Distributed Data Recovery Against False Data Injection Attacks in DC Microgrids

Zexuan Jin, Mengxiang Liu, Ruilong Deng, and Peng Cheng

***Abstract*—With the development of information and communications technology (ICT) in DC microgrids (DCmGs), the threat of false data injection attacks (FDIAs) is becoming more and more serious. However, the existing literature mainly focuses on the detection and identification of FDIAs in DCmGs, while the data recovery after the perception of FDIAs has never been thoroughly investigated yet. In this paper, we propose a distributed data recovery scheme to eliminate the adverse impact caused by FDIAs in DCmGs. Firstly, by observing the point of common coupling (PCC) voltage under FDIAs, the injected constant bias can be roughly estimated. In order to obtain the precise constant bias, the mean filter (MF) is adopted to handle the measurement noises and small oscillations. Then, the estimated precise constant bias is compensated for the communicated signal to eliminate the attack impact. Furthermore, our proposed data recovery scheme, which only needs local information, is fully distributed. Finally, the accuracy and effectiveness of the distributed data recovery scheme are evaluated through systematical hardware-in-the-loop (HIL) experiments.**

***Index Terms*—Data recovery scheme, false data injection attack, DC microgrid**

I. Introduction

Nowadays, the situation of power generation and distribution is changing rapidly with the development of distributed energy resources (DERs) such as photovoltaics and wind turbines [1]. The microgrid has been recognized as one of the most promising solutions to accommodate for DERs. The growing number of DC loads (e.g., laptops, mobile devices and LED lights), national advocacy and planning of green energy and the availability of efficient converters are making the DC microgrid (DCmG) more and more popular.

The basic control objectives of the DCmG are voltage balancing and current sharing [2], [3]. To satisfy the high scalability need, the distributed hierarchical control framework is widely adopted to achieve the two control objectives [4], where the primary and secondary controllers are deployed in each DER. Specifically, the primary controller regulates the local point of common coupling voltage (PCC), and the secondary controller sets the reference PCC voltage for the primary control layer with the information received from the neighboring DERs. The adoption of information communication technologies (ICTs) can greatly increase the control performance, but will also introduce the threat of cyber attacks, especially false data injection attacks (FDIAs). The FDIA could easily destroy the synchronization among DERs, make the voltage and current deviate from the expected ones, and even cause serious power outages. Hence, much attention has been paid to the investigation of the countermeasures against FDIAs in DCmGs.

Beg *et al.* [5] proposed a framework to detect possible FDIAs in cyber-physical DCmG by identifying a change in sets of inferred candidate invariants. Zhang *et al.* [6] evaluated the variation of FDIAs to microgrid, and defined the stable region of the microgrid. Li *et al.* [7] presented an active synchronous detection method to detect deception attacks on inverter controllers in microgrids without impeding system operations. Gallo *et al.* [8] presented a distributed monitoring scheme to provide attack-detection capabilities for linear large-scale systems, which relies on a Luenberger observer together with a bank of unknown-input observers (UIOs) at each subsystem, providing attack detection capabilities. Abhinav *et al.* [9] presented a resilient synchronization protocol to mitigate attacks on communication links and adverse effects of hijacking controllers. Sahoo *et al.* [10] proposed a novel cooperative vulnerability factor framework which can accurately identifies the stealth attack. Aluko *et al.* [11] investigated the vulnerability of frequency measurements to FDIAs in an isolated MG system. Zhang *et al.* [12] studied the distributed load sharing problem of the microgrids operating in autonomous mode under FDIAs.

However, most existing literature mainly focuses on the detection and identification against FDIAs, while the recovery strategy after the perception of FDIAs has not yet been investigated. Indeed, when the attack sources are located to a specific set of links or nodes, the adverse impact can be largely mitigated after isolating the abnormal links or nodes from the DCmG. Nevertheless, the isolation might also disconnect the communication network and thus make the control objectives unachievable any more. In this paper, we propose a distributed data recovery scheme, which works after the attack sources have been located, to totally eliminate the adverse impact caused by FDIAs. Finally, we evaluate the accuracy and effectiveness in systematical HIL experiments with the HIL emulator-based cyber security testbed for DCmGs [13]. The contributions of this paper are listed as follows:

1) We propose a distributed data recovery scheme against FDIAs in DCmGs, which requires only local information

This work was supported in part by the Natural Science Foundation of Zhejiang Province under Grant LZ21F020006, in part by the National Natural Science Foundation of China under Grant 61833015, 62073285, 62061130220, and in part by the Fundamental Research Funds for the Central Universities (226-2022-00120).

