Detecting Stealthy False Data Injection Attacks in State of Charge Estimation Using Sensor Encoding

Rodrigo D. Trevizan
Energy Storage Technology & Systems
Sandia National Laboratories
Albuquerque, NM, USA
rdtrevi@sandia.gov

Victoria A. O'Brien
Electric Grid Security & Communications
Sandia National Laboratories
Albuquerque, NM, USA
vaobrie@sandia.gov

Vittal S. Rao
Electrical and Computer Eng. Dept.
Texas Tech University
Lubbock, TX, USA
vittal.rao@ttu.edu

***Abstract*—This paper introduces a method for detecting stealthy false data injection attacks on the sensors of state of charge estimation algorithms used in battery management systems (BMSs). This method is based on sensor encoding, which is the active modification of sensor data streams. This method implements low-cost verification of the integrity of measurement data, allowing for the detection of stealthy additive attack vectors. It is considered that these attacks are crafted by malicious actors with knowledge of system models and who are capable of tampering with any number of measurements. The solution involves encoding all vulnerable measurements. The effectiveness of the method is demonstrated by simulations, where a stealthy attack on an encoded measurement vector captured by a BMS generates large residuals that trigger a chi-squared anomaly detector. Within the context of a defense-in-depth strategy, this method can be combined with other cybersecurity controls, such as encryption of data-in-transit, to equip cyberphysical systems with an additional line of defense against cyberattacks.**

***Index Terms*—bad data detection, cyberphysical security, false data injection attack, sensor encoding, state of charge estimation.**

I. Introduction

The modernization of the electric power grid has driven the adoption of devices equipped with advanced data processing and communications capabilities. For instance, battery energy storage systems (BESS) are composed of several programmable electronic devices that control and protect battery cells [1]. This combination of information technology and industrial control systems (ICSs) that composes the smart grid requires establishing a strong cybersecurity posture. This need has been acknowledged by North America's power system regulators, who have developed mandatory cybersecurity standards for bulk power systems [2]. Recent cyberattacks targeting power grids [3] and other critical infrastructure are reminders of the importance of cyberphysical security.

Among the power grid applications identified as vulnerable to cyberattacks are power system state estimators (PSSEs) employed for grid monitoring [4]. It has been conjectured that attacks on the data integrity of power grid sensors can severely impair the situational awareness of power system operators. One class of such attacks, named false data injection attacks (FDIAs), has been extensively investigated in the technical literature on cyberphysical security of ICSs. FDIAs involve the active tampering of data streams, including control signals and measurements. Further, if these data modifications cannot be detected by traditional bad data detection (BDD) methods, the FDIA is classified as stealthy.

Since [5] introduced stealthy FDIA for static PSSE, extensions to nonlinear models [6] dynamic linear systems [7], and dynamic PSSE [8] have been proposed. Proposed strategies to detect such attacks include model-based advanced detectors [9], data-driven machine learning approaches [10], and moving-target defense techniques [11].

More recently, measurement encoding approaches have been proposed to detect FDIAs in ICS. Their goal is to impair the capacity of the threat actor to create a stealthy attack sequence by preemptively modifying the values of sensor readings. The work [12] introduced a sensor encoding strategy to maximize the residuals of a previously the proposed stealthy attack sequence [7]. A sensor encoding technique for detection of stealthy FDIA in static PSSE was also presented in [13].

Beyond PSSEs, state of charge (SoC) estimation within BESS is paramount for their operation and safety [1]. SoC estimation is typically implemented in battery management systems (BMSs), which are embedded systems that measure voltage, current, and temperature from battery cells and perform several other important protection and safety functions. Some BMSs employ state estimation algorithms that could be hardened with BDD capabilities. It has been shown in [14] that nonlinear control systems are vulnerable to stealthy FDIA sequences in certain conditions. Methods for detecting small FDIAs in BMSs have shown promising results in simulations [15], [16]. These methods, however, might not be able to detect a stealthy FDIA targeting the sensor streams of BMSs.

This work was supported by the U.S. Department of Energy, Office Electricity, Energy Storage program. This article has been authored by an employee of National Technology & Engineering Solutions of Sandia, LLC under Contract No. DE-NA0003525 with the U.S. Department of Energy (DOE). The employee owns all right, title and interest in and to the article and is solely responsible for its contents. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this article or allow others to do so, for United States Government purposes. The DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan https://www.energy.gov/downloads/doe-public-access-plan. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. SAND2024-02614C.

