An Adaptive Markov Strategy for Defending Smart Grid False Data Injection From Malicious Attackers

Jianye Hao, Eunsuk Kang, Jun Sun, Zan Wang, Zhaopeng Meng, Xiaohong Li, and Zhong Ming

***Abstract*—We present a novel defending strategy, adaptive Markov strategy (AMS), to protect a smart-grid system from being attacked by unknown attackers with unpredictable and dynamic behaviors. One significant merit of deploying AMS to defend the system is that it is theoretically guaranteed to converge to a best response strategy against any stationary attacker, and converge to a Nash equilibrium (NE) in case of self-play (the attacker is intelligent enough to use AMS to attack). The effectiveness of AMS is evaluated by considering the class of the data integrity attacks in which an attacker manages to inject false voltage information into the intelligent voltage controller in a substation. This kind of attack may cause load shedding and potentially a blackout. We perform extensive simulations using a number of IEEE standard test cases of different scales (different number of buses). Our simulation results indicate that AMS enables the system to experience much lower amount of load shedding compared with an NE strategy.**

***Index Terms*—Intrusion detection, agent-based modeling, learning.**

I. Introduction

ONE OF the foremost critical infrastructures in modern society today is power grid, and its disruption and damage could cause potentially severe damages in terms of economic, environmental and social costs [5], [24]. Therefore, it has become an appealing target for more and more potential attackers. One major characteristic of power grid is its wide geographical spread and complicated interdependencies among different components, thus we are faced with a number of challenges to protect a power grid from physical attacks. The connection of modern grid systems (aka., smart-grid systems) to the Internet results in a worsen situation, making itself vulnerable to a large variety of cyber-attacks.

The attack-defend interactions between system operators and malicious attackers usually involve deliberate thinking and also the effectiveness of a defending strategy closely depends on the current attacking strategy and vice versa. One simple example would be considering physical attacks on power lines or substations. The effectiveness of a defending strategy in terms of which set of lines to protect (or monitor) depends on the attacking strategy adopted in terms of which set of lines to sabotage. This kind of dependencies also commonly exist in cyber-attack scenarios such as false data injection attack detection we focus on in this paper. Besides, the system states naturally evolve depending on whether the attacks are successful or not, which is determined by the attacking and defending strategies implemented. Game theory provides us with a number of candidate game-theoretic frameworks to model such kinds of system dynamics and analyze the strategic interactions between attackers and defenders. Recent years have seen increasing efforts and interests in adopting game-theoretic frameworks to study the interactions between attackers and system defenders [15], [19], [20], [23], and use game-theoretic solutions to devise defending strategies. One commonly used game-theoretic framework is Markov game [17], where the joint action choices of the players result in probabilistic transitions between system states. Previous work usually computes the *Nash equilibrium* (NE) solution of the corresponding Markov game modeling of the system as the defending solution to smart-grid attacks [15], [20].

However, the underlying rationale of deploying a NE strategy to defend relies on the crucial assumption of the attacker's behavior: the corresponding NE strategy is employed by the attacker to launch the attack. This assumption is reasonable for the cases when the attack is an insider attack, in which the attacker can access all necessary knowledge of the system beforehand. In practical systems, however, an outside attacker may have neither sufficient information of the system nor the necessary computational ability to obtain a Nash equilibrium strategy. More realistically, as a human being, the attacker may devise what he/she perceives best for maximizing the cost to the grid based on his/her personal experience and partial knowledge of the system. Thus, it is very likely that a

Manuscript received November 6, 2015; revised March 28, 2016 and June 3, 2016; accepted September 4, 2016. Date of publication September 16, 2016; date of current version June 19, 2018. This work was supported in part by the National Key Technology Research and Development Program of China under Grant 2015BAH52F01-1, in part by the National Natural Science Foundation of China under Grant 61304262, Grant 61202030, and Grant 71502125, and in part by the Tianjin Research Program of Application Foundation and Advanced Technology under Grant 16JCQNJC00100. Paper no. TSG-01430-2015. *(Corresponding author: Zan Wang.)*

J. Hao and Z. Wang are with the School of Computer Software, Tianjin University, Tianjin 300350, China (e-mail: jianye.hao@tju.edu.cn; wangzan@tju.edu.cn).

E. Kang is with the University of California at Berkeley, Berkeley, CA 94720 USA (e-mail: eskang@csail.mit.edu).

J. Sun is with the Singapore University of Technology and Design, Singapore (e-mail: sunjun@sutd.edu.sg).

Z. Meng is with the School of Computer Software, Tianjin University, Tianjin 300350, China, and also with the Tianjin University of Traditional Chinese Medicine, Tianjin 300193, China (e-mail: mengzp@tju.edu.cn).

X. Li is with the School of Computer Science and Technology, Tianjin University, Tianjin 300350, China (e-mail: xiaohongli@tju.edu.cn).

Z. Ming is with the School of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: mingz@szu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2016.2610582

defender may employ a non-NE strategy that he believes to be effective to launch an attack. In this case, simply adopting the pre-computed NE strategy as the defending strategy is not the most effective solution in terms of minimizing the damage to the grid. In response to this, we are required to have the capability of estimating the attacker's behavior and make the best response in an adaptive manner.

To this end, we propose a novel approach named adaptive Markov strategy (AMS) to protect a system from being attacked by attackers with unpredictable and dynamic behaviors. AMS utilizes an online learning approach to dynamically compute an optimal defending strategy against the currently estimated behavior of an attacker. We adopt two criteria from multiagent learning literature: *rationality* and *convergence*, which we believe an effective defending strategy in smart grid system should satisfy. Through satisfying the above two properties, a strategy would always converge to a best-response defending strategy (minimizing the damage) against any stationary attacker, and also converge to a Nash equilibrium if the attacker is intelligent enough to adopt the same AMS to attack. AMS is theoretically provable to satisfy both of the above criteria.

To illustrate our approach's effectiveness, we applied AMS to a class of false voltage information injection attacks in a grid substation, which may disrupt the distribution system's voltage stability and result in load shedding. We first empirically evaluated the effectiveness of AMS using a representative distribution system adopted in [15], and then further evaluated the performance using a number of IEEE standard test cases of different scales. Our results indicate that AMS can greatly help to reduce the load shedding cost compared with a NE strategy. Note that though only one particular type of security attacks in the smart grid is considered here, our defending framework is also applicable to any other types of attacks given that its underlying dynamics can be formulated as a Markov game.

The structure of the paper is organized as follows. Section II describes the false voltage information injection attack in a smart grid and formalize it as a Markov game. Section III presents the details of AMS and its theoretical properties, and follows by the extensive experimental evaluation of AMS in Section IV. Finally Section V concludes and discusses possible future work.

II. Problem Formulation

A. Threat Model

One critical requirement of a power system is to maintain its voltage within an acceptable range during its power distribution and transmission process. Unexpected voltage dropping below the critical level would result in load shedding and even worse, blackouts. In transmission and distribution systems, voltages can be controlled by a wide range of devices that either inject, absorb or redirect reactive power flow. Usually a *merging unit* first collects various analog data (such as voltage and current levels) from physical sensors, which are then converted into digital formats. Finally those digital packets are broadcasted over the process bus. Different types of intelligent electronic devices (IEDs) are connected to the process bus and examine any anomalous readings within the packets. If found, necessary regulatory actions are performed to keep the voltage level stable. It is worth noting the difference with local control devices under which no communication is required. One commonly adopted intelligent device for voltage regulation is *static synchronous compensator* (STATCOM), and it can generate (or absorb) reactive power when the information of low (or high) voltage on the lines is received through communication.