The authors are with State Key Lab. of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Hangzhou, China (e-mails: {jinzexuan, lmx329, dengruilong, lunarheart}@zju.edu.cn).

978-1-6654-3254-2/22/$31.00 ©2022 IEEE

and can eliminate the adverse impact in a timely manner after the perception of FDIAs.

2) In the data recovery scheme, the injected constant bias is roughly estimated by observing the PCC voltages under the FDIA. To obtain the precise constant bias, the mean filter (MF) is adopted to handle the measurement noises and small oscillations. Then, the estimated precise constant bias is added to the communicated signal to eliminate the attack impact.
3) Through systematical hardware-in-the-loop (HIL) experiments, the accuracy and effectiveness of the data recovery scheme is evaluated.

The remainder of this paper is organized as follows. Section II introduces the DCmG model, the UIO-based detector model, the attack model, and problems interest. Section III presents the data recovery scheme. Section IV demonstrates simulations and experiment results and Section V concludes this paper.

II. System Model and Problems of Interest

In this section, we introduce the DCmG model, the UIO-based detector model, the attack model, and our problems of interest.

A. DCmG Model

We consider the DCmG with N DERs, where each DER consists of the electrical and control parts as shown in Fig. 1. The electrical topology of the DCmG is represented by a weighted undirected graph $G_{el} = \{V, E_{el}, W_{el}\}$, where $V = \{1, 2, ..., n\}$ represents the DER nodes, $E_{el} \subseteq V \times V$ represents the edges of power lines and $W \in R^{|E|\times|E|}$ represents the weight matrix. The cyber topology is represented by a weighted undirected graph $G_c = \{V, E_c, W_c\}$, where E_c represents the edges of communication links and W_{el} represents the weight matrix. The set of neighbors of DER i is denoted by $\mathcal{N}_i^c$.

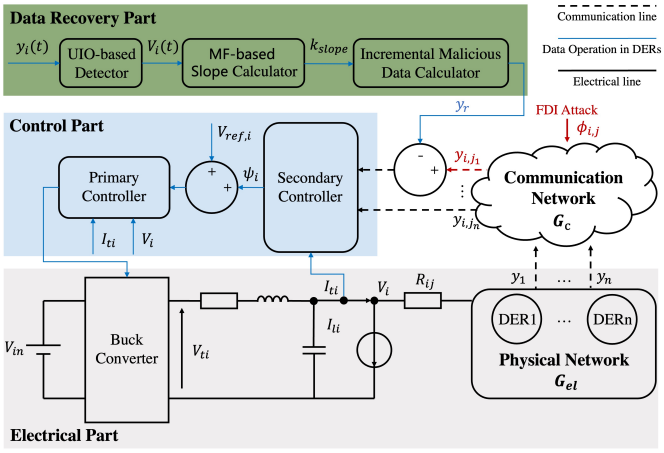


Fig. 1. This figure depicts the system diagram of the DCmG which is equipped with controller and data recovery. The FDIa data is injected into the communication network of DCMG.

The electrical part consists of a source, a buck converter, a RLC filter and a local load. Besides, each DER is connected to its neighbors with electrical lines. The V_{ti} represents the output voltage of buck converter, I_{li} is the local load current, I_{ti} and V_i is the output current and voltage of PCC, and R_{ij} is the resistance of electrical line. As illustrated in the control part of the DER, the hierarchical control framework, which consists of a primary controller and a secondary controller, is deployed. The dynamical model of DER $i \in V$ is written as

$$\begin{cases} \dot{\boldsymbol{x}}_i(t) = A_{ii}\boldsymbol{x}_i(t) + \boldsymbol{b}_i u_i(t) + \boldsymbol{g}_i\psi_i(t) + \\ \qquad M_i\boldsymbol{d}_i + \boldsymbol{\xi}_i(t) + \boldsymbol{\omega}_i(t), \\ \boldsymbol{y}_i(t) = \boldsymbol{x}_i(t) + \boldsymbol{\rho}_i(t), \end{cases} \tag{1}$$

