Adaptive Defending Strategy for Smart Grid Attacks

Jianye Hao, Eunsuk Kang,
Daniel Jackson
Massachusetts Institute of Technology
{jianye, eskang, dnj}@mit.edu

Jun Sun
Singapore University of Technology and Design
sunjun@sutd.edu.sg

ABSTRACT

One active area of research in smart grid security focuses on applying game-theoretic frameworks to analyze interactions between a system and an attacker and formulate effective defense strategies. In previous work, a Nash equilibrium (NE) solution is chosen as the optimal defense strategy [7, 9], which implies that the attacker has complete knowledge of the system and would also employ the corresponding NE strategy. In practice, however, the attacker may have limited knowledge and resources, and thus employ an attack which is less than optimal, allowing the defender to devise more efficient strategies.

We propose a novel approach called an adaptive Markov strategy (AMS) for defending a system against attackers with unknown, dynamic behaviors. The algorithm for computing an AMS is theoretically guaranteed to converge to a best response strategy against any stationary attacker, and also converge to a Nash equilibrium if the attacker is sufficiently intelligent to employ the AMS to launch the attack. To evaluate the effectiveness of an AMS in smart grid systems, we study a class of data integrity attacks that involve injecting false voltage information into a substation, with the goal of causing load shedding (and potentially a blackout). Our preliminary results show that the amount of load shedding costs can be significantly reduced by employing an AMS over a NE strategy.

Categories and Subject Descriptors

K.6.5 [**Security and Protection**]

Keywords

Smart grid security; Markov games; adaptive learning; data injection

1. INTRODUCTION

Power grid is one of the most critical infrastructures in existence today, whose disruption could cause severe economic, social, and environmental damages [3, 12], making it an attractive target for attackers. Protecting a power grid poses a number of challenges due to its wide geographical spread and complex interdependences among its components. To make the matter worse, modern grid systems are connected to the Internet, exposing itself to a wide range of cyber-attacks.

One active area of research in smart grid security focuses on applying game-theoretic frameworks to analyze the interactions between system defenders and attackers [11]. One commonly adopted game-theoretic model is called a Markov game [8], where the players' joint actions lead to probabilistic transitions between states of the system. In the previous applications of a Markov game to a smart grid attack, a *Nash equilibrium* (NE) solution is computed as the optimal defending strategy [7, 9].

The proposal on adopting a NE strategy is based on one crucial assumption: the attacker would also employ the corresponding NE strategy to attack the system. In practice, for a system as complex as a smart grid, the attacker may have neither a perfect knowledge of the system nor the computational capacity required to compute a NE strategy. More realistically, the attacker uses its experience and partial knowledge of the system to formulate what he/she believes to be the best strategy for maximizing the damage. Thus, it may be possible for the defender to employ a non-NE strategy that is just as effective, and potentially cheaper. The technical challenge is determining the attacker's behavior (initially unknown) and reformulate the defender's strategy dynamically.

We propose a novel approach called an adaptive Markov strategy (AMS) for defending a system against attackers with *unknown, dynamic* behaviors. Our AMS algorithm leverages an adaptive online learning technique to observe the attacker's behavior and reformulate an optimal defense strategy dynamically. It is guaranteed to converge to a best-response strategy against any stationary attacker, and also converges to a Nash equilibrium if the attacker is sufficiently intelligent to employ the AMS to launch the attack.

To demonstrate the effectiveness of our approach, we applied it to a class of smart grid attacks that involve injecting false voltage information into a grid substation, disrupting its voltage stability and causing load shedding. We experimentally evaluated the performance of our approach by applying it to a sample distribution system from the previous work [7]. Our preliminary results show that the amount of load shedding cost can be significantly reduced by employing an AMS over a NE strategy. Although we focus on one particular type of security attacks in this paper, our learn-

SEGS'14, November 7, 2014, Scottsdale, Arizona, USA.
Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM 978-1-4503-3154-8/14/11 ...$15.00.
http://dx.doi.org/10.1145/2667190.2667195.

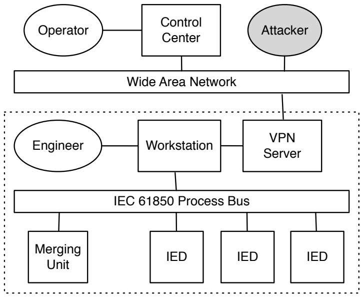


Figure 1: A high-level architecture of a smart grid. The dotted area represents a substation.

ing framework is general, and applicable to other classes of attacks that can be modeled as a Markov game.

The rest of the paper is organized as follows. We give an overview of related work in Section 2. In Section 3, we describe the false data injection attack and how this problem can be modeled as a Markov game. We present the AMS strategy and its properties in Section 4, and describe an evaluation of our approach in Section 5. We conclude with a discussion of future work in Section 6.

