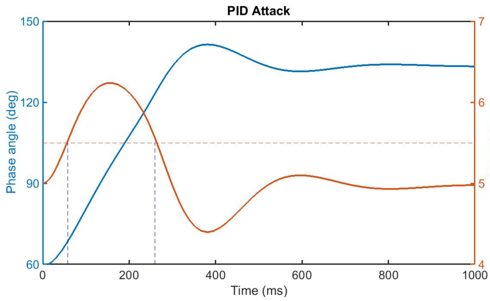


Fig. 4: Attack with a PID control. The solid lines capture the trajectories of phase angle and current. The critical current of the relay is 5.5 as illustrated by the horizontal dashed line.

B. Attack Protection: Relays and Monitors

The most widely deployed protections in power networks to enforce NERC-CIP requirements and protect against attacks are *relays* and *steady state monitors*. A relay has two parts: a sensor for detecting over-current conditions, and an actuator for electrically disconnecting power lines. Each sensor detects over-current condition on the line between a generator and a specific bus. A relay on the k-th bus registers an *over-current event* if $I_k = C_2 \sin(\theta_k - \delta_k)$ exceeds a critical current value I_{crit} (transmission line capacity) for at least a certain threshold duration c_{oc} of time. In contrast, steady state monitors check whether the steady state power output of a generator P_E deviates significantly from a desired value P_{ref} for some duration of time. These monitors detect a deviation if $P_E(t)$ is not within $\pm P_{tol}$ of P_{ref} over an interval of time of duration c_{pd} (here P_{tol} is a constant tolerance parameter).

Indeed, these protections are adequate against a broad class of attacks. One simplest attack is to set the input mechanical power P_M to a constant high value (e.g., via opening the steam turbine valves) such that the phase angle of the machine will increase and enter the asynchronous region exceeding the critical threshold value. The corresponding trajectories of the phase angle and the output power are illustrated in Figure 3. Since the phase angle is not stabilized, the output power also oscillates wildly; therefore, this attack can be caught by an appropriate steady state monitor. A more sophisticated attack would use a time-varying input power signal P_M. The attacker first computes a reference angle $\theta_r = \pi - \sin^{-1}\frac{P_{ref}}{C_2|V_k|} + \delta_k$, such that it violates the synchrony condition ($\theta_r > a_{crit}$) and it produces the steady state power output equals P_{ref}. Then, the attacker picks the PID constants $k_p, k_i, k_d \geq 0$ and compute the input mechanical power at each time $t \geq 0$ as

$$P_M(t) = k_p e(t) + k_i \int_0^t e(s)ds - k_d \omega(t), \tag{5}$$

where $e(t) = \theta_r - \theta_k(t)$ is the difference of the current phase angle and the reference angle. This attack can drive the phase angle to the asynchronous value θ_r while ensuring that the output power P_E to remains within the desired range. Hence it remains undetected by the steady state monitors. An example of how such an insidious attack may proceed is shown in Figure 4, where after the transient, the machine phase angle converges to $\theta_r = 133°$ and the output current converges to 5 p.u. (per unit). If the generator is equipped with a relay with critical current $I_{crit} = 5.5$ p.u., the relay will be triggered between the two vertical dashed lines in the plot for around 200 ms. If the critical over-current duration $c_{oc} \leq 200$ms, then the attack will be detected.

We observed that the relays and steady state monitors are able to detect some simple attacks such as maximum mechanical input attacks or PID attacks. In the following sections, we discuss how to synthesize a valid transient attack automatically that evades the deployed monitors by carefully composing the simple attacks.

IV. The Attack Synthesis Problem

We present a formal model for power networks as hybrid automata. We then use the model to synthesize and generate transient attacks automatically.

A. Formal Hybrid Model

As we saw in the previous section, the transient behavior of a power network can be modeled by several sets of differential equations—one for each configuration of the network topology and the bus parameters. These differential equations define the *trajectories* of the physical state of the system over time intervals. Together with these trajectories, the complete system description requires a set of *transitions* that define the abrupt changes in the network configuration, e.g., a relay opening. After every such transition, the power system transients evolve according to its dynamic model of differential equations. It has now become standard to model such systems combining differential equations and transitions as a *hybrid automaton* [19], [25]. A hybrid automaton is a state machine with both discrete and continuous state variables. The discrete state variable typically capture software (cyber) states and the system configurations and their changes are specified by discrete transitions rules similar to if-then statements, e.g., a relay's open/close conditions. While the continuous variables capture physical variables (for example, voltage and phase angles) and their evolution is specified by differential equations. For the sake of precision, we give a formal definition of a hybrid automaton.

Definition 1. *A* hybrid automaton (HA) $\mathbf{A}$ *is a 6-tuple* $\langle Var, Q, U, \Theta, D, T \rangle$*, where*

1) *Var is a variable set consisting of i) X a set of real-valued continuous state variables; ii) mode a single discrete variable that takes value in the finite set denoted by L.*
2) $Q = \mathbb{R}^{|X|} \times L$ *is the state space of the system; it corresponds to all possible valuations of the continuous variables in X and the discrete variable mode.*
3) *U is a set of real-valued continuous input variables.*
4) $\Theta \subseteq Q$ *is the set of initial states,*
5) $D \subseteq Q \times Q$ *is the set of possible discrete transitions.*

6) *T is the trajectory set. Each trajectory $\tau \in T$ is a function $\tau : [0,t_1] \rightarrow Q$ mapping the time interval $[0,t_1]$ to the states.*

For a transition $(q,q') \in D$, it is standard to write $q \rightarrow q'$; q is called the pre-state and q' is called the post-state of the transition. For a trajectory $\tau \in T$ and a time point t in the interval $[0,t_1]$, $\tau(t)$ is the state of the system at time t. For a particular variable $x \in X$, the valuation of that variable at $\tau(t)$ is denoted by $\tau(t).x$. A hybrid automaton with no inputs, $U = \emptyset$, is said to be *closed*. Otherwise it is *open*.