where $\boldsymbol{x}_i(t) = [V_i(t), I_{ti}(t), v_i(t)]^T$ is the state of system, $u_i(t)$ is the primary control input, $\psi_i(t)$ is the secondary control input, $y_i(t)$ is the output vector, and $\boldsymbol{\omega}_i(t) \leq \bar{\boldsymbol{\omega}}_i$ and $\boldsymbol{\rho}_i(t) \leq \bar{\boldsymbol{\rho}}_i$ are the bounded system and measurement noises, respectively. The third element of $\boldsymbol{x}_i(t)$ denotes the integral of the PCC voltage tracking error satisfying $\dot{v}_i(t) = V_{ref,i} + \psi_i(t) - V_i(t)$. The $V_{ref,i}$ is the nominal reference voltages of PCC, $d_i = [I_{Li}, V_{ref,i}]^T$ is exogenous input, and ξ_i is the physical couplings with neighboring DERs.

The primary controller is designed to track the nominal reference PCC voltage with the local output $y_i(t)$, and the primary control input is calculated as

$$u_i(t) = V_{ti} = \boldsymbol{k}_i\boldsymbol{y}_i(t), \tag{2}$$

where $k_i = [k_1, k_2, k_3]^{\mathrm{T}}$ is the control gain vector that should be carefully chosen to guarantee the voltage stability [14].

The secondary control input is calculated based on the consensus principle [4] as follows

$$\dot{\psi}_i(t) = -[0, k_I, 0] \sum_{j\in N_i^c} a_{ij}^c \left(\frac{\boldsymbol{y}_i(t)}{I_{ti}^s} - \frac{\boldsymbol{y}_{i,j}^c(t)}{I_{ti}^s}\right), \tag{3}$$

where k_I is the weight parameter, $a_{ij}^c > 0$ if DERs i and j are connected by a communication link (otherwise $a_{ij}^c = 0$), and $y_{i,j}^c$ represents the received output vector from DER j through the communication link.

When the primary and secondary controllers satisfy certain conditions that are formally given in [4], then voltage balancing and current sharing can be achieved. The definitions of the two control objectives are as follows

Definition 1 (Current Sharing): Current sharing means that each DER should supply load currents proportionally to its rated value, i.e.,

$$\frac{I_{ti}}{I_{ti}^s} = \frac{I_{tj}}{I_{tj}^s}, i, j \in V, \tag{4}$$

where $I_{ti}^s > 0$ and $I_{tj}^s > 0$ are rated output currents corresponding to DER i and DER j, respectively.

Definition 2 (Voltage Balancing): Voltage balancing means that the average of PCC voltages should be equal to the nominal reference voltage V_{ref}^{nom}, i.e.,

$$\sum_i^N V_i(t) = V_{ref}^{nom}. \tag{5}$$

B. UIO-based Detector

Each DER is equipped with a UIO-based detector [9], which observes $y^c_{j,i}(t)$ from the neighboring DERs to judge whether there are abnormalities. The dynamics of the UIO-based detector are

$$\mathrm{UIO}_{i,j}\begin{cases}\dot{\boldsymbol{z}}_{i,j}(t)=F_j\boldsymbol{z}_{i,j}(t)+\hat{K}_j\boldsymbol{y}^c_{i,j}(t),\\ \hat{\boldsymbol{x}}_{i,j}(t)=\boldsymbol{z}_{i,j}(t)+H_j\boldsymbol{y}^c_{i,j}(t),\end{cases}\tag{6}$$

where $\hat{\boldsymbol{x}}_{i,j}(t)$ represents the estimated state of $\boldsymbol{x}_j(t)$, the details of the choose of parameters F_j, H_j, $\hat{K}_j$ can refer to [15]. With the estimated system state, a detection residual can be calculated to verify the integrity of the received output vector, i.e.,

$$\boldsymbol{r}_{i,j}(t)=\boldsymbol{y}^c_{i,j}(t)-\hat{\boldsymbol{x}}_{i,j}(t)\tag{7}$$

to compare with $\bar{\boldsymbol{r}}_{i,j}(t)$ as

$$|\boldsymbol{r}_{i,j}(t)|\le\bar{\boldsymbol{r}}_{i,j}(t)=\kappa e^{-\mu t}\big(\bar{\boldsymbol{\sigma}}_{2i,j}(0)+\bar{\boldsymbol{\sigma}}_{3i,j}(t)\big)+|T_j|\bar{\boldsymbol{\rho}}_j,\tag{8}$$

where κ,μ are appropriately chosen parameters and $\bar{\boldsymbol{\sigma}}_{2i,j}(0),\bar{\boldsymbol{\sigma}}_{3i,j}(t)$ are bounds regarding the initial state estimation errors and noises [16].