2. RELATED WORK

Game theory has been widely used as a mathematical tool to model and analyse the security issues in critical infrastructures such as smart-grid systems [11, 3, 12, 10]. The interaction between the defender and attacker is usually modeled as a single-shot Stackelberg game, in which the defender and attacker are considered as the leader and follower and make sequential moves. The goal thus is to identify the optimal defending strategy (e.g., the most critical set of components to protect) for the defender to minimize the potential loss of the system. However, in practice the defender and attacker may interact with each other repeatedly and the system evolves dynamically depending on their actions.

Markov games [8] later are adopted to model the repeated strategic interactions between the defender and attacker in smart-grid systems. In [9], one specific physical attack on the transmission lines of the smart-grid system is considered and the interaction between the defender and attacker is modeled as a Markov game, in which the system states (the status of the transmission lines) evolves based on their joint actions. A NE solution is adopted as the defending strategy for the system, which specifies which transmission lines to protect. In [7], one specific cyber attack (false data injection attack) is studied and the repeated attacker-defender cyber-interaction is modeled as a Markov game. Similar to [9], the NE solution is adopted as the defending strategy which determines which action to choose to perform intrusion detection. However, adopting the NE solution as the defending strategy is rational only when the attacker is also choosing the corresponding NE strategy to launch the attack, which may not hold in practice.

3. PROBLEM FORMULATION

3.1 Threat Model

One important requirement of a power grid is maintaining a stable supply of voltage throughout its distribution and transmission lines. Since many modern grids operate close to their stability limits, even slight instability or disruption can cause the voltage to drop below a critical level, forcing load sheddings and in the worst cast, blackouts. In order to maintain stability, each substation deploys a number of devices that monitor the voltage level and dynamically regulate power. A typical high-level architecture of a smart grid system, as described in IEC 61850, is shown in Figure 1, with the dotted box representing one of its substations. A *merging unit* collects various analog data from physical sensors (such as voltage and current levels) and converts them into digital packets, which are then broadcast over the process bus. A number of intelligent electronic devices (IEDs), connected to the process bus, look for anomalous readings in the packets and perform necessary regulatory actions to maintain the voltage stability. A *static synchronous compensator* (STATCOM) is one common type of device for voltage regulation, generating (absorbing) power when notified of low (high) voltage in the load.

Many modern substations allow engineers to perform maintenance remotely through a virtual private network (VPN) or other access mechanisms. While convenient, this also opens up the substation to a wide range of security attacks, since anyone on the Internet, having bypassed the VPN, may manipulate various devices through the workstation.

In this paper, we study on one particular type of smart grid attacks and corresponding defense mechanisms, originally proposed in [7].

3.1.1 Attack

We consider scenarios in which the attacker has successfully gained access to the workstation by exploiting weakness in its network perimeter (e.g., misconfigured firewall, weak password/keys). We assume that the attacker wishes to remain undetected, and so chooses not to perform drastic actions such as shutting down the entire substation. Finally, we assume that the attacker is not capable of physically tampering with the substation components (e.g., tripping transmission lines using circuit breakers).

We focus on one class of attacks, proposed in [7], where an attacker manipulates the behavior of a voltage regulator by injecting false voltage data into the process bus. From the attacker's point of view, this attack is particularly appealing since it can be carried out in a stealthy manner; by injecting a stream of packets with small deviations from normal voltage, the attacks may remain undetected by the system until it results in a catastrophic result (i.e., a blackout), similar to the way Stuxnet [6] was carried out.

In particular, given the actual voltage v, the attacker constructs a series of packet that indicates a voltage value of $kv + b$, where k and b are constant multiplicative and additive factors determined by the attacker. A STATCOM, having received the false measurement, may unnecessarily inject (possibly causing overvoltage) or absorb power from the load (under-voltage). The effect of an injection depends on the value of k:

- k < 0: These values of k represent readings that are an 180-degree out of phase from the actual voltage values, causing the STATCOM to inject power when it should be absorbed, and absorb power when it should be injected into the system.

- k = 0: The STATCOM will consistently receive a reading of b, falsely believing that the voltage level is stable and performing no regulatory actions.
- 0 < k < 1: These values of k represent a diminished version of the actual voltage values, causing the STATCOM to apply only partial regulatory actions.
- k >= 0: These values of k represent a false amplification of the actual voltage values, causing the STATCOM to inject or absorb more power than necessary.

Two factors are under consideration for the attacker when choosing the value of k's to be used: (1) it should cause enough disruption to the system to result in a load shedding, and (2) it should allow stealthy injection of packets, as large k values are likely to be easily detected.