In comparison, cryptographic methods may place large overheads on ICSs since data-in-transit encryption requires significant computation, and data integrity verification (e.g., hashes, message authentication codes) demands transmitting additional data. However, embedded field devices, such as BMSs, operate in real time, implement safety-critical functions, have limited memory and processing capabilities, manage a large number of sensors, have limited communications bandwidth, and have low power consumption requirements. Sensor encoding, on the other hand, can implement data integrity verification with a marginal increase in computational burden. Limitations of sensor encoding include weaker security than cryptographic hashing, and need of accurate nonlinear state estimation [13]. Finally, differentiating between an attack and a system fault remains an open problem.

This paper extends [13], [16] by applying a sensor encoding technique for detection of stealthy FDIAs targeting BMS SoC estimation. In summary, the contributions of this paper are:

1) The development of a stealthy attack sequence targeting SoC estimation within a BMS considering that the attacker has knowledge of system models;
2) A sensor encoding approach that allows attack detection by impairing the attacker's ability to craft stealthy FDIA.

The remainder of the paper is organized as follows. Section II describes the problem. Section III presents the rule to generate a stealthy FDIA sequence. Section IV introduces the proposed sensor encoding-based solution. The application of the solution to a simulated problem is presented in Section V. Finally, Section VI presents the conclusions.

II. Problem Description

Let's consider the system shown in Fig. 1, where we have a defender and an attacker. The defender's state estimator collects sensor data (input and output) from BMSs and has good battery cell physical dynamic models and parameters. The attacker gets hold of these dynamic models. The goal of the attacker is to introduce a bias to the state estimates produced by the defender while avoiding attack detection by using stealthy FDIAs. The goal of the defender is to detect the stealthy FDIA injected into the measurements. Attack remediation is out of the scope.

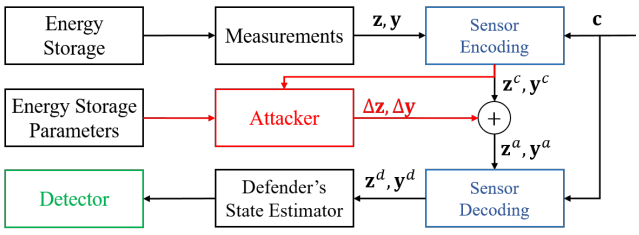


Fig. 1. Problem depiction. The attacker has access to system parameters and encoded measurements and they can change any measurement.

A. State of Charge Estimation

The problem of SoC estimation closely follows the framework described in [16]. As shown in Fig. 2, we consider the case where both inputs (e.g., battery stack current) as well as outputs of the dynamic system (e.g., battery cell and stack voltages) are corrupted by Gaussian uncorrelated noise.

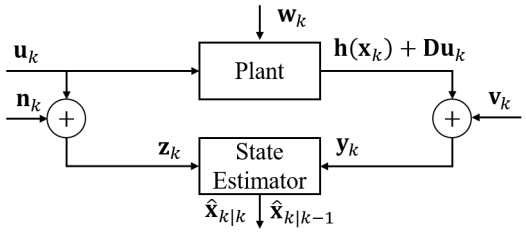


Fig. 2. The plant input $\mathbf{u}_k$ is corrupted by the noise $\mathbf{n}_k$ when it is measured, so the state estimator observes a noisy input signal $\mathbf{z}_k$ [16].

We consider that battery dynamics are correctly represented by equivalent circuit models and charge reservoir models [16], [17]. As a result, the state transition equations are linear but the SoC-dependent output functions are nonlinear. The discrete nonlinear dynamic system with noisy inputs is represented by

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k, \tag{1a}$$

$$\mathbf{y}_k = \mathbf{h}\left(\mathbf{x}_k\right) + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k, \tag{1b}$$

$$\mathbf{z}_k = \mathbf{u}_k + \mathbf{n}_k, \tag{1c}$$

where $\mathbf{y}_k \in \mathbb{R}^m$ is the measurement vector at time step k, $\mathbf{x}_k \in \mathbb{R}^n$ is the state vector, $\mathbf{u}_k \in \mathbb{R}^p$ is the input vector, $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the state matrix, $\mathbf{B} \in \mathbb{R}^{n\times p}$ is the input matrix, $\mathbf{h}(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^m$ is a state-dependent nonlinear function, $\mathbf{D} \in \mathbb{R}^{m\times p}$ is the throughput matrix, $\mathbf{w}_k \in \mathbb{R}^n$, is the process noise vector with covariance $\mathbf{Q}$, $\mathbf{v}_k \in \mathbb{R}^m$, is the vector of measurement errors with covariance $\mathbf{R}$, $\mathbf{n}_k \in \mathbb{R}^p$, is the vector of input measurement noise with covariance $\mathbf{N}$, and $\mathbf{z}_k \in \mathbb{R}^p$, is the vector of noisy inputs. All noise vectors are Gaussian, zero-mean, and uncorrelated.