In this work, we focus on one representative type of cyber attacks in smart-grid distribution systems, which exploits the above mentioned voltage regulation mechanism. Note that there also exist other types of false-data injection attacks exploiting other aspects of weakness of a smart-grid system, e.g., attack on the control signals of circuit breakers in a substation or undervoltage load shedding relays, which are out of the scope of this paper. In the following we will describe the details of this type of attack on active voltage regulators (e.g., STATCOM) and the corresponding defense mechanisms we can use [15].

1) Attack: By injecting false voltage data into the process bus, a voltage regulator can be mislead to make incorrect decisions. This kind of attack can be done in a stealthy manner. The attack can inject a sequence of packets containing the voltage information which slightly deviates from the normal level, which leads to the malfunction of STATCOM with the attacks remaining undetected. Eventually this may lead to load shedding or blackout to certain loads in the grid system. This is similar to the way Stuxnet [11] was carried out to sabotage Iran's nuclear control system.

Specifically, given the actual voltage v, the attacker generates a consequence of packets that represent a voltage value. Theoretically an attacker could have infinitely number of ways to manipulate the voltage value, which is infeasible to list all of them. To make our analysis feasible, we adopt one basic linear pattern of $kv + b$, where k and b are constant factors that are selected by the attacker. We also assume that there is no noise during transmission. After receiving the false measurement, A STATCOM controller may make incorrect decisions by injecting (possibly resulting in over-voltage) into or absorbing power from the distribution system (possibly resulting in under-voltage). In linear forms, any possible value of the false voltage can be obtained by varying the value of k while keeping the value of b as a constant. Thus we only need analyze the consequence of the false voltage information injection by considering different values of k:

- $k < 0$: It indicates that the voltage from the reading are an 180-degree out of phase from the real voltage values, i.e., the falsified voltage value and the true voltage value would always be in opposite sign. Thus it may cause the STATCOM to inject power when it should be absorbed, while absorb power when it should be injected.
- $k = 0$: The STATCOM will consistently receive a reading of b, falsely believing that the voltage level is stable and performing no regulatory actions.

- $0 < k < 1$: The STATCOM will receive a decreased version of the actual voltage values, thus the STATCOM may apply only partial regulatory actions.
- $k >= 1$: The STATCOM will receive a false amplification of the actual voltage values, thus the STATCOM may inject or absorb more reactive power than needed.

When an attack selects which value of k should be used, the following two factors are usually under consideration: (1) the system should be disturbed enough to lead to a load shedding, and (2) the attack should be stealthy enough to be undetected. Obviously setting the value of k too large or too small are more vulnerable to be easily identified. For example, if the modified voltage value is larger or smaller than 20% of the normal voltage level, it would be detected easily by using a saturation filter. By selecting appropriate value of k, the attack may successfully bypass the detection and lead to load shedding. The unpredictable feature is reflected from the fact that an attack may adopt different value of k, which significantly affect the optimal detection action of the defender. Finally, it is worth noting that the falsified measurements from attackers is essentially different from other bad measurements due to transmission noise, measurement noise or instrument failures. The noise measurements usually follows certain distribution predictions and can be overcome by applying state estimation techniques (e.g., Kalman filter). In contrast, the falsified measurements from attackers cannot be simply treated as the transmission noise. Another distinction between them is that falsified data from attackers is usually much more consistent than other bad data measurements that would trigger wrong control actions.

2) Defense: One common way of mitigating this class of cyber attack in traditional networks is resort to encryption techniques to check whether a packet has been tampered. However, encryption protection mechanisms may not be feasible due to the limited computational capabilities, strict timing requirements and high data sampling rates in the smart grid. Though a number of encryption-based approach have been proposed for communication and control in smart-grid, their performance does not seem to fulfill the stringent timing requirement of the smart grid in practice [10], [26]. A hardware-based encryption mechanism may address these issues, but they are still not common among IEDs, thus we do not discuss it in this paper.

In the paper, we consider a false voltage information detection method based on examining the trend of the current flow [15]. In more details, given the reference current I_{ref} of the current regulator and the current flowing I, we keep track of the number of times that $I - I_{ref}$ deviates from 0, [1, Ch. 5]. If the number of times crossing zero over a certain time period exceeds certain predefined threshold (frequency variable τ), then we may determine that the voltage information we receive is false (an attack has occurred). The rational behind is that the difference between I and I_{ref} should always stabilize around 0 under normal operations, and should not vary more than τ even in environments with dynamic loads [15]. It is also worth noting that current-based detection can only detect whether there is currently an attack or not, but cannot prevent an attack beforehand.

The selection of the appropriate values for τ may closely depend on how the attacker tampers the voltage information during its attack. Thus, the challenging task of the defender is the selection of the optimal value for τ to maximize the detection success rate in response to the dynamic changes of the attacker's strategies. We can see that the strategic interactions between the defender and attacker are repeated and the right value for τ should be adaptively adjusted each round depending on the attacker's behaviors. Given the system's current state (e.g., normal state or state under attack), the system evolves based on the joint actions of the defender and attackers. The payoff of each player also depends on the joint action of the players each round. Therefore, it makes Markov game the perfect candidate to model this kind of strategic interaction between the defender and attacker, which will be introduced in the following section.

Finally, when the current voltage meter is detected to have been compromised, the STATCOM will bring another backup meter online and take the current meter offline. Certain disinfection operation (e.g., refreshing the firmware including cryptographic keys in the Flash memory) will be done on the infected meter and it will be used as a backup meter.

B. Markov Games

We consider a Markov game between two players—an *attacker* and a *defender* with a possibly infinite rounds of interactions. Each round both players select an *action* to perform, and the system states change according to the joint action of the players with some probabilities. The payoff that each player receives depends on the current state and their joint action. For the attacker, its payoff can be measured by the load shedding cost incurred on a grid. Conversely, the defender suffers from the same amount of cost, and thus his payoff can be modeled as the negation of the attacker's payoff. The Markov game between the players is *zero-sum*.

Formally, a two-player zero-sum Markov game can be represented as a tuple $\langle S, N, A_i, Pr, R_i \rangle$:

- S: the set of system states.
- $N = \{d, a\}$: the set of players: a defender and an attacker.
- A_i: each player i's action space, $\forall i \in N$.
- Pr: the probabilistic state transition function. Given a state $s \in S$ and a joint action (d, a), the function $Pr(d, a, s, s')$ gives the probability that the system state changes from s to s' under the joint action (d, a).
- R_i: each player i's payoff function. Given a state $s \in S$, $a \in A_a$, and $d \in A_d$, the function $R_a(s, d, a)$ gives the average payoff of the attacker under state s when (d, a) is executed. For a zero-sum game, the sum of the attacker and the defender's payoffs are always zero, i.e., $R_a(s, d, a) + R_d(s, d, a) = 0$.

Next we model the attacker and defender's behaviors based on the informal description in Section II-A. First, the attacker's action space can be modeled as follows,

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

where $k_1, k_2, \ldots$ are real numbers and N_a denotes the size of A_a, For each $i \leq N_a$, k_i represents injecting a false packet

with the voltage level of $k_i v + b$ (i.e., the voltage reading is falsely magnified by a factor of k_i). Note that there might exist other ways of modeling the attacker's behaviors rather than the linear model adopted here. However, the Markov game modeling and the adaptive Markov strategy we will introduce next section are general and can be applied on as long as A_a is well-defined.

Similar to the attacker's actions, the action set of the defender can be modeled as follows:

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

where $\tau_1, \tau_2, \ldots$ are integer numbers and N_d is the size of A_D. For each $j \leq N_d$, τ_j corresponds to the defender detecting the false voltage information using the threshold of τ_j (i.e., the average frequency that $I - I_{ref}$ crosses 0).