3.1.2 Defense

One common mitigation against this type of attack is the use of encryption to ensure the integrity of the packets. However, relying on encryption as the sole protection mechanism may not be sufficient for two reasons: (1) a smart grid system has real-time requirements, with each packet being sent and processed in a span of milliseconds, and so resources required to encrypt and decrypt every packet may be too stringent, and (2) encryption keys are often stored as part of a configuration file, which may be easy to obtain once the attacker gains the entry to the workstation.

Instead, we consider a threshold-based method for detecting bad data packets [7]. In this method, the IED allocates an internal variable to keep track of the number of times $I - I_{ref}$ deviates from 0, where I is the current flowing through the current generator in the IED, and I_{ref} is a fixed reference current. If this number exceeds some predefined threshold (frequency variable τ) over a certain time period, then the IED concludes that the system may be under attack. The intuition behind this detection method is that $I - I_{ref}$ should remain close to 0 under normal circumstances, and that even in an unstable environment, should not vary no more frequently than τ.

The suitable values for τ for detecting the injection attack may vary depending on the specific range of the false voltage data injected by the attacker. Therefore, the challenge for the defender is to determine the right value for τ to maximize the detection success rate in response to the changes of the attacker's behaviors. We can adopt Markov game model to analyze this kind of strategic interaction between the defender and attacker, which will be described in the next section.

3.2 Markov Games

A Markov game is played between two players—the *attacker* and the *defender*—over a possibly infinite sequence of *rounds*. During each round, both players perform an *action* that may cause changes to the state of the system with some probabilities. Each player receives a corresponding payoff after selecting an action simultaneously. In our case, since the goal of the attacker is to trigger load shedding through false data injection attack, the attacker's payoff may be measured by the amount of load shedding that its action causes to a grid given the action of the defender. Conversely, the payoff for the defender is the negation of the amount of load shedding. The Markov game here is *zero-sum*; that is, the sum of the attacker's and defender's payoffs is zero.

Formally, a Markov game consists of:

- N: a finite number of players. In our setting, there are two players (defender and attacker), i.e., $N = \{d, a\}$.
- A_i: the action space of each player. A_d and A_a represent the set of defender's and attacker's actions, respectively.
- S: a finite set of system states.
- Pr: transition probability function. Given the current state s and the joint action (d, a), $Pr(d, a, s, s')$ returns the probability that the system transits from state s to s' when the defender and the attacker perform actions d and a, respectively.
- R_i: payoff function of the players. Given $s \in S$, $a \in A_a$, and $d \in A_d$, $R_d(s, d, a)$ returns the expected payoff of the attacker when the joint action (d, a) is performed under state s. Since we are interested in zero-sum games, the attacker's corresponding payoff $R_a(s, d, a)$ is exactly the negation of $R_d(s, d, a)$, i.e., $R_a(s, d, a) = -R_d(s, d, a)$.

The behaviors of the attacker and defender from Section 3.1 can be modeled as follows. First, the action space of the attacker can be defined as

$$A_a = \{k_1, k_2, ..., k_{N_a}\} \tag{1}$$

where $k_1, k_2, ...$ are real constants, N_a is the size of A_a, and for some $i \leq N_a$, k_i corresponds to the injection of a packet that indicates a voltage level of $k_i b$ (i.e., falsely magnifying the voltage reading by a factor of k_i).

Similarly, the set of the actions that the defender may perform is defined as:

$$A_d = \{\tau_1, \tau_2, ..., \tau_{N_d}\} \tag{2}$$

where $\tau_1, \tau_2, ...$ are integer constants, N_d is the size of A_D, and for some $j \leq N_d$, τ_j refers to the defender deploying the detection method with the threshold of τ_j (i.e., the number of times $I - I_{ref}$ is allowed to cross 0).

A player's *strategy* ϕ is a function that given some state s, returns a probability distribution over the set of actions that the player may perform in s.

4. ADAPTIVE PROTECTION

In the previous section, we have shown how we can model a security attack as a Markov game between the attacker and the defender. From the system architect, one important question is how the system should choose its defending strategy in order to minimize the amount of damage caused by the attacker. A conventional approach in the game theory is computing a *Nash equilibrium* (NE) strategy: that is, both the attacker and the defender play a strategy that would maximize the payoffs for both of them [7, 9]. The rationale for adopting the NE solution is that neither the attacker nor the defender can do better by choosing a different strategy.