B. Input Noise-Aware Extended Kalman Filter

The states of this nonlinear system are estimated using an Input Noise-Aware Extended Kalman Filter (INAEKF) [16]:

$$\hat{\mathbf{y}}_{k|k-1} = \mathbf{h}\left(\hat{\mathbf{x}}_{k|k-1}\right) + \mathbf{D}\mathbf{z}_k, \tag{2a}$$

$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}, \tag{2b}$$

$$\mathbf{C}_{k|k-1} = \left.\frac{\partial \mathbf{h}\left(\mathbf{x}\right)}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1}}, \mathbf{C}_{k|k} = \left.\frac{\partial \mathbf{h}\left(\mathbf{x}\right)}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k}}, \tag{2c}$$

$$\Sigma_k = \mathbf{C}_{k|k-1}\mathbf{P}_{k|k-1}\mathbf{C}_{k|k-1}^{\mathsf{T}} + \mathbf{D}\mathbf{N}\mathbf{D}^{\mathsf{T}} + \mathbf{R}, \tag{2d}$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{C}_{k|k-1}^{\mathsf{T}}\Sigma_k^{-1}, \tag{2e}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\tilde{\mathbf{y}}_k, \tag{2f}$$

$$\mathbf{U}_k = \mathbf{N}\mathbf{D}^{\mathsf{T}}\Sigma_k^{-1}, \tag{2g}$$

$$\hat{\mathbf{u}}_k = \mathbf{z}_k + \mathbf{U}_k\tilde{\mathbf{y}}_k, \tag{2h}$$

$$\hat{\mathbf{y}}_{k|k} = \mathbf{h}\left(\hat{\mathbf{x}}_{k|k}\right) + \mathbf{D}\hat{\mathbf{u}}_k, \tag{2i}$$

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}\hat{\mathbf{x}}_{k|k} + \mathbf{B}\hat{\mathbf{u}}_k, \tag{2j}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1}\left(\mathbf{I} - \mathbf{C}_{k|k-1}^{\mathsf{T}}\mathbf{K}_k^{\mathsf{T}}\right), \tag{2k}$$

$$\begin{aligned}\mathbf{P}_{k+1|k} = \mathbf{A}\mathbf{P}_{k|k}\mathbf{A}^{\mathsf{T}} + \mathbf{B}\left(\mathbf{N}_k - \mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{U}_k^{\mathsf{T}}\right)\mathbf{B}^{\mathsf{T}} + \mathbf{Q} \\ - \mathbf{B}\mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{K}_k^{\mathsf{T}}\mathbf{A}^{\mathbf{T}} - \mathbf{A}\mathbf{K}_k\boldsymbol{\Sigma}_k\mathbf{U}_k^{\mathbf{T}}\mathbf{B}^{\mathsf{T}},\end{aligned} \tag{2l}$$

where $\hat{\mathbf{x}}_{k|k}$ is the corrected state estimate, $\hat{\mathbf{x}}_{k|k-1}$ is the predicted state estimate, $\mathbf{P}_{k|k}$ is the *a posteriori* state error covariance, $\mathbf{P}_{k|k-1}$ is the predicted state error covariance, $\hat{\mathbf{y}}_{k|k}$ is the *a posteriori* output estimate, $\tilde{\mathbf{y}}_{k|k-1}$ is the predicted output estimate, $\tilde{\mathbf{y}}_k$ is the vector of innovations, Σ_k is the covariance of the innovations, $\hat{\mathbf{u}}_k$ is the estimated input vector, $\mathbf{C}_{k|k}$ and $\mathbf{C}_{k|k-1} \in \mathbb{R}^{m\times n}$ are the partial derivatives of $\mathbf{h}(\mathbf{x})$ with respect to $\mathbf{x}$ at $\hat{\mathbf{x}}_{k|k}$ and $\hat{\mathbf{x}}_{k|k-1}$ respectively, $\mathbf{K}_k$ is the Kalman gain, and $\mathbf{U}_k$ is the input estimation gain matrix.