Next we instantiate the above definition to model a specific power network. There are software tools [3], [24], [11] that automatically convert engineering models constructed in different commercial tools like Simulink, LabView, and Mathematica to the hybrid automata. Consider the 3-bus power network shown in Figure 5. For the sake of illustration, we consider two different topologies of the network: topology 1 with the line between buses 2 and 3 closed and topology 2 with the line open. These two topologies and their corresponding steady state values for voltages, phase angles, etc., give rise to two configurations. With each topology $i \in \{1,2\}$, the steady state voltage $|V_{ik}|$ and phase angle δ_{ik} at bus the k^{th} bus are determined by the admittance matrix of the i^{th} topology. With those quantities fixed, the dynamics for the generator at bus k, is given by Equations (3)-(4):

$$\begin{aligned}\dot{\theta}_k &= \omega_k\\ \dot{\omega}_k &= C_1(P_M - C_2|V_{ik}|\sin(\theta_k - \delta_{ik}) - C_3\omega_k).\end{aligned}$$

For bus 2, P_M is the mechanical power input at the bus and for the other buses this quantity is set to zero. These two sets of differential equations for the two topologies are represented by two functions $[\dot{\theta},\dot{\omega}] = f_i(\theta,\omega,P_M)$, where $i \in \{1,2\}$.

A schematic representation of the corresponding hybrid automaton is shown in Figure 6. Its components are specified as follows: (1) The set of continuous variables X consists of θ_k and ω_k for each of the N generators and a timer variable *clk* for the relay. The discrete *mode* variable can take values from the set $\{\text{on1},\text{off1},\text{on2},\text{off2}\}$; *mode* $=$ on1 (respectively *mode* $=$ off1) and corresponds to the configuration where the topology 1 is active and the relay timer (*clk*) is on (and off). (2) The set of states $Q = \mathbb{R}^{2N} \times \{\text{on1},\text{off1},\text{on2},\text{off2}\}$. (3) The single continuous input variable P_M corresponds to the mechanical power input (4) The mode is switched from offi to oni, with topology $i \in \{1,2\}$, if the current $C_2\sin(\theta - \delta_i)$ goes above a threshold I_{crit} and switched back otherwise. The switch between the two topology is assumed to be controlled by the attacker (5) The set of trajectories T is defined by the differential equations. Specifically, a trajectory $\tau : [0,T] \rightarrow Q$ is a valid trajectory if and only if it is a trajectory for one of the four modes. For example, if it is a trajectory of mode off1 then at each time $t \in [0,T]$ along the trajectory, (i) $\tau(t).mode = \text{off1}$ remains constant, (ii) $\tau(t).clk = \tau(0).clk$, i.e., the relay timer remains constant, and (iii) $\tau(t).\theta$ and $\tau(t).\omega$ are solutions of the differential equation $[\dot{\theta},\dot{\omega}] = f_1(\theta,\omega)$ for some input signal for the mechanical power P_M. The trajectories for the other modes are defined analogously. For the on1 and on2, at each time $t \in [0,T]$ along any trajectory $\tau(t).clk = \tau(0).clk + t$, i.e., the relay timer measures time elapsed.

B. Attack Synthesis Problem

To present the attack synthesis problem precisely, we define the semantics of hybrid models. The semantics of a hybrid automaton model is given in terms of its runs or execution. An *execution* of a hybrid automaton $\mathbf{A}$ is a finite sequence of trajectories $\alpha = \tau_0, \tau_1, \ldots, \tau_m$, such that each τ_i is a valid trajectory of the automaton. It is a solution of the differential equations corresponding to one of the modes—which in power networks correspond to a particular configuration. For any pair of consecutive trajectories $\tau_i : [0,t_1] \rightarrow Q$ and $\tau_{i+1} : [0,t_2] \rightarrow Q$ in the sequence α, there must be a valid transition from the last state of τ_i to the first state of τ_{i+1}, i.e., $\tau_i(t_1) \rightarrow \tau_{i+1}(0)$. Finally, a sequence of trajectories meeting the above two criterion is deemed an execution only if the first state of $\tau_0(0)$ is one of the initial states of the automaton.

A state $q \in Q$ of $\mathbf{A}$ is said to be *reachable* if there is some execution α that arrives at it. The set of all the reachable states of the automaton $\mathbf{A}$, written as $Reach_{\mathbf{A}}$, plays an important role in analysis and verification of automata. For instance, for some specified set of bad states $U \subseteq Q$, if $Reach_{\mathbf{A}} \cap U = \emptyset$, then it follows that the system is safe with respect to U. If we can find a set of input signals for which $Reach_{\mathbf{A}} \subseteq U$ then it establishes that the system always end up in a bad state. An invariant for $\mathbf{A}$ is any over-approximation of $Reach_{\mathbf{A}}$. Thus, a fundamental problem in analysis of automaton models is to compute or approximate $Reach_{\mathbf{A}}$. In Section IV-C, we give a brief overview of the related work in this area.

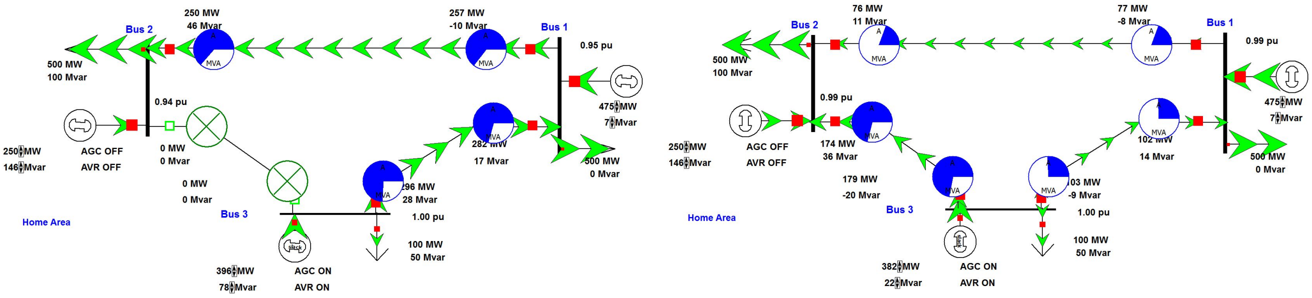
A hybrid automaton is non-deterministic in general. It may have multiple initial states, more than one transition may occur from a given state, and even more than one trajectory may emanate with different inputs (for example, the input power P_M) from a given state (and mode). On the one hand, these different sources of non-deterministic choice make the models expressive. They enable us to capture uncertainties in the model parameters, measurements, and the environment. On the other hand, non-determinism makes it difficult to design algorithms for precise approximation of the set of reachable states $Reach_{\mathbf{A}}$.

We state the transient attack synthesis problem using reachability. First, suppose the attacker's input signal is generated as a PID control input as given in Equation (5) parameterized by the constants k_p, k_i, and k_d ranging over a parameter space $\mathcal{P}$. This assumption makes sense because (a) PID is the predominant method for generating control signals, and (b) a broad class of continuous functions can be approximated as PID signals. To find a transient attack on the hybrid automaton $\mathbf{A}$, we would like to find PID parameters $\langle k_p, k_i, k_d \rangle \in \mathcal{P}$ such that there exits a time $t \leq T$ when $Reach_{\mathbf{A}}(t) \subseteq Unsafe$ while for all $t \leq T$, $Reach_{\mathbf{A}}(t) \cap Detected = \emptyset$. Here *Unsafe* is the unsafe ansynchronous state ($\theta_k > \pi/2 + \delta_{ik}$), *Detected* is the set of states that can be detected by either the relay or the steady state monitor, and $Reach_{\mathbf{A}}(t)$ is the set of states reached by the system at time t before the time horizon of analysis T.