However, in practice, choosing a NE strategy is not necessarily optimal for the defender, since it depends on a number of assumptions that might not hold. First, computing a NE relies on a perfect knowledge of the system; however, in reality, the attacker might not have the capability to collect enough information to construct an accurate model of

Algorithm 1 Description of AMS

```
Compute NE strategy ($\pi_i^*, \forall i \in \{d, a\}$)
repeat
  Initialize $h_a^{prev}$, $h_a^{curr}$ to nil
  $s = s_0$, $\beta = false$, $t = 0$
  Set defender strategy $\phi_d$ as NE strategy ($\phi_d = \pi_d^*$)
  while true do
    for $r$ : 0 to $N^t$ do
      Play($\phi_d(s)$)
      Update($h_a^{curr}$)
    end for
    $h_a^{prev} = h_a^{curr}$
    $t := t + 1$
    if Distance($h_a^{curr}$, $\pi_a^*$) $> \epsilon_e^t$ then
      break
    end if
  end while
  $\phi_d$ = RandomStrategy()
  while true do
    for $r$ : 0 to $N^t$ do
      Play($\phi_d(s)$)
      Update($h_a^{curr}, h_a^{prev}$)
    end for
    $t := t + 1$
    if $\beta$ = true then
      if Distance($h_a^{curr}$, $h_a^{prev}$) $> \epsilon_s^t$ then
        break
      end if
    end if
    $h_a^{prev} = h_a^{curr}$
    $\beta := true$
    $\phi'_d$ := BestResponseStrategy($h_a^{curr}$)
    if $V(s, \phi'_d, h_a^{curr}) > V(s, \phi_d, h_a^{curr}) + 2|A|^{|S|}\epsilon_s^{t+1}\mu$, $\forall s \in S$ then
      $\phi_d = \phi'_d$
    end if
  end while
until
```

a smart grid. Second, since the attackers are humans, it is likely that they would launch attacks based on their intuition or past experience, which might be different from the NE strategy. Lastly, if there are multiple equilibria, the players may not pick the same matching strategy.

Instead, an effective defending strategy should be *adaptive*, i.e., it should be able to learn the attacker's strategy and dynamically compute the *best response* strategy to counter the attacking strategy. However, assuming that the attacker may change its strategy arbitrarily is neither useful nor practical. Therefore, to make our technique feasible, we assume that the attacker's strategy is *stationary* (i.e., the probability of choosing each action is unchanged under the same state)[1].

An effective defending strategy must satisfy the following two desirable properties [2].

Rationality A *rational* defending strategy must always learn to play the best response strategy given that the attacker is adopting a stationary attacking strategy. Satisfying this property guarantees that the cost to the system can be minimized as long as the attacker's strategy is stationary. Note that a NE strategy is a special case of stationary strategy.

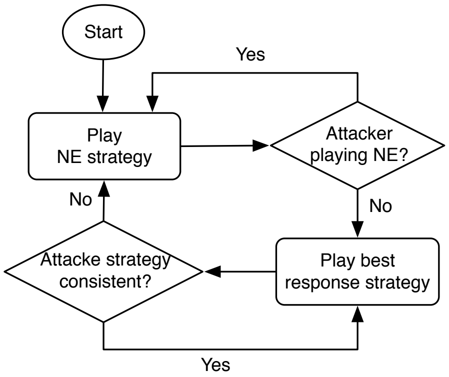


Figure 2: Overview of the AMS

Convergence The defending strategy must always converge to a stationary strategy under self-play. This property takes into consideration the possibility that the attacker might be as intelligent as the defender and employ the same adaptive strategy as the defender. We can see that under self-play, if both rationality and convergence properties are satisfied, the defender and attacker will eventually converge to a NE strategy. This means that the maximum cost to the system can be bounded to the cost when the attacker adopts a NE strategy, even when the attacker is as intelligent as the defender.

4.1 AMS: Adaptive Markov Strategy

In this section, we propose an algorithm for computing an adaptive Markov strategy (AMS) for defending a system against security attacks. Our algorithm is based on the AWESOME algorithm [4], which computes adaptive defending strategies for repeated games only (i.e., a special case of Markov games where the system has exactly one state); we extend AWESOME to Markov games where the system may have any finite number of states.

The overview of the AMS algorithm is shown in Figure 2. AMS begins by assigning an NE strategy as the defending strategy, and observes the behavior of the attacker for some fixed number of rounds (called a period). If the estimated strategy of the attacker is consistent with its NE strategy, then AMS keeps the original NE as the defending strategy. Otherwise, it computes a new best response strategy to play against its current estimation of the attacker's strategy. After playing the new strategy for another period of rounds, AMS checks whether attacker's strategy remains the same as the one from the previous period; if not, this implies that the previous estimation of the attacker's strategy was incorrect, and so AMS restarts the whole process again by retreating to the original equilibrium strategy.

Before introducing the AMS algorithm in details, we need to explain a few terms first. First, to determine whether the attacker is employing the NE or any other stationary strategies, we define the *distance* between two strategies to compare whether they are the same or not.

[1] We allow the attacker to change its strategy to another stationary strategy during the game.