Finally, we define a player's (defender or attacker) *strategy* ϕ as a function that for each state s, gives a probability distribution over the actions that the player may execute under state s.

III. Adaptive Protection

As a defender, we are interested in how the system should be defended to maximize the detection probability and thus minimize the amount of damage incurred. We argue that 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. A strategy of an attacker (defender) is defined as a function which maps each state to a probability distribution over its action space. It is reasonable to assume that an attacker may change its strategy from time to time. Specifically we propose that an effective defending strategy must satisfy the following two desirable properties [4].

Rationality - A defending strategy is rational if it always learns towards the best-response strategy as long as the attacker is employing a fixed attacking strategy. By satisfying this property, it is guaranteed that the system cost can be minimized given that the attacker's strategy is unchanged. Satisfying this property also requires a strategy to be adaptive, i.e., adjusting the defending behaviors according to the dynamic changes of the attacker's behavior to ensure that the best response is achieved eventually.

Convergence - The defending strategy must always converge to a fixed strategy under the case of self-play. This property considers the cases when the attacker might be sufficiently intelligent to employ the same adaptive strategy as the defender. It is not difficult to verify that, if both of the above properties are satisfied, the players would converge to a NE eventually under self-play. This indicates that we have the lowest bound on the system's cost under the case of self-play: the worst-case cost is equal to that when the attacker adopts a NE strategy.

A number of learning strategies have been proposed to satisfy some of the above properties in the multiagent learning literature, however, all of them suffer from either of the following two problems: 1) long learning periods are required before converging to the best response strategy, thus resulting in significant losses during learning period and failing to make timely response [21], [22]; 2) some strategies are designed for repeated game setting only and also do not satisfy all the above properties [7], [9], [12], [27]. Thus we cannot directly apply the existing learning strategies into the malware detector placement problem. In this paper, we propose an adaptive Markov strategy (AMS) for Markov games which satisfies all the above properties.

A. AMS: Adaptive Markov Strategy

We start with introducing the definitions of a few terms necessary for describing the AMS algorithm. First, given any two strategies, we need a criterion of *distance* to check whether they are the same or not, which is defined as follows.

Definition 1: Given two strategies ϕ and ϕ', the distance $Distance(\phi, \phi')$ between them is:

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

where A_s denotes the action space at state s, S represents the set of states, and $\phi(s, a)$ and $\phi'(s, a)$ denotes the respective probability of selecting action a at state s following strategy ϕ and ϕ'.

Second, we define the value $V(s, \phi_1, \phi_2)$ of playing strategy ϕ_1 against strategy ϕ_2 under state s, This value is calculated as the sum of the discounted expected payoff obtained over an infinite number of rounds.

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

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

Here $0 \leq \delta \leq 1$ denotes the discounting factor indicating the weight of future payoffs and $Pr(\phi_1(s), \phi_2(s), s, s')$ is the system state transition probability from s to s' given that action $\phi_1(s)$ and $\phi_2(s)$ are chosen by the players. For each state $s \in S$, its V-value corresponds to one equation following Definition 2. Thus, we can obtain the value of each state by computing the system of $|S|$ linear equations using techniques such as iterative methods [8].

The overall AMS algorithm is shown in Algorithm 1, and the list of symbols used in AMS is summarized in Table I. Initially, the AMS starts by selecting the precomputed NE strategy as the defending strategy for an initial period of rounds (Line 5). The strategy of the attacker can be estimated from the history of attack, which is computed as the frequency of actions taken by the attacker during this period (Line 7 to 10). Next AMS checks whether the *distance* between the estimated strategy h_a^{curr} of the attacker and its NE strategy π_a^* is larger than the given threshold (line 13). If yes, AMS assumes that the attacker is following a non-NE attacking strategy, and then a random strategy is chosen as the defending strategy for the next period (Line 17).

After the second period terminates, AMS calculates the best-response strategy ϕ_d' against the attacker's estimated

Algorithm 1 Description of AMS

1: Compute a NE strategy $(\pi_i^*, \forall i \in \{d, a\})$
2: **repeat**
3: Initialize h_a^{prev}, h_a^{curr} to nil
4: $s = s_0$, $\beta = false$, $t = 0$
5: Set defender strategy ϕ_d as NE strategy $(\phi_d = \pi_d^*)$
6: **while** true **do**
7: **for** $r : 0$ **to** N^t **do**
8: Play($\phi_d(s)$)
9: Update(h_a^{curr})
10: **end for**
11: $h_a^{prev} = h_a^{curr}$
12: $t \coloneqq t + 1$
13: **if** Distance(h_a^{curr}, π_a^*) $> \epsilon_e^t$ **then**
14: **break**
15: **end if**
16: **end while**
17: ϕ_d = RandomStrategy()
18: **while** true **do**
19: **for** $r : 0$ **to** N^t **do**
20: Play($\phi_d(s)$)
21: Update(h_a^{curr}, h_a^{prev})
22: **end for**
23: $t \coloneqq t + 1$
24: **if** β = true **then**
25: **if** Distance(h_a^{curr}, h_a^{prev}) $> \epsilon_s^t$ or Distance(h_{self}^{curr}, h_{self}^{prev}) $> \epsilon_s^t$ **then**
26: **break**
27: **end if**
28: **end if**
29: $h_a^{prev} = h_a^{curr}$
30: $\beta \coloneqq true$
31: $\phi_d' \coloneqq$ BestResponseStrategy(h_a^{curr})
32: **if** $V(s, \phi_d', h_a^{curr}) > V(s, \phi_d, h_a^{curr}) + 2|A|^{|S|}\epsilon_s^{t+1}\mu(s)$, $\forall s \in S$ **then**
33: $\phi_d = \phi_d'$
34: **end if**
35: **end while**
36: **until** the end of detection

TABLE I
LIST OF SYMBOLS IN THE *AMS*

Symbols	Description
$\|A\|$	the number of actions in a single-state matrix game of the Markov game
$\|S\|$	the number of states in the Markov game
$\mu(s)$	the V-value difference between the AMS player's best and worse outcomes under state s
N^t	the number of rounds in period t
h_a^{curr}	the current round estimated strategy of the attacker
h_{self}^{curr}	the current round estimated strategy of the AMS agent
h_a^{prev}	the previous round estimated strategy of the attacker
h_{self}^{prev}	the previous round estimated strategy of the AMS agent
ϵ_s^t	the parameter threshold for comparing the difference between h_a^{prev} and h_a^{curr} at the current round t
ϵ_e^t	the parameter threshold for comparing the difference between h_a^{curr} and π_a^* at the current round t

strategy h_a^{curr} based on the second period's interaction history (Line 31). If the difference between the V-value of ϕ_d' against h_a^{curr} (see Definition 2) and the V-value of ϕ_d against h_a^{curr} ($\forall s \in S$) is larger than the threshold $2|A|^{|S|}\epsilon_s^{t+1}\mu(s)$, AMS replaces the current defending strategy ϕ_d by strategy ϕ_d' (Line 32-34). Note that $|A|^{|S|}$ represents the total number of pure strategies of the Markov game and $\mu(s)$ denotes the payoff difference between the defender's best and worse outcomes. Therefore the overall value of $2|A|^{|S|}\epsilon_s^{t+1}\mu(s)$ thus reflects the upper threshold for the V-value difference between $V(s, \phi_d', h_a^{curr})$ and $V(s, \phi_d, h_a^{curr})$ when the distance between h_a^{curr} and h_a^{prev} is within ϵ_s^{t+1}.