C. FDIA Detection

In this paper, we chose the χ^2 (chi-squared) test to the innovation (2b) for FDIA detection. This well-known statistical test for BDD quantifies how well the data fits the system model. At each time step, the defender applies this test to their datasets to detect FDIA. To obtain a χ^2 distribution, the INAEKF innovations are normalized by their covariance (3).

$$J_{\tilde{\mathbf{y}}_k} = \tilde{\mathbf{y}}_k^{\mathsf{T}}\Sigma_k^{-1}\tilde{\mathbf{y}}_k \tag{3}$$

If all assumptions regarding the INAEKF hold, $J_{\tilde{\mathbf{y}}_k}$ will follow a χ^2 distribution with $\nu = m-n$ degrees of freedom. A level of significance α is selected such that $P(J_{\tilde{\mathbf{y}}_k} \geq C) = \alpha$, i.e., the false positive rate for a threshold $C = \chi^2_{\nu,\alpha}$ is α. Consequently, if the number of data points flagged by the chi-squared detector is significantly larger than the expected false positive rate, the attack is detected.

III. Stealthy FDIA Sequence

In this paper, it is assumed that goal of the FDIA is to modify measurements to introduce a bias in the state estimation vector such that a state-dependent protection or safety feature of a BMS fails to protect the system appropriately. It is conjectured that such a sequence can be crafted to cause damage to the battery cells by circumventing protection algorithms, or by incorrectly triggering a safety mechanism to cause loss of availability of the BESS to the power grid.

To achieve that, attack vectors $\Delta\mathbf{z}_k$ and $\Delta\mathbf{y}_k$ are devised to bias the measurements $\mathbf{z}_k^a = \mathbf{z}_k + \Delta\mathbf{z}_k$ and $\mathbf{y}_k^a = \mathbf{y}_k + \Delta\mathbf{y}_k$ used as inputs to the defender's INAEKF, where ^{a} indicates attacked variables. The strategy to develop a stealthy FDIA sequence is to craft attack vectors $\Delta\mathbf{z}_k$ and $\Delta\mathbf{y}_k$ such that the attacked and unattacked innovation vectors are the same, i.e., $\tilde{\mathbf{y}}_k^a = \tilde{\mathbf{y}}_k$. In this way, the number of attacked samples flagged by the BDD will be very close to the expected false positive rate and no attack is detected. The vector is calculated by

$$\Delta\mathbf{y}_k = \mathbf{h}\left(\hat{\mathbf{x}}_{k|k-1}^a\right) - \mathbf{h}\left(\hat{\mathbf{x}}_{k|k-1}\right) + \mathbf{D}\Delta\mathbf{z}_k, \tag{4}$$

where $\Delta\mathbf{z}_k$ is chosen freely by the attacker.

Proof: considering that an attack starts at k, so $\hat{\mathbf{x}}_{k|k-1}^a = \hat{\mathbf{x}}_{k|k-1}$ and $\Delta\mathbf{z}_l = \mathbf{0}$, $\Delta\mathbf{y}_l = \mathbf{0}, \forall l \leq k-1$, applying (4) yields

$$\begin{aligned}\tilde{\mathbf{y}}_k^a &= \mathbf{y}_k^a - \hat{\mathbf{y}}_{k|k-1}^a = \mathbf{y}_k + \Delta\mathbf{y}_k - \left[\mathbf{h}\left(\hat{\mathbf{x}}_{k|k-1}\right)\right. \\ &\left.+\mathbf{D}\left(\mathbf{z}_k + \Delta\mathbf{z}_k\right)\right] = \mathbf{y}_k + \mathbf{D}\Delta\mathbf{z}_k - \hat{\mathbf{y}}_{k|k-1} - \mathbf{D}\Delta\mathbf{z}_k = \tilde{\mathbf{y}}_k\end{aligned}$$

for any $l \geq k+1$ we have the following

$$\begin{aligned}\tilde{\mathbf{y}}_l^a &= \mathbf{y}_l^a - \hat{\mathbf{y}}_{l|l-1}^a = \mathbf{y}_l + \Delta\mathbf{y}_l - \left[\mathbf{h}\left(\mathbf{x}_{l|l-1}^a\right) + \mathbf{D}\left(\mathbf{z}_l + \Delta\mathbf{z}_l\right)\right] \\ &= \mathbf{y}_l + \mathbf{h}\left(\hat{\mathbf{x}}_{l|l-1}^a\right) - \mathbf{h}\left(\hat{\mathbf{x}}_{l|l-1}\right) + \mathbf{D}\Delta\mathbf{z}_l - \left[\mathbf{h}\left(\mathbf{x}_{l|l-1}^a\right)\right. \\ &\left.+\mathbf{D}\left(\mathbf{z}_l + \Delta\mathbf{z}_l\right)\right] = \mathbf{y}_l - \mathbf{h}\left(\hat{\mathbf{x}}_{l|l-1}\right) - \mathbf{D}\mathbf{z}_l = \tilde{\mathbf{y}}_l\end{aligned}$$