Definition 1. *The* distance $Distance(\phi_1, \phi_2)$ *between two stationary strategy* ϕ_1 *and* ϕ_2 *is:*

$$Distance(\phi_1, \phi_2) = \max |\phi_1(s, a) - \phi_2(s, a)|, \forall a \in A_s, s \in S \tag{3}$$

where A_s *is the action space at state* s *and* S *is the state space, and* $\phi_1(s, a)$ *and* $\phi_2(s, a)$ *is the probability that action* a *is played at state* s *for strategy* ϕ_1 *and* ϕ_2 *respectively.*

Second, given two strategies ϕ_1 and ϕ_2, we define the value $V(s, \phi_1, \phi_2)$ of playing strategy ϕ_1 against strategy ϕ_2 under state s, which is defined as the sum of the discounted expected payoff obtained over infinite number of interactions.

Definition 2. *The value* $V(s, \phi_1, \phi_2)$ *of playing strategy* ϕ_1 *against strategy* ϕ_2 *under state* s *is defined as follows,*

$$\begin{aligned} V(s, \phi_1, \phi_2) = &R(s, \phi_1(s), \phi_2(s)) + \\ &\delta \sum_{s \in S} Pr(\phi_1(s), \phi_2(s), s, s') V(s', \phi_1, \phi_2) \end{aligned} \tag{4}$$

where δ is the discounting factor reflecting the relative importance of future payoffs and $Pr(\phi_1(s), \phi_2(s), s, s')$ is the probability that the system state transits from s to s' given that the players choose actions $\phi_1(s)$ and $\phi_2(s)$ respectively. We can construct one equation for V-value of each state $s \in S$ following Definition 2, and thus the value of each state can be calculated by solving the system of $|S|$ linear equations using different techniques such as iterative methods [5].

The AMS algorithm (Algorithm 1) takes place over consecutive *periods* (where each period is some number of round). Initially, the AMS begins by playing the precomputed NE strategy[2] for the initial period N^0 (Line 5) and estimates the strategy of the attacker based on the actions taken in this period (Line 7 to 10). If the *distance* between the estimated strategy h_a^{curr} and the NE strategy π_a^* of the attacker is larger than the given threshold (line 13), the attacker is considered playing a non-NE strategy, and a random strategy is chosen as the defense strategy for the next period (Line 17).

At the end of the next period, AMS computes the best response strategy ϕ_d' against the current estimated strategy h_a^{curr} of the attacker based on the last period's interaction (Line 31).[3] If for every state $s \in S$, the difference between the V-value of ϕ_d' against the h_a^{curr} and that of ϕ_d is larger than the given threshold $2|A|^{|S|}\epsilon_s^{t+1}\mu$ (where $|A|^{|S|}$ represents the total number of pure strategies of the Markov game and μ is the payoff difference between the AMS player's best and worse outcomes), the current defending strategy ϕ_d is replaced by a more optimal strategy ϕ_d' (Line 32-34).

At the end of each following period, AMS compares the estimated strategy h_a^{curr} and h_a^{prev} of the attacker in the last and preceding periods (Line 25). If the distance between these two is larger than the given threshold ϵ_s^t, it indicates that the opponent is not playing according to the estimated strategy h_a^{prev}, and the AMS will restart by breaking from the second while loop (Line 26). Otherwise, the AMS computes a best response strategy ϕ_d' based on the last period's interaction, and employs ϕ_d' as its strategy if it is more optimal than ϕ_d (Line 31-34). This process repeats as indicated by the outer *Repeat* loop.

The remaining question is how the set of parameters of the AMS algorithm should be adjusted, described as follows.

Definition 3. *A schedule of adjusting the parameters* $\{\epsilon_e^t, \epsilon_s^t, N^t\}$ *is valid if*

- $\epsilon_e^t, \epsilon_s^t$ *are decreased monotonically and converge to zero eventually.*
- *the value of* N^t *is increased monotonically to infinity.*
- $\Pi_{t \in \{1,2,\ldots\}}(1 - A_S \frac{1}{N^t(\epsilon_s^{t+1})^2}) > 0$, *where* A_S *is the total number of actions of the defender summed over all states.*

4.2 Properties of the AMS

As previously mentioned, an effective defending strategy must satisfy two desirable properties: rationality and convergence. It can be theoretically proved that the AMS satisfies both properties, which are formalized as the following two theorems:

Theorem 1. *Given a valid schedule of adjusting the parameters, if the attacker employs a stationary attacking strategy, the defender adopting AMS eventually converges to a best response to the attacker's strategy with probability one.*

Proof. *We provide a sketch of the proof, which has two parts. First, we prove that with a non-zero probability, the AMS never restarts. We can show that the joint probability that the AMS not restarts for every period t is greater than zero based on the triangle inequality and ChevyshevâĂŹs inequality theorem. Second, we prove that the probability that the AMS never restarts and does not converge to a best response strategy against the attacker is 0 by continuity and ChevyshevâĂŹs inequality theorem. By proving both parts, we can conclude that the AMS always converge to a best response strategy against the attacker with probability 1.* □