The same checking procedure is repeated for the following periods. Besides, AMS also evaluates whether the attacker is employing the same strategy starting from the end of the third period. Specifically AMS compares the distance between the estimated strategy h_a^{curr} and h_a^{prev} of the attacker (Line 25): if their distance exceeds the threshold ϵ_s^t, then AMS assumes that the attacker is not following the strategy h_a^{prev} as we predicted, thus AMS will restart to the beginning of itself (break from Line 26). Otherwise, AMS recomputes its best response strategy ϕ_d' against h_a^{curr}, and resort to strategy ϕ_d' if it is better than ϕ_d (Line 31-34).

The set of parameters of the AMS algorithm should be adjusted in a valid manner to ensure its convergence property, which is 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.

Since we model the interaction between attacker and defender as a zero-sum game (the sum of the attacker and defender's payoffs is always 0), calculating its Nash equilibrium strategy can be transformed into computing its maxmin/minmax strategy of the Markov game [25]. Therefore, the complexity of computing its Nash equilibrium of a Markov game can be reduced to be polynomial in the size of the Markov game (its states and action space). One common approach of computing the maxmin/minmax strategy of a Markov game is based on the extension of Shapley's value iteration algorithm [25], which is omitted due to space limitation. We also remark that both agents are assumed to compute the same Nash equilibrium under the case of self-play (both agents use AMS strategy). Finally, given the estimated strategy of the attacker, we can calculate the best-response strategy for the defender based on the generalization of the value iteration technique [25] as follows.

We first define the Q-value $Q_d(s, d, a)$ of the defender as its expected long-term value starting at state s by choosing action d (the attacker chooses action a), and the attacker and defender choose its estimated strategy ϕ_a and the best-response strategy against the estimated strategy of the attacker thereafter. This can be formally represented as follows,

$$Q_d(s, d, a) = R_d(s, d, a) + \delta \sum_{s' \in S} Pr\big(s, d, a, s'\big) V_d'\big(s'\big) \quad (5)$$

which $V_d'(s')$ is the long-term expected payoff of the defender if the attacker and defender choose its estimated strategy ϕ_a and the best-response strategy against the attacker respectively.

The value of $V_d'(s)$ for any state s can be defined based on $Q_d(s, a, d)$ as follows,

$$V_d'(s) = \max_{\phi_d(s) \in \Pi(A_d)} \sum_{d \in A_d} \left(\sum_{a \in A_a} Q_d(s, d, a) \phi_a(s, a) \right) \phi_d(s, d) \quad (6)$$

where $\Pi(A_d)$ is the set of all the probability distributions (mixed strategies) over the action set A_d of the defender.

Based on the generalization of the value iteration technique [25], we can obtain the best-response strategy for the defender against the estimated strategy of the attacker by updating the V-values and Q-values repeatedly until convergence.

B. Properties of the AMS

We propose that an effective defending strategy should at least satisfy the following two properties: rationality and convergence. As shown in the following theorems, AMS satisfies both of them. We omit the proofs due to space limitation.

Theorem 1: Given a valid schedule of adjusting the parameters, if the attacker's strategy is fixed, with probability one, AMS will eventually converge to a best response strategy to the attacker's strategy.

Proof: We prove this theorem by dividing it into two parts. First, we prove that with a non-zero probability, the AMS strategy never restarts. Second, we prove that the probability that the AMS strategy never restarts and does not converge to a best-response strategy against the attacker is 0. By proving both parts, we can reach the conclusion that the AMS strategy will converge to a best-response strategy against the attacker with probability 1.

From Algorithm 1, we know that the AMS strategy restarts if and only if the second *break* statement is executed, at certain period t; that is, either condition 1) $Distance(h_a^{curr}, h_a^{pref}) > \epsilon_s^t$ or condition 2) $Distance(h_{self}^{curr}, h_{self}^{pref}) > \epsilon_s^t$ is satisfied. For condition 1), based on the triangle inequality and the fact the ϵ_s^t is decreasing, we know that

$$\begin{aligned} Distance\big(h_a^{curr}, h_a^{prev}\big) > \epsilon_s^t \implies & Distance\big(h_a^{curr}, h_a\big) \\ & + Distance\big(h_a^{prev}, h_a\big) \\ > \epsilon_s^t \implies & Distance\big(h_a^{curr}, h_a\big) > \frac{\epsilon_s^t}{2} \\ & \vee Distance\big(h_a^{prev}, h_a\big) \\ > \frac{\epsilon_s^t}{2} \implies & Distance\big(h_a^{curr}, h_a\big) \\ > \frac{\epsilon_s^{t+1}}{2} & \vee Distance\big(h_a^{prev}, h_a\big) > \frac{\epsilon_s^t}{2} \end{aligned}$$

Thus, we only need to prove that with a positive probability, for all period t, the following is true: $Distance(h_a^{curr}, h_a) \leq \frac{\epsilon_s^{t+1}}{2}$.

The probability P that $Distance(h_a^{curr}, h_a) \leq \frac{\epsilon_s^{t+1}}{2}, \forall t$ can be represented as $\Pi_{t\in\{1,2,\ldots\}}(1 - Pr(Distance(h_a^t, h_a) > \frac{\epsilon_s^{t+1}}{2}))$, which is greater than

$$P' = \Pi_{t\in\{1,2,\ldots\}}\left(1 - \sum_{s\in S}\sum_{a\in A_a} Pr\left(\left|h_a^t(s,a) - h_a(s,a)\right| > \frac{\epsilon_s^{t+1}}{2}\right)\right)$$

Since $E(h_a^t(s,a)) = h_a(s,a)$, and observing that $Var(h_a^t(s,a)) \leq \frac{1}{4N^t}$, by applying Chebyshev's inequality theorem [3], we can reach the conclusion that $P' > \Pi_{t\in\{1,2,\ldots\}}(1 - A_S\frac{1}{N^t(\epsilon_s^{t+1})^2})$, where A_S is the total number of actions summed over all states. Since this is always greater than 0 for a valid schedule, we can say that with a positive probability, the first condition will never be satisfied.

Next we prove that the second condition will also never be reached with a positive probability. First, we know that when the AMS algorithm reaches the second while-loop, there is probability $\frac{1}{|A|^{|S|}}$ that the AMS chooses a strategy which is the best-response strategy to the stationary strategy of the opponent (a random strategy is selected). We only need to show that the AMS will never change its strategy once this best-response strategy is selected, which is guaranteed by the upper threshold we use $(2|A|^{|S|}\epsilon_s^{t+1}\mu(s))$ for comparing the difference between $V(s, \phi_d', h_a^{curr})$ and $V(s, \phi_d, h_a^{curr})$. Thus we can state that with a positive probability, the AMS strategy will never restart.

The second part is to prove that the probability that the AMS strategy never restarts but never converges to the best-response strategy is 0. There are two possible conditions under which this might happen. The first condition is that the AMS strategy always stays in the first *while-loop* (Line 6-16). In this case, it is impossible that the opponent is playing its corresponding Nash equilibrium strategy, since the AMS strategy would be playing the best-response strategy otherwise. Let us denote the actual strategy of the attacker as ϕ_a, its Nash equilibrium strategy as ϕ_a^*, and its current estimated strategy in period t is ϕ_a^t. Given a state s and an action a, let us denote $d = |\phi_a(s,a) - \phi_a^*(s,a)|$. From Chebyshev's inequality theorem, we know that $Pr(|\phi_a^t(s,a) - \phi_a(s,a)| < \frac{d}{2}) \geq 1 - \frac{1}{N^t d^2}$, which goes to 1 as t goes to infinity. Also since ϵ^t will become less than $\frac{d}{2}$ eventually, we can have that $|\phi_a^t(s,a) - \phi_a(s,a)| < \frac{d}{2} \implies |\phi_a^t(s,a) - \phi_a^*(s,a)| > \frac{d}{2} \implies |\phi_a^t(s,a) - \phi_a^*(s,a)| > \epsilon_e^t$. This implies that the AMS strategy will execute the *break* (Line 14) command and jump out of the *while-loop* (Line 6-16) eventually.