C. Attack Model

In this subsection, we introduce the attack model. We assume that the attacker has the capability to intrude into the communication link and manipulate the communicated data. When the communication link $(i,j)\in E_c$ is under attack, the corrupted data $y^{a,c}_{i,j}(t)$ is modeled as

$$y^{a,c}_{i,j}(t)=\boldsymbol{y}_j(t)+\beta(t-T_a)\boldsymbol{\phi}_{i,j}(t),\tag{9}$$

where $\beta(t-T_a)$ represents a step function meaning that the attack is activated at $t=T_a$ and $\boldsymbol{\phi}_{i,j}(t)$ is the attack vector set by the attacker. To simplify the subsequent analysis, we have the following Assumption:

Assumption 1: The attacker only intrudes into one communication link each time and the injected bias vector $\phi_{i,j}$ is constant, i.e., $\boldsymbol{\phi}_{i,j}=[\phi^{(1)}_{i,j},\phi^{(2)}_{i,j},\phi^{(3)}_{i,j}]^{\mathrm{T}}$.

In this paper, we only consider the simplest case, and the general case where multiple communication links are attacked simultaneously with time-varying bias vectors will be investigated in the further work.

D. Problems of Interest

The UIO-based detector and the improved one [15] can detect and locate the attacked communication links. However, when the communication links are in key positions of the network, and simply cutting off the links will disconnect the network, under which the control objectives may be unachievable. Hence, in this paper, we propose a distributed data recovery scheme to eliminate the attack impact after the attacked link has been located. The following problems are formulated: 1) How to estimate the injected bias with merely the local information? 2) How to obtain the accurate injected bias to totally eliminate the adverse impact caused by FDIAs?

III. Distributed Data Recovery Scheme

In this section, we introduce the design of the distributed data recovery scheme. Under Assumption 1, the PCC voltage will grow up with a constant slope. Specifically, the PCC voltage under any single FDIA satisfies

Lemma 1: Under Assumption 1, any single FDIA can result in

$$V^a_i(\infty)=-\frac{k_I a^c_{ij}}{NI^s_{tj}}\boldsymbol{k}^T\phi^{(2)}_{i,j}(t-T_a)+V_{ref,i},\tag{10}$$

the slope of curve of $V^a_i(\infty)$ is represent by

$$k_{slope}=-\frac{k_I a^c_{ij}}{NI^s_{tj}}\boldsymbol{k}^T\phi^{(2)}_{i,j}.\tag{11}$$

The intuition of the data recovery scheme is to estimate the injected bias $\phi^{(2)}_{i,j}$ by observing the slope k_{slope}. It is noted that the estimation of $\phi^{(2)}_{i,j}$ can fully eliminate the adverse impact of the FDIA as only $\phi^{(2)}_{i,j}$ actually affects the secondary controller according to (3). Additionally, the mean filter is adopted to reduce the impact of measurement noises and small oscillations, and finally the obtained precise bias is added to the communicated signal. The overview of the distributed data recovery scheme is shown in Fig. 2.

UIO-based detector observers state y_i and checks whether DER i is under attack or not. Then, MF-based calculator calculates k_{slope}, which represents the slope of V_i. Finally, incremental attack bias calculator (IABC) work out the injected attack data from k_{slope}.

The IABC achieves data recovery based on a conclusion in [16]. With the conclusion of attack impact, we can get the value of attack injection from k_{slope}.

The whole data recovery scheme is designed under the requirement of distributed framework. Each DER only needs local information (voltage and output current of PCC) to estimate the injected bias.

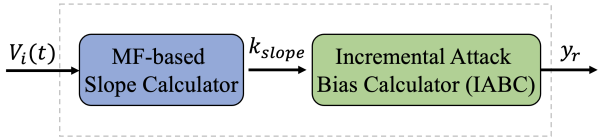


Fig. 2. This figure shows the overview of the data recovery scheme which includes MF-based slope calculator and incremental attack bias calculator.

A. MF-based Slope Calculator

This subsection introduce the MF-based slope calculator.The MF-based slope calculator consists of mean-filter and slope calculator. As shown in Fig. 3, mean-filter smooths the curve of V_i by calculating the average of V_i within the specified length of time window size (TWS). The original data of V_i can not be used to calculate the injected data with k_{slope} directly. Because in the real DCmG system, the curve of V_i oscillates violently because of the integral component of controller, the communication delay and noise.