If the search for the parameters is successful, then the system is guaranteed to enter the bad state *Unsafe* within the time horizon T while evading detection. On the other hand, if the search terminates without finding appropriate parameters, then we can conclude that there is no such attack at least within the parameter space $\mathcal{P}$.

C. Algorithm for Attack Synthesis

In this section, we sketch the main idea underlying the automatic attack synthesis algorithm. The key is to parameterize the space of attacks and then search for successful attacks by eliminating parts of the parameter space that cannot yield attacks. This elimination procedure is at the heart of our synthesis algorithm and it combines numerical simulations with formal static analysis of the models. In what follows, we first discuss synthesis of attacks consisting of a single attack mode



(a) Topology 1. The line between bus 2 and 3 is open. More power on other two lines. The complex voltage of bus 2 is $0.94\angle-17°$ p.u.

(b) Topology 2. The line between bus 2 and 3 is closed. Less power on other two lines. The complex voltage at bus 2 is $0.99\angle-5°$ p.u.

Fig. 5: Two topologies of a 3-Bus Power system

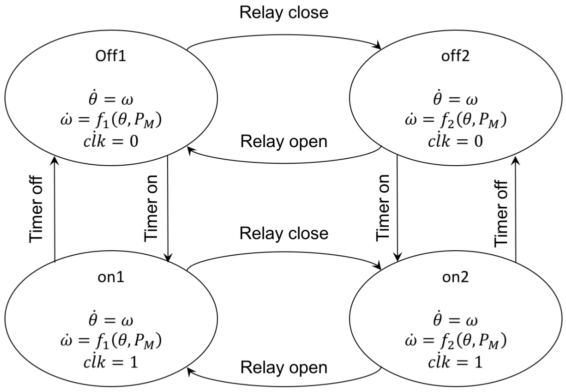


Fig. 6: Schematic diagram of the HA modeling power network with two topologies. The four modes and their differential equations are shown in the circles. The arrows show the discrete transitions.

defined by a single configuration of the network and a single set of attack parameters. In Section V we discuss how more complex attacks consisting of a sequence of configurations and attack parameters can be synthesized using the same principle.

In the preceding example, for instance, we chose to define the space of parameters $\mathcal{P}$ as the values of the PID gains k_p, k_i and k_d and also the timing of the topology switches. For a given hybrid model $\mathbf{A}$, let us fix the mode or configuration of the network to be i. Now if we also fix a specific value of these parameters $p \in \mathcal{P}$, arbitrarily, then that defines a unique trajectory $\tau_p \in T$ that satisfies the differential equations of the hybrid automaton.

We can simulate this system to infer if this arbitrary trajectory τ_p, $p \in \mathcal{P}$ yields a successful attack. If it does not, however, we learn very little as there are infinitely many other choices in $\mathcal{P}$. This is where we use sensitivity and simulation-based analysis technique from the formal methods literature that has been quite successful in verifying industrial scale control systems [8], [11]. For a time $t \geq 0$, compact set $P \subseteq \mathcal{P}$ of parameter values, let us denote by $Reach_{\mathbf{A}}(P,t)$ the set of states that are reachable at time t with parameter values in chosen from the set P.

The idea is to generalize from this one choice $p \in \mathcal{P}$ of parameter values, a much larger set of values $P \subseteq \mathcal{P}$ that is also guaranteed have no attack. We can then eliminate P from the search and move on to a different part of $\mathcal{P}$. We formalize this generalization using *discrepancy* used in [10], [16]:

Definition 2. *A function $\mathcal{P}^2 \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is the* discrepancy function *of* $\mathbf{A}$ *if*

1) *for any pair of parameter values $p, p' \in \mathcal{P}$:*
$$|\tau_p(t) - \tau_{p'}(t)| \leq \beta(|p - p'|, t), and \tag{6}$$
2) *for any $t \geq 0$, $\beta(|p - p'|, t)$ converges to 0 as $p \rightarrow p'$. Here $|\cdot|$ is the standard ∞-norm of Euclidean space*

Discrepancy function β upper bounds the sensitivity of the trajectories τ_p and $\tau_{p'}$ to the changes in the parameters as a function of time. Several algorithms for computing the discrepancy functions have been presented in literature [10], [8]. By computing (or simulating) one of the trajectories $\tau_p(t)$ and enlarging it by the factor $\beta(\varepsilon_1, t)$, we can compute all the trajectories—and the states they reach—resulting from changes in parameters. In summary then, discrepancy functions can be used to compute over-approximations of $Reach_{\mathbf{A}}(P,t)$ just from simulations of the system.

The algorithm for attack synthesis using this principle is shown in algorithm 1. The input to the algorithm is the automaton model $\mathbf{A}$, the discrepancy function(s) for the different modes, the unsafe set *Unsafe* (in power networks this is the asynchronous state), the detectable set *Detected*, the time horizon T_{max}, and the set of parameters $\mathcal{P}$. The algorithm maintains a data structure S which is a *cover* of the parameter space $\mathcal{P}$. Concretely, S is a collection of pairs $\{\langle p_i, \varepsilon_i \rangle\}_i$, such that the compact parameter space $\mathcal{P}$ is contained in $\cup_i B(p_i, \varepsilon_i)$, the union of the ε_i-balls[2] around the p_i's. More generally, the cover S could have balls of different sizes.

For each parameter value $\langle p_i, \varepsilon_i \rangle \in S$, the algorithm first simulates the system with parameter p_i to generate one execution α_{p_i} of duration T_{max} (line 4). It then uses the discrepancy functions to compute the set of states R by bloating α (line 5). From the property of discrepancy functions and the argument presented above we know that R is an over-approximation of the set of reachable state $Reach_{\mathbf{A}}(B(p_i, \varepsilon_i), T_{max})$.

Next in line 7 the algorithm checks if there exists a time t at which the reach set $R(t)$ from the parameter range $B(p_i, \varepsilon_i)$ is contained in the unsafe set *Unsafe* and for each $t \leq T_{max}$, $R(t) \cap Detected = \emptyset$. If this condition holds, then the parameter p_i yields a successful attack—one that reaches *Unsafe* while avoiding *Detected*. Having found an attack with p_i then the

[2] The set $B(x, \varepsilon)$ is the ball of radius ε centered at x, i.e., $\{y \mid |y - x| \leq \varepsilon\}$.