The second condition is that the AMS strategy always stays in the second while-loop (Line 18-35), but the AMS strategy is not playing the best response to the attacker's strategy. In this case, we only need to prove that the AMS strategy will eventually switch its strategy to the best-response one with probability 1. If the payoff of playing a pure strategy ϕ_1 against the attacker's true strategy ϕ_a is k less than that of playing another strategy ϕ_2, then by continuity, for some ϵ, for any strategy ϕ_a' that is within the distance of ϵ of the true strategy of the attacker ϕ_a, the payoff obtained by playing ϕ_1 against ϕ_a' should be at least $\frac{k}{2}$ less than that of playing ϕ_2. Similar to the proof in the first condition, we know that $Pr(Distance(\phi_a^t, \phi_a) < \epsilon) \to 1$ as t goes to infinity. Also we know that $2|A|^{|S|}\epsilon_s^{t+1}\mu$ will become smaller than $\frac{k}{2}$ eventually with probability 1. Thus we can have the conclusion that the AMS agent will switch its strategy eventually with probability 1. ■

Theorem 2: Given a valid schedule, under the case of self-play, the defender and attacker eventually converge to a Nash equilibrium with probability 1.

Proof: We prove this theorem by dividing it into two parts. First, we prove that with a positive probability, the AMSs for

both players will never restart and are always within the first while-loop. Second, we need to prove is that the probability that the AMS strategy never restarts but does not converge to equilibrium strategy is zero.

For the first part, we only need to prove that the AMSs can always stay in the first while-loop for all periods t with positive probability. This is in equivalent to prove the following statement: there is positive probability that $Distance(h_a^t, \pi_a^*) > \epsilon_e^t$ is always false for all periods t, which is similar to the first part proof in Theorem 1, and can be proved using the Chebyshev's inequality theorem in a similar manner.

Secondly, we need to prove is that the probability that the AMS strategy never restarts but does not converge to equilibrium strategy is zero. In this case, the AMS strategy must be within the second while-loop, and not playing the best-response to each other. Thus similar to the proof in the second part of Theorem 1, we only need to show that one player (either attacker or defender) adopting the AMS will eventually switch its strategy since at least one player's strategy is not the best-response strategy to its opponent. Also since an AMS agent is always synchronized with its opponent under self-play (Line 25), the strategy deviation of one AMS agent would trigger both AMS players to restart. Thus it leads to a contradiction.

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. ■

C. Discussion on Convergence Time

In previous section, we have shown that the AMS is theoretically guaranteed to converge to the best-response strategy if the attacker employs a stationary strategy, and a NE strategy if the attacker also employs the same AMS. Intuitively the number of interactions required before convergence is increased as the action and state space are increased. Thus it is expected that the convergence time would also be increased given that the interaction frequency (attacking frequency) is unchanged. However the practical performance of the AMS is not significantly sacrificed before converging to the best response due to the following observations:

- the initial NE defending strategy can be precomputed before the AMS is deployed on the system, thus does not violate the real-time protection requirement of the system. By initially adopting NE defending strategy, the defender can avoid being exploited when the attacker's strategy is unknown. Specifically for any time when the attacker's strategy is not equal to (deviates from) the NE strategy we expect, based on the definition of NE, we know that its payoff under each state is always lower than that obtained by choosing the NE strategy. Since the game itself is also zero-sum, we can see that the actual long-term payoff of the defender is always higher than that obtained when both players choose the NE strategy (the lower bound).
- in AMS, the strategy only needs to be updated at the end of a period of rounds and the length of each period is also gradually increased. The calculation of best-response strategy can be done during the last few interactions during each period to ensure it is ready when it is needed, which thereby does not violate the real-time requirement of the system.

Finally, we remark on the behaviors of AMS when it encounters an adaptive attacker, which can change its strategy dynamically. Theorem 1 and 2 only characterizes the behaviors of AMS when it is against a fixed-strategy attacker and an AMS attacker respectively. However the AMS strategy also works when the attacker can dynamically change its strategy. Without loss of generality, let us assume that the attacker adopts strategy ϕ_1 for certain periods and then change to another strategy ϕ_2. Following the description of AMS, we can verify that the AMS first converges to the best response against ϕ_1, and also continue detecting whether the attacker is still following the same strategy at the end of each period. After the attacker switches to strategy ϕ_2 which has been detected, the AMS will restart by learning from scratch until converging to the best response towards ϕ_2. In Section IV, we will consider a case where the attacker can dynamically change its strategy and evaluate the performance of the AMS against it to support this claim. It is worth noting that the time between the attacker's strategy change should be sufficient for the AMS to learn towards a best response. Otherwise, it would be equivalent as the attacker is adopting a random strategy, from which no useful defending strategy can be learned and deployed.

IV. Experimental Evaluation

We present the evaluation results of AMS for defending power distribution systems against false data injection attacks compared with NE strategy under a number of test cases with increasing sizes. Unless mentioned otherwise, the initial period length N^0 is set to 500 rounds and is increased by 10 per period. The value of ϵ_e^t and ϵ_s^t are decreased by 50% at the end of each period.

A. One-Generator Four-Bus Distribution System

In the distribution system in Figure 1, one generator provides power to four loads (L0-L3), and a STATCOM connecting to the system through Bus 3 (B3) is in charge of regulating the voltage levels of the system. The STATCOM executes the detection algorithm every 0.5 seconds until an anomaly is detected. The regulation is done through injecting (or absorbing) reactive power to (from) the system based on the voltage feedback sent from merging units. Note that we assume that the STATCOM is the only device to regulate the voltage level near the load side, and other components may also adjust the voltage level of other parts of the power network (e.g., near the generator side), which is out of the scope of our setting. If the voltage near the bus 2 or 4 drops below certain threshold, the corresponding under voltage load shedding (UVLS) relay would shed the corresponding load. The specific rules adopted in this testbed is as follows [16].

- if voltage $V_{B4} < 0.94$ p.u. for 0.4 s, then shed load L1;
- if voltage $V_{B4} < 0.92$ p.u. for 0.3 s, then shed load L2;
- if voltage $V_{B4} < 0.90$ p.u. for 0.2 s, then shed load L3;
- if voltage $V_{B2} < 0.90$ p.u. for 0.4 s, then shed load L0.

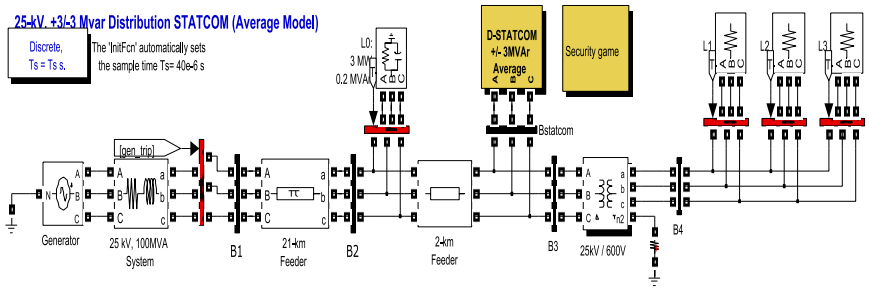


Fig. 1. An example distribution system [15], originally from the D-STATCOM model in SimPowerSystems [2].