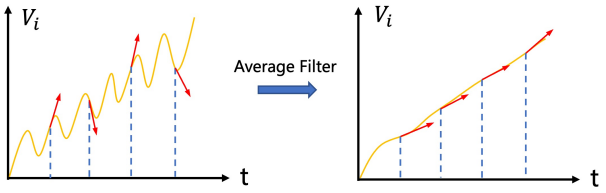


Fig. 3. This figure shows the effect of mean-filter

The average of V_i can be calculated as

$$V_i^{avg}(t) = \begin{cases} V_i^{avg}(nT_{TWS}), & (1+n)T_{TWS} > t \geq nT_{TWS}, \\ & n \in R \\ \dfrac{\int_{nT_{TWS}}^{(1+n)T_{TWS}} V_i(\tau)\, d\tau}{T_{TWS}}, & t = (1+n)T_{TWS}, n \in R \end{cases} \tag{12}$$

Where T_{TWS} represents the length of TWS, which equal to the number of V_i sample points each period of average calculating in the discrete system. The T_{TWS} should be set as smaller as possible, and meanwhile satisfies

$$\frac{\sqrt{\sum_{i=1}^{n_{sa}} \left(\dot{V}_i^{avg}(\frac{i}{T_{da}}) - \mu_d\right)^2}}{n_{sa}} < \kappa_{sa} \tag{13}$$

where T_{da} and n_{sa} is the length of time and the number of points in a sampling process which record the derivative of each sampling point to calculate the smoothness of the curve after filtering, $\mu_d = \sum_{i=1}^{n_{sa}} \dot{V}_i^{avg}(\frac{n_{sa}}{T_{da}})$ represents the mean value of the sampling points. κ_{sa} is a constant which should be small and it's easily to get.

Slope calculator calculates the slope of voltage V_i^{avg} smoothed by mean-filter as

$$k_{slope} = \frac{V_{start}^{sc} - V_{end}^{sc}}{T_{slope}}, \tag{14}$$

where V_{start}^{sc} represents the first sampling value of V_i^{avg} in a period of slope calculating, and V_{end}^{sc} represents the last value. T_{slope} represents the sampling period of slope calculator.

B. Incremental Attack Bias Calculator

In this subsection, the method of IABC is introduced, base on (10), we can calculated injected malicious data with k_{slope}. When the attack is detected, we can calculate the value injected, and eliminate it from the communication link where the observed state y_i is sent to its neighbor DERs.

IABC calculates the injected malicious data of FDI attack by

$$d_{i,j}^a = (\frac{k_I a_{ij}^c}{N I_{tj}^s} \boldsymbol{k}^T t^{3\times 1})^{-1} k_{slope} \tag{15}$$

Where $\boldsymbol{t}^{3\times 1} = [0,1,0]^T$, $\boldsymbol{k}^T = [0,1,0]$.

So, the incremental data recovery is proposed as:

$$y_r = y_r^{last} + \left(\frac{k_I a_{ij}^c}{N I_{tj}^s} k^T t^{3*1}\right)^{-1} k_{slope}, \tag{16}$$

where, y_r represents the recovered data, y_r^{last} represents the calculated y_r in last calculating period.

Finally, the calculated recovery data will added with $y_{i,j}^{a,c}$ as

$$y_{i,j}^c(t) = y_{i,j}^{a,c} - y_r \tag{17}$$

The whole process of data recovery is shown in Algorithm 1. Firstly, the residual $r_{i,j}(t)$ is calculated and compared with $\bar{r}_{i,j}(t)$, if the later is larger, it represent the communication link between DER i and DER j is under attack. Then, the attack eliminating phase start. In the beginning, mean-filter soft $V_i(t)$. Then, (13) is calculated to judge whether V_i^{avg} is soft enough to calculate k_{slope} or not. Then,k_{slope} is calculated with (14), and the attacked data can be calculated with (15). Finally, the recovery value is calculated with (16), and added to $y_{i,j}^{a,c}$ in the communication link where the result will be sent to the secondary controller.