Theorem 2. *Given a valid schedule, if both the defender and attacker employ the AMS, they eventually converge to a NE with probability one.*

Proof. *The sketch of the proof is as follows. Similar to the proof of Theorem 1, we prove this theorem by dividing it into two parts. First, we prove that with a positive probability, the AMS for both players will always be within the first while-loop in Algorithm 1, i.e., always playing the corresponding NE strategy. To prove this, we only need to prove that the probability that the distance between the estimated strategy and the precomputed equilibrium strategy is not greater than the value of Îţte for all periods t is larger than zero.*

In the second part, we need to prove is that the probability that the AMG strategy never restarts but does not converge to equilibrium strategy is zero. We only need to show that in this case, one player (attacker or defender) adopting the AMS will eventually switch its strategy, which would trigger both players to restart.

[2] Note that we only need to compute the minmax strategy instead since for a zero-sum Markov game, the minmax/maxmin strategy for each player is equivalent to its corresponding NE strategy [13]. The generalized value iteration algorithm [13] can be used to compute the minmax strategy efficiently.

[3] The generalized value iteration algorithm can be used here to compute the best response strategy in a Markov game [13].

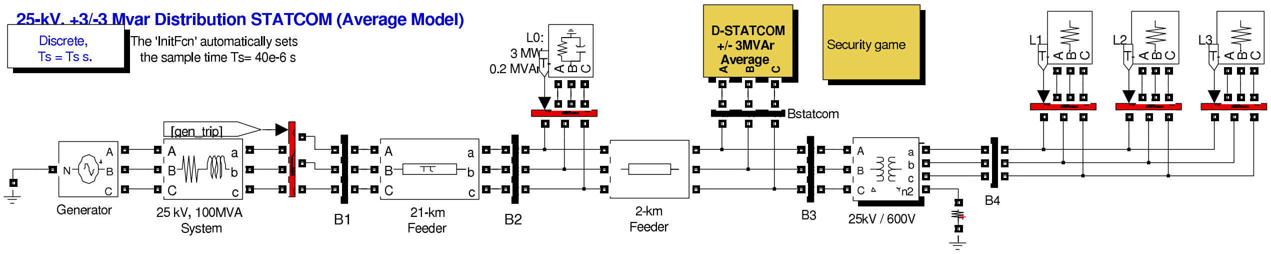


Figure 3: An example distribution system with 1 generator and 4 buses [7], originally from the D-STATCOM model in SimPowerSystems [1].

By combining both parts, we can conclude that the defender and attacker will eventually converge to an NE with probability one if both of them adopt the AMS. □

5. PRELIMINARY EVALUATION

In this section, we describe a preliminary evaluation of the performance of the AMS for defending a substation against false data injection. We used a model of 1-generator 4-bus distribution system shown in Figure 3 (adopted from [7]) as the testbed, and compared the AMS's performance with that of the NE strategy proposed in [7, 9]. In this distribution system, the generator supplies electric power to four loads (L0, L1, L2, L3); a single STATCOM, connected to the system through Bus 3 (B3), regulates the voltage level by injecting (absorbing) reactive power to the system based on voltage feedback.

5.1 Markov Game

For our experiment, we adapted the same Markov game used in the previous study of the false data injection attack [7]. To obtain the various parameters of the Markov game (attacker/defender actions, transition probabilities, and payoffs), the authors of the previous work performed a MATLAB/Simulink simulation of the testbed system in Figure 3 under various load conditions. Since the construction of the Markov game is not our contribution, we only briefly discuss the Markov model that we adapted for our study and refer the readers to [7] for more details on how the game was constructed from the simulation[4].

In this game, the state space is abstracted into a set S of two states, $S = \{s_1, s_2\}$, where s_1 and s_2 represent the states in which the system experiences (1) no load shedding, and (2) some amount of load shedding, respectively. The defender's action set is represented by

$$A_d = \{\tau_1 = 11, \tau_2 = 32\}$$

which correspond to the two thresholds that the defender uses to detect an injection attack.

The attacker's action set is denoted by

$$A_a = \{k_1 = 1.1, k_2 = -0.8\}$$

which correspond to two false voltage values that the attacker may choose to inject into the STATCOM. The values for the attacker's actions were chosen because based on the output of the Simulink simulation, they were effective in causing a load shedding, and avoided $I - I_{ref}$ crossing zero a large number of times.