The result above indicates that, in order to generate a stealthy FDIA sequence, the attacker needs both a state estimator with access to the untampered data set and another state estimator processing the attacked data set to generate (4).

IV. Sensor Encoding

The encoding approach consists of modifying the measurements such that the sensor data cannot be interpreted by the attacker. If this goal is achieved, then the term $\mathbf{h}\left(\hat{\mathbf{x}}_{k|k-1}\right)$ cannot be accurately estimated and a stealthy attack will not be generated, eventually causing innovations large enough to trigger the defender's anomaly detector (3). Given the secret encoding vector sequences $\mathbf{c}_k^z$ and $\mathbf{c}_k^y$, the encoding function should produce the encoded vectors $\mathbf{z}_k^c$ and $\mathbf{y}_k^c$ that the attacker will have access to. Conversely, with knowledge of the same keys, the decoding function recovers the original measurement vectors $\mathbf{z}_k^d = \mathbf{z}_k$ and $\mathbf{y}_k^d = \mathbf{y}_k$. Distributed encoding of data with measuring spans $\mathbf{a}^z$ and $\mathbf{a}^y$ is implemented with a pair of encoding (5a) and decoding (5b) functions.

$$\mathbf{z}_k^c = mod\left(\mathbf{z}_k + \mathbf{c}_k^z, \mathbf{a}^z\right), \mathbf{y}_k^c = mod\left(\mathbf{y}_k + \mathbf{c}_k^y, \mathbf{a}^y\right), \tag{5a}$$

$$\mathbf{z}_k^d = mod\left(\mathbf{z}_k^c - \mathbf{c}_k^z, \mathbf{a}^z\right), \mathbf{y}_k^d = mod\left(\mathbf{y}_k^c - \mathbf{c}_k^y, \mathbf{a}^y\right), \tag{5b}$$

where mod is the modulo function that returns the remainder of division of the first by the second argument.

A simple implementation can use pseudorandom number (PRN) generators to obtain $\mathbf{c}_k^z$ and $\mathbf{c}_k^y$. If the sensor and state estimator PRN generators have access to the same seed and algorithm parameters, they would be able to generate the same sequence of keys and the encoding and decoding process would be effective for as long as synchronization is kept. It is important to note that there are many types of algorithms to generate sequences of PRNs, each one with different characteristics such as statistical properties or their suitability for cryptographic applications, for example. To maximize the entropy, a uniform distribution is used.

Another point to consider is the detectability of the encoding by the attacker. If size of noise added by PRNs is small, the attacker might not notice the encoding method and may not devise a solution to bypass it. However, a small encoding vector might not be sufficient to create a large state bias so that a stealthy FDIA can still be generated by the attacker. On the other hand a large encoding vector will be easily detected by the attacker, but it will generate large state estimation errors in the attacker's state estimation, which is critical for stealthy FDIA detection using this encoding scheme.

V. Case Study

To evaluate the effectiveness of the stealthy FDIA and the sensor encoding approaches, we utilize a simulation where we have a battery system composed of three series-connected batteries with cell voltage, stack voltage, and stack current measurements. The cells are simulated using a second-order equivalent circuit model, and a third-order polynomial represents the SoC-dependent open-circuit voltage. More details on the simulation setup are described in [16]. The PRN keys are generated using the same algorithms and seeds for encoding and decoding, and they follow an uniform distribution that spans the entire range of measurements (0 to 5 V for cell voltages, 0 to 15 V for cell stack voltage, and -20 to 20 A for stack current). The false positive rate α is set to 0.01%. Simulation time is 8100 s and the estimator sampling time is 0.1 s. The battery system is cycled with constant current in the simulation. Both defender and attacker apply the INAEKF to track the systems' states and generate FDIA.