Here p.u. is short for per unit and is calculated as the ratio of the absolute voltage value to the reference voltage value.

1) Markov Game: Since the construction of the Markov game is not our focus here, we only give the specific instance of the Markov model for our study obtained from Matlab/Simulink simulation and interested readers may refer to [15]. Note that the Markov game itself is a discrete abstraction of the continuous dynamics of interactions. The state transitions can be considered as the snapshots of the system states which capture the key moments of the system. The payoffs and transition probabilities in the Markov game are obtained from real-time simulation on the testbed system using Matlab/Simulink. Specifically, the transition probability from state s_1 to s_2 under a joint action (a_1, d_1) is determined as the empirical frequency of the system transiting from state s_1 to s_2 after (a_1, d_1) is performed. The payoff of the attacker under state s_1 and a joint action (a_1, d_1) is determined as the average load shedding cost over the session time starting from state s_1 given the attacker and defender's actions are fixed as a_1 and d_1.

We abstract the system's state space into two states, $S = \{s_1, s_2\}$, where s_1 and s_2 correspond to the cases when the system suffers from (1) zero load shedding, and (2) certain amount of load shedding, respectively. In general, there is an infinite number of actions available for both the defender and attacker. However, in practice, to make the analysis feasible, we adopt the empirical game-theoretic analysis approach [18] here, assuming that both players would choose actions from a finite set of actions.

The attacker's action set is represented as $A_a = \{k_1 = -0.8, k_2 = 1.1\}$, which are the false voltage values that the attacker may choose to modify and send back to the STATCOM. The reason why we choose the above values as the attacker's actions is that they are the most stealthy (i.e., the thresholds to distinguish the attacking actions are close to the threshold of being normal) and effective actions in triggering a load shedding (i.e., both actions can successfully trigger load shedding) obtained from the Simulink simulation results [15].

The defender's action set A_d consists of the following actions, $A_d = \{\tau_1 = 11, \tau_2 = 32\}$, which represent the two thresholds that the defender employs to detect an injection attack. These two values correspond to the number of zero-crossings per 0.5 seconds for the normal case and the case of $k = 1.1$ respectively based on Matlab/Simulink simulation. We omit the unnecessary details of action selections and interested readers may refer to [15]. Different actions of the attacker result in different optimal actions for the defender to detect the attack. For example, intuitively, if the attacker's action is $k = 1.1$, the best action for the defender would be 11 to avoid high false-negative; on the other hand, if the attacker chooses the action of -0.8, it is better for the defender to switch to the action of 32 to avoid high false-positive. Thus the optimal strategy of the defender closely depends on the strategy taken by the attacker. This kind of dependency can be observed from the payoff matrix under state s_1 we give next.

From the Simulink simulation results, the average payoff of the attacker/defender under state s and joint action (a_a, a_d) is calculated as the expected amount of load shedding by executing (a_a, a_d) under state s, which is given as follows,

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

Note that the defender's average reward (cost) includes the expected cost of meter switching process under false-positive, which is equivalent to the switching cost per time times the false-positive probability. Since it is a zero-sum game, the attacker's payoffs are exactly the negation of those for the defender and are omitted here. Besides, by verifying the defender's payoff matrix under state s_1, we can easily observe that $R_d(s_1)(\tau_2, k_1) > R_d(s_1)(\tau_1, k_1)$ and $R_d(s_1)(\tau_1, k_2) > R_d(s_1)(\tau_2, k_2)$, which is consistent with the intuition we previously described.

The transition probabilities between states under each joint action are given as follows,

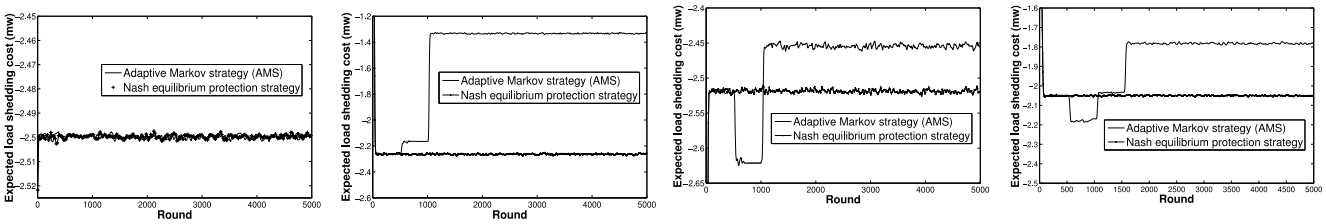
$$Pr(d_1, a_1) = \begin{vmatrix} 43/45 & 2/45 \\ 1/2 & 1/2 \end{vmatrix} Pr(d_2, a_1) = \begin{vmatrix} 0 & 1 \\ 1/47 & 46/47 \end{vmatrix}$$

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

Finally we remark that the system dynamics is continuous and the Markov game modeling here can be considered as the snapshot of the system dynamics when the defender and attacker make decisions.

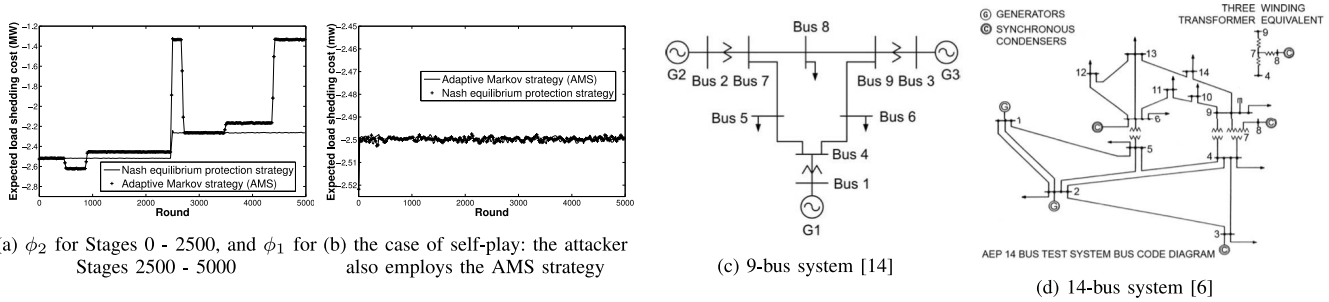
2) Simulation Results: To evaluate the performance of AMS, we compare its performance against different attacker's strategies with a NE defending strategy. In each scenario, we ran a simulation of playing the Markov game for 5000 rounds, and measured the average load shedding costs when the defender employs AMS and NE strategy respectively.

a) Performance against different stationary opponents: Figure 2a shows the average load shedding costs when the



(a) Nash equilibrium strategy (b) Strategy $\phi_1(s_1) = \phi_1(s_2) = a_1$ (c) strategy ϕ_2 of always choosing action a_2 (d) strategy ϕ_3 of choosing action a_1/a_2 randomly under each state

Fig. 2. The average load shedding cost when the attacker uses different attacking strategies.



(a) ϕ_2 for Stages 0 - 2500, and ϕ_1 for Stages 2500 - 5000 (b) the case of self-play: the attacker also employs the AMS strategy (c) 9-bus system [14] (d) 14-bus system [6]

Fig. 3. (a)-(b): The average load shedding cost when the attacker uses different attacking strategies; (c)-(d): 9-bus and 14-bus testbed distribution systems.

attacker employs the NE strategy and the defender chooses the NE and AMS as the defending strategy respectively. It can be observed that both strategies result in approximately the same amount of load shedding. Intuitively, after recognizing that the attacker is employing the NE strategy, AMS learns to choose its corresponding (optimal) NE strategy as the defending strategy. Also we can notice that no learning period is required for the AMS defending strategy, which is able to learn to use the NE strategy to defend immediately.