Algorithm 1: Transient attack synthesis.

```
1  input $\mathbf{A}, \beta, Unsafe, Detected, \mathbf{T_{max}}, \mathcal{P}$;
2  $S \leftarrow\ Partition(\mathcal{P})$;
3  while $S \neq \emptyset$, for $\langle p_i, \varepsilon_i \rangle \in S$ do
4      $\alpha \leftarrow Simulate(\mathbf{A}, \mathbf{p_i}, \mathbf{T_{max}})$;
5      $R \leftarrow\ Bloat(\alpha, p_i, \varepsilon_i, \beta)$;
6      if $\exists\ t \leq T_{max}\ R(t) \subseteq Unsafe$ and
         $\forall\ t \leq T_{max}\ R(t) \cap Detected = \emptyset$ then
7          return (Success, $p$)
8      else if $\forall\ t \leq T_{max}\ R(t) \cap U = \emptyset$ or
         $\exists\ t \leq T_{max}\ R(t) \subseteq Detected$ then
9          $S \leftarrow S \setminus \langle p_i, \varepsilon_i \rangle$;
10     else
11         $S \leftarrow S \cup\ Refine(p_i, \varepsilon_i)$;
12     end
13 end
14 return (Fail,$\perp$)
```

algorithm terminates. If there is no proof of successful attack from $B(p_i, \varepsilon_i)$ then the algorithm checks (line 9) if (a) for all $t \leq T_{max}$ $R(t)$ is disjoint from *Unsafe*, i.e., it does not make the system unsafe or (b) there is a time $t \leq T_{max}$ at which the system is detected, i.e., $R(t) \subseteq Detected$. In either case, we can infer that none of the parameter values in $B(p_i, \varepsilon_i)$ gives a successful attack, and therefore, it eliminates $\langle p_i, \varepsilon_i \rangle$ from the set S. Finally if none of the above conditions hold, i.e., the algorithm is not able to infer conclusively if the set $B(p_i, \varepsilon_i)$ gives a successful attack or not, then this set is partitioned more finely for future consideration using the *Refine* function. Once all the elements in the set S are removed (in line 9), the algorithm concludes that there is no attack for U possible from the set of parameters that is $\mathcal{P}$.

In the above we discussed the attack synthesis algorithm using discrepancy functions, and we did not delve into the details of how *Bloat* works with discrete transitions or switches between different topologies and configurations. Our implementation of attack synthesis for this paper does handle the transitions and it uses hybrid models [11].

D. Switched Transient Attack Parameter Space

A switched attack (attack with transitions) consists of a sequence of *attack modes* $A_0, A_1, \ldots, A_{n-1}$ and a sequence of time points $t_0, t_1, \ldots, t_n$. Attack A_i is active over the time interval $[t_i, t_{i+1})$. Each attack mode A_i records, as before, a network topology and the mechanical input signal P_M over the interval. Thus, synthesizing a switched attack of length n is the same as identifying the n topologies, the switching times, and the parameters defining the input signal (P_M) in each of the attack modes. In each attack mode i, the input signal P_M can either be some constant $(constPM_i)$ or follow a PID law as before:

$$P_M(t) = kp_i e(t) + ki_i \int_0^t e(s)ds - kd_i \omega(t), \tag{7}$$

where $e(t)$ is the difference between the reference phase angle and current phase angle. In summary, a program for executing the switched attack would look like the pseudocode in Figure 2 with all the parameters are synthesized from our algorithm.

Otherwise, the mechanical input P_M is set to a constant $?constPM_i$. The attack synthesis problem is reduced to assigning proper values to parameters t_i, top_i, $usePID_i$, kp_i, ki_i, kd_i, and $constPM_i$.

Algorithm 2: template of attack mode A_i

```
if $Time \in [t_i, t_{i+1}]$ then
    $Topology \leftarrow top_i$;
    if $?usePID_i = 1$ then
        $P_M \leftarrow PID(kp_i, ki_i, kd_i, State)$;
    else
        $P_M \leftarrow constPM_i$;
    end
end
```

To synthesize all the parameters in an attack sequence, we use Algorithm 1 as a subroutine. Notice that the there are only a finite (typically small) number of choices for the topology top_i depending on the number of relays and lines the adversary can have access to. Thus, we use a brute force search for all possible choices of top_i. Then for each sequence of choices for the topologies, we run Algorithm 1 to check if there exists a valuation of the other parameters that produce an undetectable attack.

V. Evaluations

In Section III-B, we discussed two attacks by which the mechanical input (P_M) to one of the generators in the network were controlled to make the system unstable and we showed how these attacks are detected by the conventional protection mechanisms. Now, equipped with the attack synthesis algorithm of Section IV-C, we present synthesized attacks that evade detection with relays and steady state monitors. In addition to the mechanical input, now we endow the attacker with limited capability in changing the topology of the network by opening or closing one of the lines in the network.

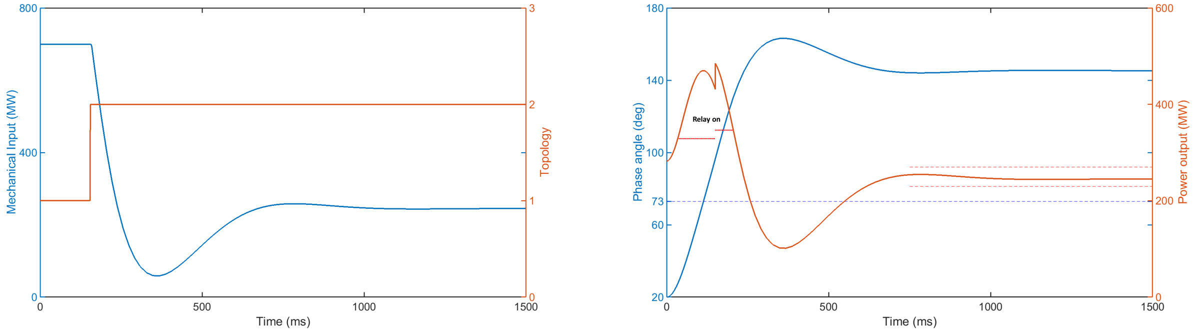
A. Application to 3-Bus Power Network

As a first illustration of a synthesized transient attack that is undetected by the relays and the steady state monitors, we revisit the 3 bus System of Figure 5. We consider the scenario where the attacker can control the mechanical input to the generator at bus 2 and also switch the topology between 1 and 2. The machine is equipped with a relay and a steady state monitor. The critical current of the relay is $I_{crit} = 3.5$ p.u. and the over-current duration is $c_{oc} = 175$ ms. The steady state monitor uses a reference power $P_{ref} = 250$ MW with tolerance $P_{tol} = 20$ MW. The mechanical input has to take values in $[0, 700]$ MW.