A. Stealthy FDIA: No Encoded Measurements

Let's suppose the attacker injects a current bias of -1 A. Then, the input attack vector sequence $\Delta\mathbf{z}_k$ and the corresponding output attack vector sequence $\Delta\mathbf{y}_k$ would look as shown in Figs. 3a and 3b, respectively, for an attack starting at 1,200 s. The output vector data correspond to voltages of cells 1, 2, 3, and the battery stack (sum of all voltages), respectively.

When the defender's state estimator ingests the attacked data streams, the state estimates diverge from the true states of the battery cells, as shown in Fig. 3c. More specifically, on the top plot of Fig. 3c, one can notice that before the attack inception, the SoC plot of estimated states (red $SoC[k|k]$ and yellow $SoC[k|k-1]$ lines) and the true SoC of cell one (green line) are superimposed, meaning that the INAEKF can track accurately the states of the system. Once the attacker starts to inject the stealthy FDIA sequence, the plots of estimated and true states of the system separate due to a bias introduced by the attacker. In spite of the incorrect state estimates, the defender's error detector only detects a few anomalous data points (0.0014%) after the filter settles (about 800 s), which is in line with the expected false positive data rate predicted by the theory, as shown in Fig. 3d.

B. Detecting the Stealthy FDIA with Sensor Encoding

By applying the encoding method described in Section IV we obtain the input and output encoding sequences shown in Fig. 4a and Fig. 4b, respectively. Those signals appear random and are dominated by the encoding signal with uniform distribution ranging between the maximum and minimum range of each sensor. Then, for the same input attack vector sequence shown in Fig. 3a, the corresponding output attack sequence is shown in Fig. 5a. The states estimated by the attacker's state estimator are depicted in Fig. 5b, where it is clear that the attacker loses track of the true system states.

The decoded data feeds the defender's state estimator, as shown in Fig. 6a. The defender's BDD scheme can now detect that the dataset is corrupted by bad data, as shown in Fig. 6b.

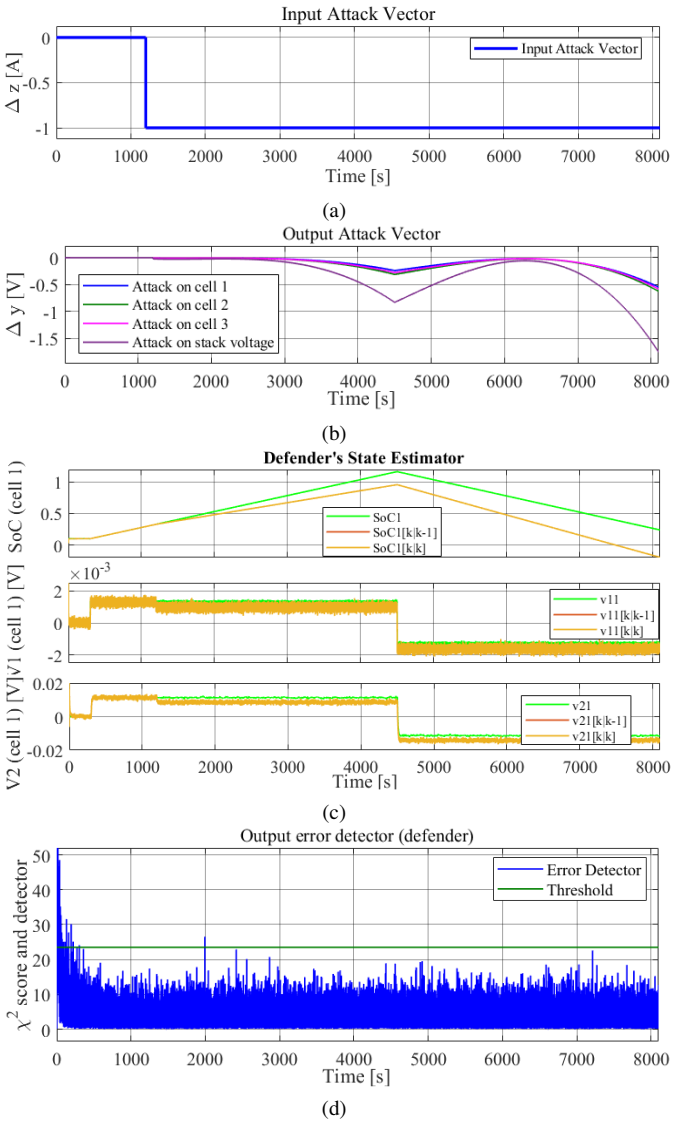
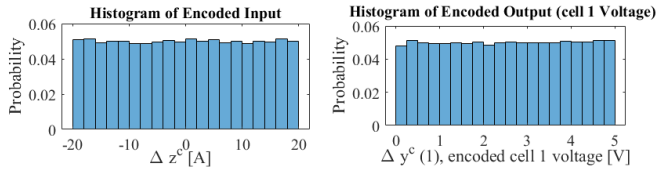


Fig. 3. Stealthy attack sequences in (a) system input and (b) system output. Those impair the capacity of the defender to track the systems' states (c) while not being captured by the BDD (d).