Figure 2b shows the expected cost of load shedding when the attacker chooses strategy ϕ_1 (i.e., always choosing action a_1), and the defender employs the NE and AMS as its defending strategy respectively. It shows that the system's average load shedding cost is significantly reduced when AMS is deployed. Intuitively speaking, when the AMS realizes that the attacker employs a strategy different from the NE strategy, it will compute the corresponding best-response strategy that exploits the attack pattern, and thus the load shedding cost can be minimized. Similarly, if the attacker uses strategy ϕ_2 where a_2 is always selected (Figure 2c), the AMS computes the corresponding optimal strategy that significantly reduces the load shedding cost than the NE strategy. Besides, we notice that there is a temporary drop-off around the 500th round in Figure 2c. This is due to the fact that the AMS strategy determines the best-response defending strategy based on which strategy it believes the attacker is currently employing. Thus at the beginning before the AMS can obtain an accurate estimation of the attacker's strategy, it may temporarily resort to a strategy which turns out to be less than optimal.

Figure 2d illustrates the dynamic change of the expected load shedding costs when the attacker uses another strategy ϕ_3 under which the two actions, a_1 and a_2 are randomly selected under each state. Similar to the results in Figure 2b, AMS initially results in a slightly higher cost, however, as its estimation of the attacker's strategy becomes more accurate and stabilized (around the 1500th round), it significantly outperforms the NE strategy thereafter since AMS is the best-response defending strategy against the attacker's strategy.

Finally we evaluate a scenario when the attacker may alternate between different strategies during the attack. Figure 3a illustrates the average load shedding cost for the case when the attacker first employs the strategy ϕ_2 and switches to another strategy ϕ_1 in the middle of the attack (around 2500 round). It can be observed that most of the times, the AMS can intelligently adjust its defending action and fully exploit the attacker's dynamic behaviors to greatly decrease the load shedding cost than the NE strategy.

b) Performance under self-play: In this section, we consider the case when the attacker is sufficiently intelligent to employ the same AMS strategy to launch the attack. We have theoretically proved that the players are guaranteed to converge to the Nash equilibrium under self-play. Here we empirically evaluate the performance of the AMS strategy when the attacker also adopts the AMS strategy comparing with the case of adopting a NE strategy to defend.

Figure 3b illustrates the average load shedding costs when the attacker employs the AMS strategy and the defender employs AMS and a NE strategy to defend respectively. It shows that AMS can achieve the same performance as the NE strategy. The reason can be explained as follows. When the attacker adopts AMS, the players always converge to Nash equilibrium no matter whether the defender employs AMS or a NE strategy. This indicates that a defender employing AMS can successfully avoid being exploited by an intelligent attacker, since the worse-case damage to the grid is the same as the damage when the attacker employs the NE strategy.

c) Influence of the action space: In this section, we investigate the influence of the action space on the learning

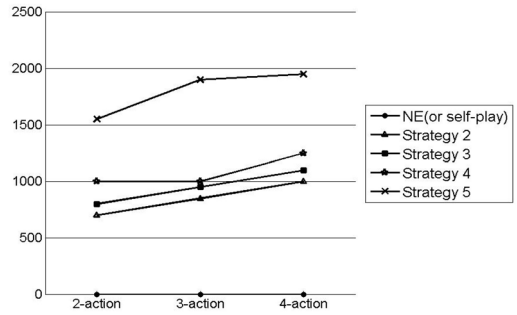


Fig. 4. Average Convergence Time (in Rounds).

TABLE II
AVERAGE CONVERGENCE TIME (IN ROUNDS)

Attacker's strategies	1-generator 4-bus distribution system	3-generator 4-bus distribution system	2-generator 14-bus distribution system
Strategy in Figure 2a	0	0	0
Strategy in Figure 2b	853	860	856
Strategy in Figure 2c	734	738	731
Strategy in Figure 2d	1576	1571	1578
Strategy in Figure 3a	976	971	978
Strategy in Figure 3b	0	0	0

performance of the AMS. We consider the following two cases with the number of actions for both players increased from two to three and four respectively.

The attacker's action sets are denoted by $A_a = \{k_1 = 1.1, k_2 = -0.8, k_3 = -1.2\}$ and $A'_a = \{k_1 = 1.1, k_2 = -0.8, k_3 = -1.2, k_4 = 1.2\}$ These values represents the false voltage values that the attacker can choose to modify the voltage and send back to the STATCOM, and are selected based on the Simulink simulation results. The defender's corresponding action sets are represented by $A_d = \{\tau_1 = 11, \tau_2 = 32, \tau_3 = 148\}$ and $A'_d = \{\tau_1 = 11, \tau_2 = 13, \tau_3 = 32, \tau_4 = 148\}$ which correspond to the set of thresholds the defender can employ to detect the attack, obtained from the Simulink simulation.

Figure 4 lists the average convergence times of the AMS against different types of attacker's strategies in rounds when the number of actions varies from two to four. For the attacker's strategies, strategy II - V are constructed as follows for all cases. First we randomly select two distinct actions $a1$ and $a2$, and then construct the strategies as follows: II) always choosing $a1$; III) always choosing a_2; IV) choosing each action randomly; V) choosing $a1$ under s_1 and $a2$ under s_2. For all cases, the unselected actions are chosen with a small probability to model the noise of the environment. From Figure 4, we can observe that the average convergence time is increased with the increase of the action space of the players. This is expected since more rounds of interaction experience is needed before obtaining a sufficiently accurate estimation of the opponent's strategy. Thus the average time (in rounds) required before convergence to the optimal best-response strategy is increased accordingly. For the cases in Figure 2a and Figure 3b, since the AMS agent always employs the precomputed NE strategy as the initial strategy, its behavior is always optimal (best response) starting from the beginning against opponents adopting either the corresponding NE strategy or the AMS. Therefore, the time needed before converging to the best response is always zero.

B. 9-Bus and 14-Bus Distribution Systems

In this section, we consider two more complex distribution systems: the 9-bus system (Figure 3c) [14] and the 14-bus system (Figure 3d) [6] to further evaluate the effectiveness and scalability of the AMS. For the 9-bus distribution system, there are three generators and three loads; for the 14-bus system, there are two generators and ten loads. For both systems, similar to the testbed system in Figure 1, we assume that there is one STATCOM connected to the system through Bus 8 to regulate the system's voltage level, while the attacker can compromise the merging unit and send false voltage data back to the STATCOM.

Similar to what we did in previous section, we abstract the system's state space into two states, $S = \{s_1, s_2\}$, representing the conditions when the system suffers from zero load shedding and certain amount of load shedding. Regarding the defender and attacker's actions, we first perform Simulink simulation to investigate whether load shedding happens and the number of times that $I - I_{ref}$ crossing zero for different actions of the attacker. Based on the Simulink results, we select two most stealth and effective actions as the attacker's action set for the 9-bus and 14-bus systems as follows.

The attacker's action set consists of the following actions,

$$A_a = \{k_1 = 2.1/1.7, k_2 = -1.3/-1.8\},$$

which denote the two false voltage values that the attacker may choose to inject into the STATCOM for the 9-bus and 14-bus systems respectively.

Given the attacker's action set, the defender's corresponding action set consists of the most effective thresholds to detect the stealthy attack from A_a, which is represented as $A_d = \{\tau_1 = 13/18, \tau_2 = 35/43\}$. These values are the thresholds the defender can use to detect injection attacks for the 9-bus and 14-bus systems respectively.