The attack can be make arbitrarily complicated by having more attack modes. Here, we study switched attacks with two attack modes. Our experiments show that the class of attack is rich enough for finding valid transient attacks to the power system.

One such attack is shown in Figure 7a. It first applies a constant input $P_M = 700$ MW upto 154 ms in topology 1, and then in the second attack mode, P_M is generated as a PID input with parameters $k_p = 6, k_i = 0.1, k_d = 0.2$ in topology 2. The resulting transient behaviors of the phase angle and the output power are illustrated in Figure 7b. The asynchronous states *Unsafe* are those with phase angle greater than the critical angle 72°. It is clear that the attack drives the system to *Unsafe*.

Since the output power is derived by multiplying the output current by the bus voltage $|V_i|$, the relay will be on if the output power is above $|V_i| \cdot I_{crit}$. The thresholds for both topologies are plotted as red dashed line segments in Figure 7b. We observe



(a) First apply a constant input power 700 MW and topology 1 for 154 ms. Then switched to topology 2 and a PID controller with $k_p = 6, k_i = 0.1, k_d = 0.2$.

(b) The phase angle converges to 145° and the power output converges to 250MW. The power output has a discrete jump at time 154ms because of the topology change.

Fig. 7: An attack to a 3 bus system. The critical angle of machine 2 is 72°.

that the relay is on for 167 ms. Thus the attack is not detected by the relay. Moreover, after around 600 ms, the output power stabilized within the range $[230, 270]$ MW, such that the attack is not detected by the steady state monitor.

B. Attack Generalization

We applied the same analysis to a couple of different power systems including the Western system coordinating council (WSCC) 9-bus power system[3].

The relay parameters are $I_{crit} = 2.25$ p.u. and $c_{oc} = 175$ ms. $P_{ref} = 163$ MW, $P_{tol} = 20$ MW. We successfully synthesize an attack as illustrated in Figure 8. The input mechanical power is plotted in Figure 8a: In the first 237 ms, the mechanical input is set to 400 MW, and afterwards the attacker use a PID control with $k_p = 7, k_i = 1.2, k_d = 0$. The resulting phase angle and power out are illustrated in Figure 8b, where the relay is on for 159 ms and the power output converges to 163 MW.

The experiments suggests that, the proposed algorithm can be applied to general power networks and produce sound attacks. Even though there are dramatic difference between the work topologies, power flow conditions, sensor parameters and input constraints of the 3 bus and 9 bus systems, the attacks follow the same pattern. That is, first attack mode applies a large mechanical power and the second mode is a PID control law. This pattern makes intuitive sense. A maximum mechanical input drives the phase angle to *Unsafe* with minimized duration of the relay on. Then a PID law guarantees the output power to stabilize without triggering the steady state monitor. This observation suggests that by applying our algorithm we could recognize attack patterns to the system.

C. Robustness of Attacks

So far our attack synthesis approach assumed perfect knowledge about the model of the system. In reality however, detailed models may not be available to the attacker and the network parameters like loads, line admittances, and generator inputs fluctuate over time. Since our synthesis approach not only computes a single attack (with a single set of parameters) but it computes a set of parameter values $B(p_i, \varepsilon_i)$ that produce a successful attack, we expect that these attacks enjoy some degree of robustness. That is, even if the network parameters change to some degree, the same attack will continue to produce similar qualitative results, i.e., evade detection and drive the system to an unsafe asynchronous state. In this section, we put this hypothesis to test by perturbing several of the network parameters and subjecting it to the same attack as the one synthesized in Section V-A.

As we discussed in Section III, the change in the load parameters and the admittance matrix results in different bus voltages and phase angles. These are obtained by solving the power flow equations (1)-(2). The bus voltages and phase angles directly affect the dynamics of the generator (4).

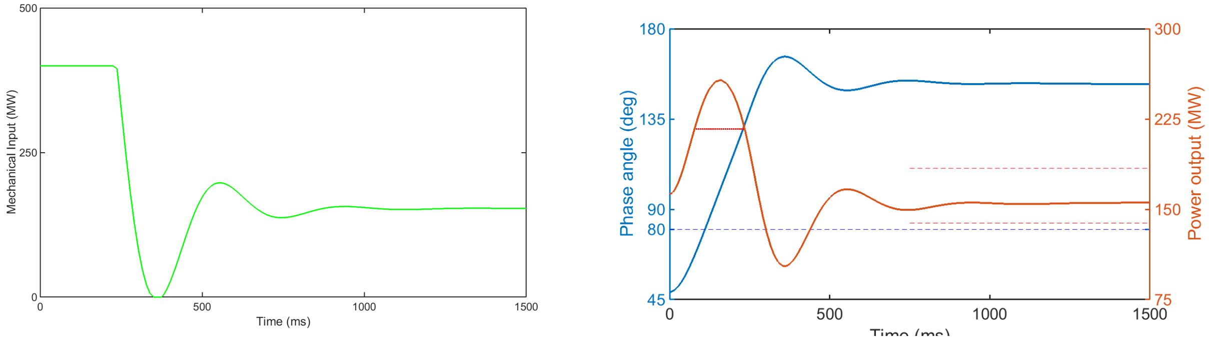
For example, consider the same system topology as illustrated in Figure 5b. If we perturb the power demand at bus 2 from 500 MW to 550 MW and the line admittances from 5.5-j18.3 p.u. to 5-15j p.u., then the voltage of bus 2 will change from $0.99\angle -5°$ p.u. to $0.98\angle -5.8°$ p.u., which alters the differential equation of machine 2.

In this section, we allow the voltage of bus 2 has a ± 0.2 p.u. perturbation, phase angle have a $\pm 1°$ perturbation. With this level of uncertainty in the parameters, we compute the reachable states of the system under the same attack. The computation uses a similar reachability algorithm as presented in Section IV-C and the result is that the reach set satisfies the same properties as a successful attack: it avoids *Detected* and eventually reaches the *Unsafe* set. The reach set of the phase angle and power output of generator 2 are plotted in Figure 9a. We observed that, the set of reachable phase angle is eventually contained by the *Unsafe* (72° or above). By examining the reachable states, we observe that the relay is on for at most 172 ms and the power output converges to 250 MW in all cases. That is, the reach set does not intersect with the detectable states *Detected*. A similar robustness property is proved for the attack to the 9 bus system as shown in Figure 9b. These experiment results suggest that the attacker can synthesize an attack without knowing the precise model of the system.