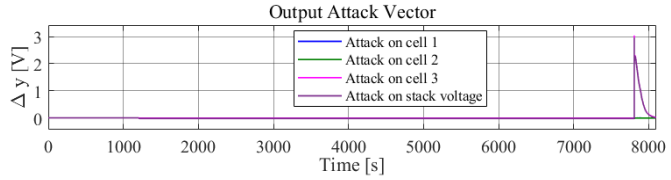
(a) (b)

Fig. 4. Histograms of encoded (a) system input, (b) system output signals.

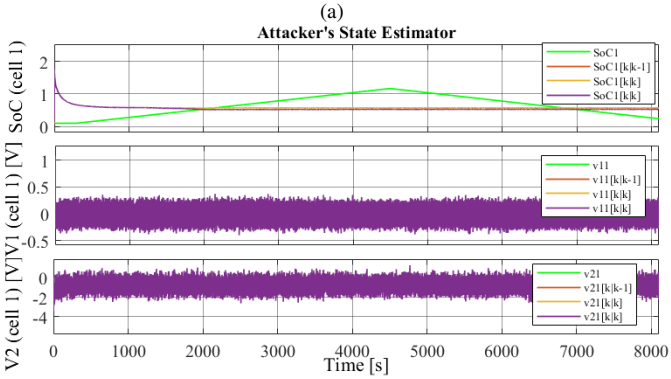
About 2.48% of data points are flagged as anomalous after the first 800 seconds of the simulation considering encoded measurements, which is much higher than the expected false positive rate of 0.01%, so the attack is detected.

VI. Conclusion

In this paper, we proposed a protocol for launching stealthy FDIAs on nonlinear systems and a sensor encoding scheme to detect stealthy attacks. The simulations have shown that

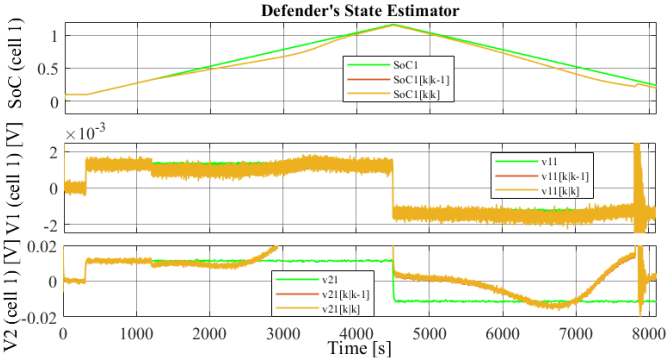


(a)

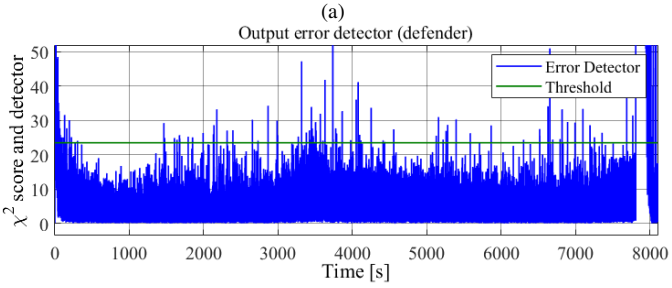


(b)

Fig. 5. Plots of (a) output attack vector, and (b) the attacker's state estimates when encoded data is used by the attacker.



(a)



(b)

Fig. 6. Defender's state estimates (a) and error detector (b) with encoding.

both the attack sequence and the sensor encoding are effective. The generation of stealthy attack requires system knowledge, access to all battery data stream, and the ability to run two state estimators simultaneously to generate stealthy attack sequences in real time. The encoding approach relies on the generation of PRNs. Detection is not instantaneous and requires a significant deviation between attacker state estimate and system state, which is dependent oh how the battery is being cycled. This cost-effective approach could be implemented in BMSs performing SoC estimation with little computational overhead, no increase in communication throughput, and no additional hardware. If needed, it could be combined with more security features, such as data encryption.