The transition probabilities and the expected immediate payoff of the defender and the attacker for each joint action were also obtained based on the output of the Simulink simulation. Given a pair of joint actions (a_d, a_a), the transition probability from state s_i to state s_j is measured as the expected probability that the system starts from state s_i and ends in state s_j in a session due to the execution of the joint action (a_d, a_a):

$$Pr(d_1, a_1) = \begin{vmatrix} 43/45 & 2/45 \\ 1/2 & 1/2 \end{vmatrix} \quad Pr(d_2, a_1) = \begin{vmatrix} 0 & 1 \\ 1/47 & 46/47 \end{vmatrix}$$

$$Pr(d_1, a_2) = \begin{vmatrix} 48/49 & 1/49 \\ 0 & 1 \end{vmatrix} \quad Pr(d_2, a_2) = \begin{vmatrix} 25/32 & 7/32 \\ 7/17 & 10/17 \end{vmatrix}$$

The expected payoff for the attacker/defender under given state s and joint action pair (a_a, a_d) is the expected amount of load shedding by performing those actions from that state; in [7], this value was computed as the ratio of the total energy shed (E_s) throughout the simulation in that state over the duration of the load shedding (T_s). Since this is a zero-sum game, the payoffs for the defender are exactly the negation of those for the attacker[5]:

$$R_a(s_1) = \begin{vmatrix} 44/46 & 0 \\ 42/49 & 24/33 \end{vmatrix} \quad R_a(s_2) = \begin{vmatrix} 2 & 2.50 \\ 2 & 2.15 \end{vmatrix}$$

$$R_d(s_1) = \begin{vmatrix} -44/46 & 0 \\ -42/49 & -24/33 \end{vmatrix} \quad R_d(s_2) = \begin{vmatrix} -2 & -2.50 \\ -2 & -2.15 \end{vmatrix}$$

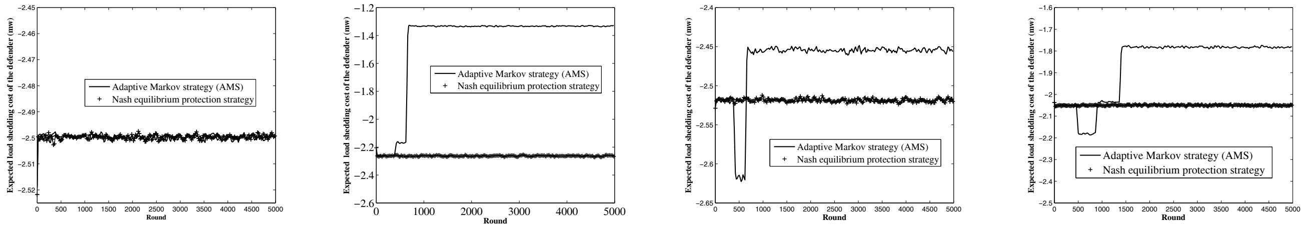
5.2 Experimental Results

To evaluate the effectiveness of the AMS, we compare the performance of both the AMS and NE defending strategy under different attacker's strategies. We considered four different scenarios, where the attacker employs (1) a NE strategy, (2) a strategy where the attacker always performs a_1, (3) an a_2-only strategy, and (4) a random strategy. In each scenario, we ran a simulation of the Markov game for 5000 rounds, and measured the expected load sharing costs of the defender when it employed the AMS and NE strategy.

Figure 4a shows the dynamics of the expected load shedding costs when the attacker selects the NE strategy and the defender employs the NE and AMS defending strategy respectively. We can see that both defending strategies result in roughly the same load shedding cost. This is as expected; recognizing that the attacker is employing the NE strategy,

[4] In particular, we studied the Markov game generated from Scenario 2 in Section VI(B) of [7], where load L3 is a variable load instead of a fixed load.

[5] [7] also discusses the cost of false positives of the detection method for the defender, which we omit here.



(a) ϕ_0: the NE strategy

(b) $\phi_1(s_1) = \phi_1(s_2) = a_1$

(c) $\phi_2(s_1) = \phi_2(s_2) = a_2$

(d) ϕ_3: choose a_1/a_2 randomly

Figure 4: Expected load shedding cost of the system when the attacker employs different attacking strategies.

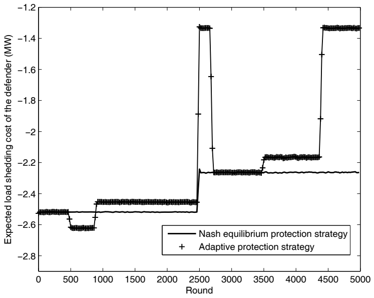


Figure 5: Expected load shedding cost when the attacker employs the strategy ϕ_2 for Stages 0 - 2500, and ϕ_1 for Stages 2500 - 5000.

the AMS learns to employ its optimal counterpart NE strategy for the defender.