VI. Transient Attacks vs. NERC CIP-Compliance

Following the attacks synthesized automatically in the previous section, we implemented transient attacks against a realistic version of the 9-bus system with NERC-CIP mandated protection in the PowerWorld simulator [26]. PowerWorld is the de facto simulator for power system operation and control. Its models are much more detailed, higher dimensional, and complicated with many more parameters than the ones used in models of Section III-A. For example, these models allow the

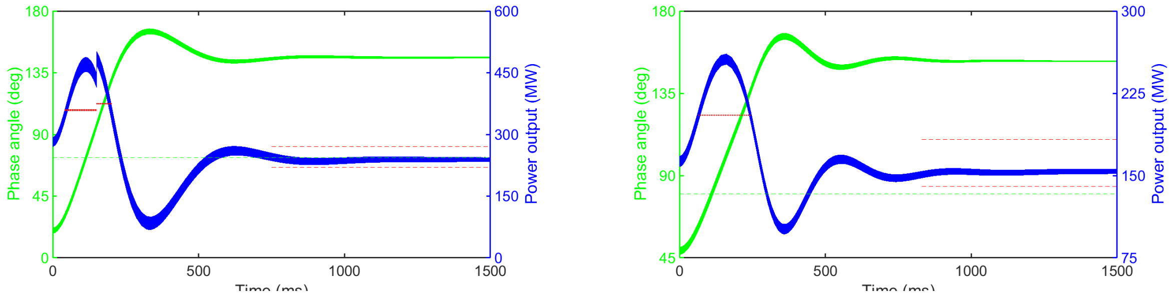
[3] Available at http://publish.illinois.edu/smartergrid/wscc-9-bus-system/



(a) First apply a constant input 400 MW for 154 ms. Then switched to a PID controller with $k_p = 7, k_i = 1.2, k_d = 0$.

(b) The phase angle converges to 152° and the power output converges to 163 MW.

Fig. 8: An attack to a 9-bus system



(a) Robustness of attack to 3 bus system.

(b) Robustness of attack to 9 bus system

Fig. 9: Robustness of the attack. The bus voltage and phase angle have a ±0.2 and ±1° uncertainty respectively. The reachable states of the phase angle and output power with the same attack. The reachable states do not intersect with *Detected* and are contained in *Unsafe* eventually.

modeler to choose various IEEE standard wiring configurations for the stators and rotors in the electrical machines. Our goal for this section is to study the mechanics of the progression of transient attacks, despite the NERC-CIP protections, in these realistic models. In the future, we hope to undertake the considerably harder problem of automatically synthesizing attacks for PowerWorld models.

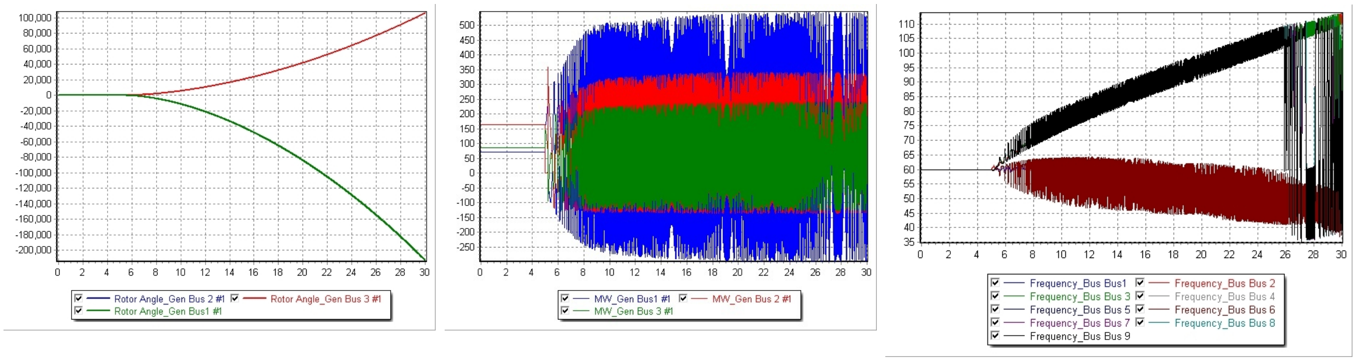
The attacks caused global instability against power systems that comply with the NERC-CIP N-1 contingency requirements. PowerWorld reported no violation as the result of NERC-CIP steady-state contingency that includes transmission line outages, transformer outages, and the power generator outages. The transient attack involved a sequence of discrete switches (transitions) changing the network topology, which in turn altered the system's continuous dynamics.

Each successful attack is a sequence of relay switches changing the topology of the power network. One attack, for example, starts at second 5, and opens the breaker on the bus 2, and waits for 1.18 seconds before re-closing it. Figure 10a shows the generators' rotor angles that fall out of step once the line gets re-closed. Intuitively speaking, the attack is successful because of how the generators operate. While the relay is open, the generator on bus 2 is isolated, that is, its output electrical power becomes zero. This is while its input mechanical power from the governor steam valve stays the same as before and is positive. This turns the power generator into an energy buffer, where its input mechanical power is a constant positive value while its output electrical power is zero. The buffered energy gets stored by the generator through its accelerated rotor, and hence its rotor's angular velocity increases. The rotor's angular velocity is directly related to the generator's generated electrical power frequency if it is connected to the rest of the grid (when its electrical power is not zero).

When the attacker connects the generator back to the grid by re-closing the line, its accelerated rotor will cause a frequency disturbance to the rest of the grid. Other adjacent generators attempt[4] to absorb the target generator's high frequency by increasing their rotor's angular velocity. Given the power system's transient swing equations [2], there is a limit to every generator's rotor angle beyond which the generator cannot re-stabilize itself whatsoever, which is the critical angle introduced in Section III-A. In the attack above, the wait caused the generator on bus 2 to gain so much acceleration and angular velocity that it was not able to recover the system stability by absorbing its increased frequency while at the same time not triggering the over-current relays upon reconnection. Figure 10b shows the electrical power output of the generators. As the graph shows, the generators' output power oscillate out of control due to the system's raised frequency. The system is unable to stabilize itself and the grid experiences a global collapse leading to a blackout. Figure 10c shows the system frequency for individual power generators, and how the system loses its synchrony (as discussed in Section II-B) as the result of subsequent discrete state transitions that even return to its original continuous dynamics.