Future work will include practical considerations on how the sequences of PRNs are generated and how to synchronize them among all sensors, incorporate constraints in the number of encoded measurements, and apply other detection algorithms like CUSUM [16]. Further, we plan to extend the encoding algorithm for other state estimators and develop analytical proofs for stealthy FDIA detectability.

Acknowledgment

The authors thank Dr. Imre Gyuk, from the Energy Storage Program, for his continued support, and Dr. Hyungjin Choi for his technical review and insightful contributions to the work.

References

[1] R. D. Trevizan, J. Obert, V. De Angelis, T. A. Nguyen, V. S. Rao, and B. R. Chalamala, "Cyberphysical security of grid battery energy storage systems," *IEEE Access*, vol. 10, pp. 59 675–59 722, 2022.

[2] "CIP-002-5.1a—Cyber Security —BES Cyber System Categorization," North American Electric Reliability Corporation, Standard, Dec 2016.

[3] A. Greenberg, "Crash override malware took down Ukraine's power grid last december," Jun 2017. [Online]. Available: https://www.wired.com/story/crash-override-malware/

[4] K. Chatterjee, V. Padmini, and S. A. Khaparde, "Review of cyber attacks on power system operations," in *2017 IEEE Region 10 Symp. (TENSYMP)*, July 2017, pp. 1–6.

[5] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proc. of the 16th ACM Conf. on Computer and Comm. Security*, ser. CCS '09, 2009, pp. 21–32.

[6] M. A. Rahman and H. Mohsenian-Rad, "False data injection attacks against nonlinear state estimation in smart power grids," in *2013 IEEE Power Energy Society General Meeting*, 2013, pp. 1–5.

[7] C. Kwon, W. Liu, and I. Hwang, "Security analysis for cyber-physical systems against stealthy deception attacks," in *2013 American Control Conf.*, June 2013, pp. 3344–3349.

[8] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using kalman filter," *IEEE Trans. Control. Netw. Syst.*, vol. 1, no. 4, pp. 370–379, Dec. 2014.

[9] R. D. Trevizan, C. Ruben, K. Nagaraj, L. L. Ibukun, A. C. Starke, A. S. Bretas, J. McNair, and A. Zare, "Data-driven physics-based solution for false data injection diagnosis in smart grids," in *2019 IEEE Power Energy Society General Meeting (PESGM)*, 2019, pp. 1–5.

[10] S. A. Foroutan and F. R. Salmasi, "Detection of false data injection attacks against state estimation in smart grids based on a mixture gaussian distribution learning method," *IET Cyber-Physical Systems: Theory Applications*, vol. 2, no. 4, pp. 161–171, 2017.

[11] Z. Liu, Y. Li, Q. Wang, and J. Li, "Moving target defense of FDIAs for battery energy storage systems in smart distribution networks," *J. Energy Storage*, vol. 72, p. 108652, 2023.

[12] F. Miao, Q. Zhu, M. Pajic, and G. J. Pappas, "Coding sensor outputs for injection attacks detection," in *53rd IEEE Conf. on Decision and Control*, Dec 2014, pp. 5776–5781.

[13] R. D. Trevizan and M. Reno, "Detection of false data injection attacks in power system state estimation using sensor encoding," in *IEEE Kansas Power & Energy Conf. (KPEC)*, 2022, pp. 1–6.

[14] A. Khazraei and M. Pajic, "Resiliency of nonlinear control systems to stealthy sensor attacks," in *IEEE 61st Conf. on Decision and Control (CDC)*, 2022, pp. 7109–7114.

[15] V. Obrien, V. Rao, and R. D. Trevizan, "Detection of false data injection attacks in ambient temperature-dependent battery stacks," in *IEEE Electr. Energy Storage Appl. & Tech. Conf. (EESAT)*, 2022, pp. 1–6.

[16] V. Obrien, V. S. Rao, and R. D. Trevizan, "Detection of false data injection attacks in battery stacks using input noise-aware nonlinear state estimation and cumulative sum algorithms," *IEEE Trans. Industry Appl.*, vol. 59, no. 6, pp. 7800–7812, 2023.

[17] D. M. Rosewater, D. A. Copp, T. A. Nguyen, R. H. Byrne, and S. Santoso, "Battery energy storage models for optimal control," *IEEE Access*, vol. 7, pp. 178 357–178 391, 2019.