Figure 4b shows the expected load shedding costs when the attacker employs a strategy ϕ_1 where he or she always performs the single action a_1, and the defender adopts the NE and AMS defending strategy respectively. We can observe that the system's load shedding cost of the system is significantly reduced when the system employs the AMS. Intuitively, when the AMS recognizes that the attacker uses such a simple strategy, it also constructs an optimal strategy that exploits the attack pattern, thus minimizing the load shedding cost. Similarly, if the attacker employs the strategy ϕ_2 where he always selects a_2 (Figure 4c), the AMS constructs a corresponding optimal strategy that results in a significantly reduced cost over the NE strategy. The temporary drop-off around the 500th round is due to the fact that before the AMS can determine to an accurate estimation of the attacker's strategy, it may temporarily employ a strategy that is less than optimal.

Figure 4d illustrates the load shedding costs when the attacker employs the strategy ϕ_3 where he randomly alternates between the two actions, a_1 and a_2 under each state, and the defender adopts the NE and AMS defending strategy respectively. Similar to the previous scenario, the AMS initially incurs a greater cost, but as its estimation of the attacker's strategy stabilizes (around the 1500th round), it continually significantly outperforms the NE strategy.

Lastly, we consider a scenario when the attacker may not adopt a stationary strategy, and instead switch to a different strategy during the game. Figure 5 considers the case when the attacker initially employs the stationary strategy ϕ_2 and switches to another stationary strategy ϕ_1 in the middle period (after 2500 round). From Figure 5, we can see that AMS can learn to exploit the attacker's dynamic behaviors to significantly reduce the load shedding cost compared with NE strategy.

6. CONCLUSION AND FUTURE WORK

In this paper, we discussed why the convention approach of using a NE strategy in a Markov game might not be an optimal choice for the system defender, due to a number of assumptions about the attacker that may not hold in practice, especially in a system as complex as a smart grid. We proposed a new type of adaptive strategy called the AMS, and performed a preliminary evaluation of the technique on one class of security attacks on smart grid systems—injecting false voltage information.

Further investigation is needed to test the feasibility of our approach in practical settings. One potential limitation of the AMS, as currently designed, is the number of rounds required to converge to an accurate estimation of the attacker and obtain the best response strategy. In a smart grid, the number of packets transmitted to IEDs per second is typically in hundreds, and so in the data injection attack, it is conceivable that the AMS may converge to an optimal strategy in matter of minutes. However, other types of attacks that are less frequent (e.g., where an attacker's action involves physical hampering), a dynamic technique such as the AMS might not be suitable. We plan to investigate possible ways to reduce the duration of the convergence (e.g., using an approximation). Furthermore, we plan to study the effectiveness of the AMS on other types of smart grid attacks, and explore techniques for scaling the computation of the AMS to larger distribution systems.

Acknowledgements.

We would like to thank the anonymous reviewers for their valuable comments and suggestions on this paper. The first author was supported by the MIT-SUTD Postdoctoral Fellowship.

7. REFERENCES

[1] *SimPowerSystems documentation: D-STATCOM (Average Model)*, 2014. `http://www.mathworks.com/help/physmod/sps/examples_v2/d-statcom-average-model.html`.

[2] Michael Bowling and Manuela Veloso. Convergence of gradient dynamics with a variable learning rate. In *ICML*, pages 27–34, 2001.

[3] Gerald Brown, Matthew Carlyle, Javier Salmerón, and Kevin Wood. Defending critical infrastructure. *Interfaces*, 36(6):530–544, 2006.

[4] Vincent Conitzer and Tuomas Sandholm. Awesome: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents. *Machine Learning*, 67(1-2):23–43, 2007.

[5] Stanley C Eisenstat, Howard C Elman, and Martin H Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM Journal on Numerical Analysis*, 20(2):345–357, 1983.

[6] James P Farwell and Rafal Rohozinski. Stuxnet and the future of cyber war. *Survival*, 53(1):23–40, 2011.

[7] Yee Wei Law, Tansu Alpcan, and Marimuthu Palaniswami. Security games for voltage control in smart grid. In *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*, pages 212–219, 2012.

[8] M. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of ICML'94*, pages 322–328, 1994.

[9] Chris YT Ma, David KY Yau, Xin Lou, and Nageswara SV Rao. Markov game analysis for attack-defense of power networks under possible misinformation. *Power Systems, IEEE Transactions on*, 28(2):1676–1686, 2013.

[10] Ali Pinar, Juan Meza, Vaibhav Donde, and Bernard Lesieutre. Optimization strategies for the vulnerability analysis of the electric power grid. *SIAM Journal on Optimization*, 20(4):1786–1810, 2010.

[11] Walid Saad, Zhu Han, H Vincent Poor, and Tamer Basar. Game-theoretic methods for the smart grid: An overview of microgrid systems, demand-side management, and smart grid communications. *Signal Processing Magazine, IEEE*, 29(5):86–105, 2012.

[12] Javier Salmeron, Kevin Wood, and Ross Baldick. Analysis of electric grid security under terrorist threat. 2004.

[13] Olivier Sigaud and Olivier Buffet. *Markov decision processes in artificial intelligence*. John Wiley & Sons, 2013.