Similar to Section IV-A, we evaluated different cases where the attacker may employ different stationary strategies (including NE) or the same AMS. The simulation results for both 9-bus and 14-bus systems share similar patterns with the results for the 1-generator 4-bus distribution system in Section IV-A. We omit the unnecessary detailed results to save space and summarize the main results as follows.

- the AMS defender can achieve the same average payoff as NE strategy when the opponent adopts the corresponding NE strategy to attack;
- the AMS defender can achieve statistically significant higher average payoff than NE strategy when the opponent adopts non-NE strategies after convergence.

The average convergence times in rounds for both systems are listed in Table II. We also list the results for the 1-generator 4-bus distribution system in previous section again for comparison purpose. From Table II, we can see that there is no statistically significant difference among the average convergence times for these three distribution systems. This is expected since the convergence time only depends on the size

of the Markov game modeling the underlying system, which does not depends on the size and complexity of the system itself. To some degree, this observation suggests that through appropriate abstraction, the AMS defending strategy based on Markov game modeling is not only more effective comparing with employing NE strategy, but also applicable for complex distribution system in practice.

V. Conclusion

Game-theoretical modeling and analysis is an important paradigm to handle cyber security in smart-grid systems. The conventional approach of adopting the Nash equilibrium solution concept from game theory as the defending strategy might not be an optimal choice, due to a number of assumptions that may not be valid in practice, especially when a system is as complex as a smart grid. To this end, we proposed a novel adaptive strategy called AMS, which is theoretically proved to be rational and convergent. We performed extensive experimental evaluation on one important class of cyber attacks on power distribution systems—false voltage injection attack, and showed AMS's superior performance compared with the conventional NE strategy under a number of testbed systems.

As future work, we intend to study the applicability and effectiveness of AMS on other types of cyber attacks in smart-grid systems, and investigate more efficient techniques for further reduce the computational complexity of AMS to better suit larger scale smart-gird systems and cyber-physical systems in general. Besides, we assume that the transmission channel is perfect and It would be interesting to take into consideration the transmission noise as future work. Finally, another interesting direction to explore is considering the case of multiple attackers, who can launch either individual or coordinated attack simultaneously. It is worthwhile exploring how the AMS can be applied or extended for effectively handling these more complicated attacking scenarios or different application domains (e.g., cloud computing domain [13]).

References

[1] *SimPowerSystems Users Guide R2012a, Version 5.6*, Hydro-Québec and The MathWorks, Inc., Natick, MA, USA, 2012.

[2] (2014). *SimPowerSystems Documentation: D-STATCOM (Average Model)*. [Online]. Available: http://www.mathworks.com/help/physmod/sps/examples_v2/d-statcom-average-model.html

[3] G. Alsmeyer, "Chebyshev's inequality," in *International Encyclopedia of Statistical Science*. Heidelberg, Germany: Springer, 2011, pp. 239–240.

[4] M. H. Bowling and M. M. Veloso, "Convergence of gradient dynamics with a variable learning rate," in *Proc. ICML*, Williamstown, MA, USA, 2001, pp. 27–34.

[5] G. Brown, M. Carlyle, J. Salmerón, and K. Wood, "Defending critical infrastructure," *Interfaces*, vol. 36, no. 6, pp. 530–544, 2006.

[6] R. Christie. *Power System Test Archive*. [Online]. Available: http://www2.ee.washington.edu/research/pstca/pf14/pg_tca14bus.htm

[7] V. Conitzer and T. Sandholm, "AWESOME: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents," *Mach. Learn.*, vol. 67, nos. 1–2, pp. 23–43, 2007.

[8] S. C. Eisenstat, H. C. Elman, and M. H. Schultz, "Variational iterative methods for nonsymmetric systems of linear equations," *SIAM J. Numer. Anal.*, vol. 20, no. 2, pp. 345–357, 1983.

[9] M. Elidrisi, N. Johnson, and M. Gini, "Fast learning against adaptive adversarial opponents," in *Proc. AAMAS*, Valencia, Spain, 2012, pp. 1–8.

[10] X. Fan and G. Gong, "Security challenges in smart-grid metering and control systems," *Technol. Innov. Manag. Rev.*, vol. 3, no. 7, pp. 42–49, 2013.

[11] J. P. Farwell and R. Rohozinski, "Stuxnet and the future of cyber war," *Survival*, vol. 53, no. 1, pp. 23–40, 2011.

[12] J. Hao and H.-F. Leung, "Introducing decision entrustment mechanism into repeated bilateral agent interactions to achieve social optimality," *Auton. Agents Multi Agent Syst.*, vol. 29, no. 4, pp. 658–682, 2015.

[13] J. Li *et al.*, "Online optimization for scheduling preemptable tasks on IaaS cloud systems," *J. Parallel Distrib. Comput.*, vol. 72, no. 5, pp. 666–677, 2012.

[14] P. Kundur, N. J. Balu, and M. G. Lauby, *Power System Stability and Control*, vol. 7. New York, NY, USA: McGraw-Hill, 1994.

[15] Y. W. Law, T. Alpcan, and M. Palaniswami, "Security games for voltage control in smart grid," in *Proc. ICACCCT*, Ramanathapuram, India, 2012, pp. 212–219.

[16] D. Lefebvre, S. Bernard, and T. Van Cutsem, "Undervoltage load shedding scheme for the hydro-québec system," in *Proc. IEEE Power Eng. Soc. Gen. Meeting*, Denver, CO, USA, 2004, pp. 1619–1624.

[17] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proc. ICML*, New Brunswick, NJ, USA, 1994, pp. 157–163.

[18] J. Estelle *et al.*, "Strategic interactions in a supply chain game," *Comput. Intell.*, vol. 21, no. 1, pp. 1–26, 2005.

[19] C. Y. T. Ma, D. K. Y. Yau, and N. S. V. Rao, "Scalable solutions of Markov games for smart-grid infrastructure protection," *IEEE Trans. Smart Grid*, vol. 4, no. 1, pp. 47–55, Mar. 2013.

[20] C. Y. T. Ma, D. K. Y. Yau, X. Lou, and N. S. V. Rao, "Markov game analysis for attack-defense of power networks under possible misinformation," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 1676–1686, May 2013.

[21] M. Bowling and M. Veloso, "Rational and convergent learning in stochastic games," in *Proc. IJCAI*, vol. 2. Seattle, WA, USA, 2001, pp. 1021–1026.

[22] R. Powers, Y. Shoham, and T. Vu, "A general criterion and an algorithmic framework for learning in multi-agent systems," *Mach. Learn.*, vol. 67, nos. 1–2, pp. 45–76, 2007.

[23] W. Saad, Z. Han, V. Poor, and T. Basar, "Game-theoretic methods for the smart grid: An overview of microgrid systems demand-side management, and smart grid communications," *IEEE Signal Process. Mag.*, vol. 29, no. 5, pp. 86–105, Sep. 2012.

[24] J. Salmeron, K. Wood, and R. Baldick, "Analysis of electric grid security under terrorist threat," *IEEE Trans. Power Syst.*, vol. 19, no. 2, pp. 905–912, May 2004.

[25] O. Sigaud and O. Buffet, *Markov Decision Processes in Artificial Intelligence*. London, U.K.: Wiley, 2013.

[26] W. Wang and Z. Lu, "Cyber security in the smart grid: Survey and challenges," *Comput. Netw.*, vol. 57, no. 5, pp. 1344–1371, 2013.

[27] X. Li *et al.*, "Socially-aware multiagent learning towards socially optimal outcomes," in *Proc. 22nd Eur. Conf. Artif. Intell. (ECAI)*, The Hague, The Netherlands, 2016, pp. 533–541.

Authors' photographs and biographies not available at the time of publication.