In Section V-C, we discussed that synthesized attacks have a certain degree of robustness with respect to system parameters due to the continuity of the dynamics. Next, we show with a typical example that how there are limits to this

[4] This is due to the physics of the system and there is no explicit controller involved.

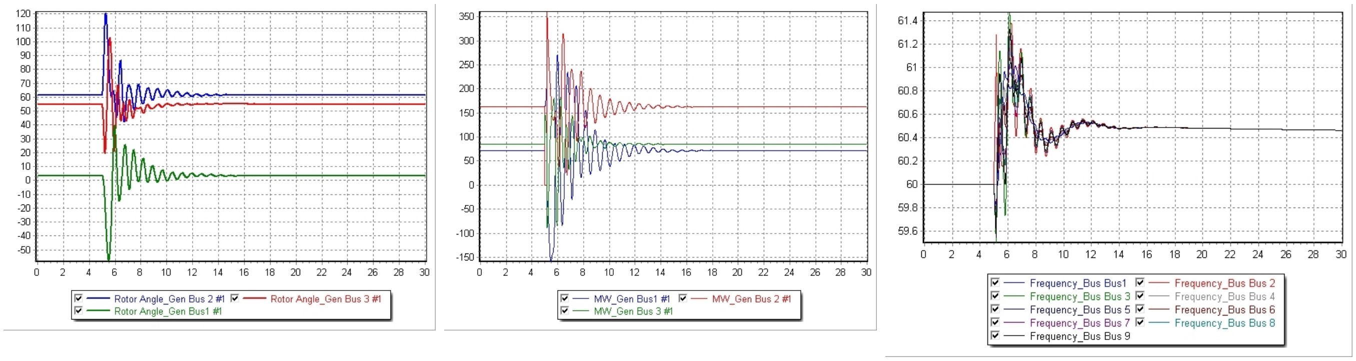


(a) Generator rotor angle falls out of step (destabilizes).

(b) Generator's output electrical power becomes unstable.

(c) Bus frequency cannot regain stability.

Fig. 10: Successful Transient Attack Analysis against a Power Generator



(a) Generator rotor angle stabilizes safely after a sequence of relay switches.

(b) Generator's output electrical power stabilizes.

(c) Power bus frequency stabilizes safely.

Fig. 11: Unsuccessful Transient Attack Analysis against a Power Generator

robustness. For example, if we change the timing of the above attack sequence so that then the relay on bus 2 closes after 1.17 seconds of waiting (0.01 seconds earlier) then the critical angle is reached and the attack is unsuccessful. First, in Figure 11a we show the evolution of the rotor angles for the power system generators under this benign sequence of switches (unsuccessful attack). The power system stabilizes and converges a safe steady-state value. Figure 11b and Figure 11c show the electrical power outputs for the power generators (MW) and the transient frequency dynamics of the power system buses, both stabilizing approximately 11 seconds after the switches.

Practical implications. In summary, these experiments suggest that switched transient attacks that evade NERC-CIP protection are indeed feasible in realistic power systems, and therefore, transient attacks and their protection should be considered by the governmental regulatory agencies, power system operators, and security researchers in the field. Secondly, it is possible to quantitative estimate robustness of synthesized attacks with respect to different network and attack parameters, and these measures may be useful as metrics for evaluating security of power networks.

VII. Related Work

Since the past real-world critical infrastructure attacks, there has been an increasing number of security protection solutions proposed. We review the most related work.

As a fundamental power system monitoring tool, state estimation is the process of fitting power sensor data to a system model and determining the current system state , e.g., using weighted least squares [27]. The estimated state is then used in stability analysis [14] through solving nonlinear AC or linear DC power flow equations for a series of "what if" scenarios, or contingency analysis [31], [14] that investigate the potential power system state in the case of an event, e.g., a generator outage. Almost all the current solutions, e.g., contingency analyses, do not consider the cyber-side controllers and/or take into account adversarial settings, and hence those solutions miss maliciously induced topological errors in modern cyber-physical infrastructures. Additionally, power system stability analysis concentrates on continuous dynamics only, and does not fully consider the possibility of subsequent discrete logic events in the system.

Recently, cyber security solutions have been proposed to harden critical infrastructures. These include practical best-effort techniques such as regulatory compliance such as attack tree analysis , NIST guidelines [28], and perimeter protection recommendations [21]. These approaches have been confirmed to be insufficient by the past major security incidents [1], and recently discovered fundamental security flaws in power grid control devices [30] and popular human machine interfaces (HMIs) from major vendors. From an adversarial viewpoint, the past cyber attacks are mostly not physics-aware, and do not complete the attack path by sending malicious control inputs to the underlying physical plant components. The very few real-world security incidents with physical impact [13], however, use manually crafted malicious control parameters such as setting them to an unsafe high value like in Stuxnet. Those trivial strategies are to be addressed by NERC-CIP regulations [29] that mandate local safety measure deployment to protect unsafe component operational points.

One specific related line of research is false data injection (FDI) attacks [22] that have been explored over the past few years. FDI assumes compromised set of sensors and make them send corrupted measurements to electricity grid control centers to mislead the state estimation procedures. The authors propose a system *observability* [22] analysis to determine the required minimal subset of compromised sensors to evade the electricity grid's bad data detection algorithms [23]. FDI's focus domain differs completely from our objective. Rather than sensor measurement corruption, we concentrate on malicious control inputs to the plant and by performing formal *controllability* analysis of the electricity grid for attack feasibility assessment. Additionally, [22] leverage a linear DC model of the grid for designing attacks, and hence ignores all the system non-linearities. The attack also ignores the discrete logic-based incidents, such as relay openings/closings that occur frequently in the grid. Using steady-state models, FDI [22] cannot evade NERC-CIP compliant protection schemes.

Automatic verification and specifically reachability analysis of hybrid models has enjoyed sustained attention from researchers in computer science and control theory for over three decades. The latest generation of tools from this research include Flow* [5], C2E2 [11], and Breach [8] that can approximate bounded time reach sets of nonlinear hybrid models. We refer the interested reader to the proceedings of the hybrid systems conference for recent developments [12]. The approach used in this paper in the transient attack synthesis algorithm, is comparable to the analysis approaches used in Breach and C2E2 that use static analysis methods for computing sensitivity or discrepancy measures of the model.

Algorithmic control synthesis, in contrast, is still in its infancy (see, for example, [15], [6]). Inductive synthesis is considered in [17], [18]. For synthesis of provably correct controller, the community adopt reachability-based approach [9], [7]. Particularly, a reachability-based approach similar to ours for parameter synthesis has been presented in [9].

VIII. Conclusions and Mitigations

In this paper, we presented an automated cyber-physical attack synthesis algorithm. Unlike previous work on electricity grid security analysis, the algorithm makes a complete use of both discrete and continuous dynamics of the system simultaneously. Through its formal hybrid system analyses, the algorithm demonstrates that the most recent governmental electricity grid cyber security NERC-CIP requirements can fall short in protecting our national grid against malicious adversarial parities. Our experimental results show that the use of non-steady-state system transient dynamics enables attackers design recipes that are not feasible using existing state-of-the-art attacks, which NERC-CIP is designed to prevent. The solution to mitigate the proposed transient attacks would be: *(i)* NERC-CIP should have transient-aware contingency analysis enforcement along with the N-1 steady-state contingencies; *(ii)* more fine-grained monitoring of system transients using more advanced sensors such as phasor measurement units (PMU) is required; and *(iii)* using algorithmic attack tool as a contingency analysis technique would guarantee the system security against transient attacks.

References

[1] The industrial control systems cyber emergency response team (ics-cert); available at https://ics-cert.us-cert.gov/. 2015.

[2] P. M. Anderson and A. A. Fouad. *Power system control and stability*. John Wiley & Sons, 2008.

[3] S. Bak, S. Bogomolov, and T. T. Johnson. Hyst: A source transformation and translation tool for hybrid automaton models. In *Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control*, pages 128–133. ACM, 2015.

[4] B. Bamieh and D. F. Gayme. The price of synchrony: Resistive losses due to phase synchronization in power networks. In *American Control Conference (ACC), 2013*, pages 5815–5820. IEEE, 2013.

[5] X. Chen, E. Ábrahám, and S. Sankaranarayanan. Flow*: An analyzer for non-linear hybrid systems. In *Computer Aided Verification*, pages 258–263. Springer, 2013.

[6] J. Daafouz, P. Riedinger, and C. Iung. Stability analysis and control synthesis for switched systems: a switched lyapunov function approach. *Automatic Control, IEEE Transactions on*, 47(11):1883–1887, 2002.

[7] J. Ding, E. Li, H. Huang, and C. Tomlin. Reachability-based synthesis of feedback policies for motion planning under bounded disturbances. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2160–2165, May 2011.

[8] A. Donzé. Breach, a toolbox for verification and parameter synthesis of hybrid systems. In *Computer Aided Verification*, pages 167–170. Springer, 2010.

[9] A. Donzé, B. Krogh, and A. Rajhans. Parameter synthesis for hybrid systems with an application to simulink models. In *Hybrid Systems: Computation and Control*, pages 165–179. Springer, 2009.

[10] P. S. Duggirala, S. Mitra, and M. Viswanathan. Verification of annotated models from executions. In *Proceedings of the Eleventh ACM International Conference on Embedded Software*, page 26. IEEE Press, 2013.

[11] P. S. Duggirala, S. Mitra, M. Viswanathan, and M. Potok. C2e2: A verification tool for stateflow models. In *Tools and Algorithms for the Construction and Analysis of Systems*, pages 68–82. Springer, 2015.

[12] M. Egerstedt and B. Mishra, editors. *Hybrid Systems: Computation and Control, 11th International Workshop, HSCC 2008, St. Louis, MO, USA, April 22-24, 2008. Proceedings*, volume 4981 of *Lecture Notes in Computer Science*. Springer, 2008.

[13] N. Falliere, L. O. Murchu, and E. Chien. W32.Stuxnet Dossier. Technical report, Symantic Security Response, Oct. 2010.

[14] J. Glover, M. Sarma, and T. Overbye. *Power System Analysis and Design*. Cengage Learning, 2011.

[15] L. Habets, P. J. Collins, and J. H. van Schuppen. Reachability and control synthesis for piecewise-affine hybrid systems on simplices. *Automatic Control, IEEE Transactions on*, 51(6):938–948, 2006.

[16] Z. Huang and S. Mitra. Proofs from simulations and modular annotations. In *Proceedings of the 17th international conference on Hybrid systems: computation and control*, pages 183–192. ACM, 2014.

[17] Z. Huang, Y. Wang, S. Mitra, G. E. Dullerud, and S. Chaudhuri. Controller synthesis with inductive proofs for piecewise linear systems: an smt-based algorithm. *arXiv preprint arXiv:1509.04623*, 2015.

[18] S. Jha and S. A. Seshia. A Theory of Formal Synthesis via Inductive Learning. *ArXiv e-prints*, May 2015.

[19] D. K. Kaynar, N. Lynch, R. Segala, and F. Vaandrager. *The Theory of Timed I/O Automata*. Synthesis Lectures on Computer Science. Morgan Claypool, November 2005. Also available as Technical Report MIT-LCS-TR-917.

[20] P. Kundur, N. J. Balu, and M. G. Lauby. *Power system stability and control*, volume 7. McGraw-hill New York, 1994.

[21] T. G. Lewis. *Critical infrastructure protection in homeland security: defending a networked nation*. John Wiley & Sons, 2006.

[22] Y. Liu, P. Ning, and M. K. Reiter. False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security (TISSEC)*, 14(1):13, 2011.

[23] Z. Lu and Z. Zhang. Bad data identification based on measurement replace and standard residual detection. *Automation of Electric Power Systems*, 13:011, 2007.

[24] K. Manamcheri, S. Mitra, S. Bak, and M. Caccamo. A step towards verification and synthesis from simulink/stateflow models. In *Proceedings of the 14th international conference on Hybrid systems: computation and control*, pages 317–318. ACM, 2011.

[25] S. Mitra. *A verification framework for hybrid systems*. PhD thesis, Massachusetts Institute of Technology, 2007.

[26] T. J. Overbye and J. D. Weber. Visualization of power system data. In *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, pages 7–pp. IEEE, 2000.

[27] S. J. Sheather. Weighted least squares. In *A Modern Approach to Regression with R*, Springer Texts in Statistics, pages 115–123. Springer New York, 2009.

[28] K. Stouffer, J. Falco, and K. Scarfone. Guide to industrial control systems (ICS) security. *NIST special publication*, pages 800–82, 2011.

[29] U.S. Department of Energy Office of Electricity Delivery and Energy Reliability. North american electric reliability corporation critical infrastructure protection (nerc-cip) standards; available at http://www.nerc.com/pa/Stand/Pages/CIPStandards.aspx, 2015.

[30] S. E. Valentine. *PLC code vulnerabilities through SCADA systems*. PhD thesis, University of South Carolina, 2013.

[31] M. Vutsinas. *Contingency Analysis Using Synchrophasor Measurements*. PhD thesis, Clemson University, 2008.