Algorithm 1 Distributed Data Recovery Scheme

Input: The state vector y_i
Output: ***Attack Detect Phase***
Calculate residual $r_{i,j}(t)$ with (7)
if $r_{i,j}(t) \leq \bar{r}_{i,j}(t)$ **then**
    Continue;
else
    Start *attack eliminating phase*
end if
Output: ***Attack Eliminating Phase***
Filter $V_a(t)$ with (12);
if (13) satisfies **then**
    Continue;
else
    Return to attack eliminating phase;
end if
Calculate the slope k_{slope} with (14);
Calculate the attacked data $d_{i,j}^a$ with (15);
Calculate the recovery value y_r with (16);
Add y_r and $y_{i,j}^{a,c}$, and send the result to secondary controller;

IV. Simulations and Experiment Results

In this section, we verify the accuracy and effectiveness of the data recovery scheme through HIL experiments. First, the accuracy of the estimated injected bias is tested, and then the effectiveness of the data recovery scheme is validated.

The HIL experiments are established in the Typhoon HIL testbed to study the data recovery scheme systematically. As shown in Fig. 4, typhoon HIL 602+ emulator is used for ultra-low-latency, ultra-high-fidelity, real-time emulation of power electronics enabled microgrids. Raspberry Pi is used to simulate a real communication environment and PC is used to achieve supervisory control and data acquisition.

We set 6 DERs each of which has a complete hierarchical controller. The nominal PCC reference voltage is set $V_{ref} = 48V$, the load currents of the DERs are set to 2.5A, 2.75A, 2.4A, 2.25A, 2.5A and 2.75A. The attack is launched in the

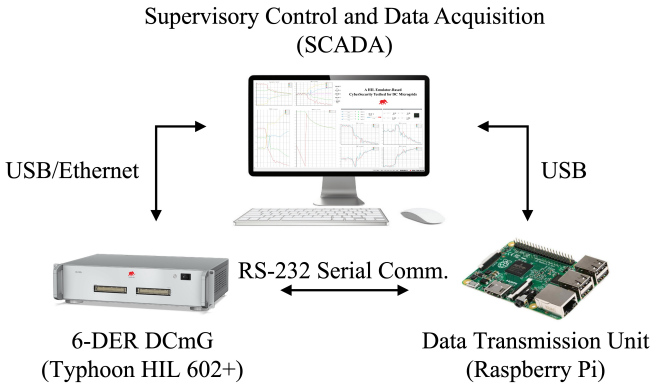


Fig. 4. Implementation overview of the DCmG HIL testbed.

communication link where $y^c_{2,1}$ is sent from DER 2 to DER 1.

A. Accuracy of Estimated Injected Bias

In this subsection, we test the accuracy of the estimated injected bias under constant and ZTS attack. The attack is launched at $t = 6$s. The recovery starts at $t = 7$s, and the period of recovery is set as $T_r = 1$s. The time after which the error between the estimated bias y_r and the actual one is smaller than 5% is calculated and denoted by T_{eb}.

1) Constant Attack: The constant attack means that the injected bias is constant. In this case, the bias vector is set as $\boldsymbol{\phi}_{i,j} = [0, 2, 0]^{\mathrm{T}}$. As shown in Fig. 5, the estimated bias y_r has a nontrivial deviation with the actual bias $d^a_{i,j}$ at the beginning. After $T_{eb} = 4$s, the estimation error of y_r is smaller than 5%. Finally, y_r can infinitely close to the $d^a_{i,j}$.

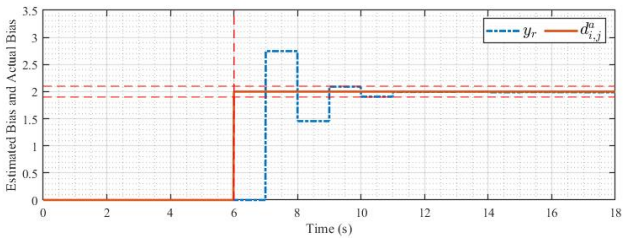


Fig. 5. This figure shows the estimated bias and actual bias under constant attack

2) ZTS Attack: Besides, we verify the accuracy of data recovery to ZTS attack either, the value of which changes with time [16], and the steady-state value of ZTS is set equal to constant attack as $\boldsymbol{\phi}^z_{i,j} = [0, 2, 0]^{\mathrm{T}}$. As shown in Fig. 6, compared with the accuracy of data recovery under constant attack, the estimation error of y_r is larger at the beginning, and $T_{eb} = 6$s, which is longer than constant attack. However, y_r can track the $d^a_{i,j}$ either.

B. Effectiveness of Distributed Data Recovery Scheme

In this subsection, we test the attack impact and effectiveness of the data recovery scheme under constant and ZTS attack.

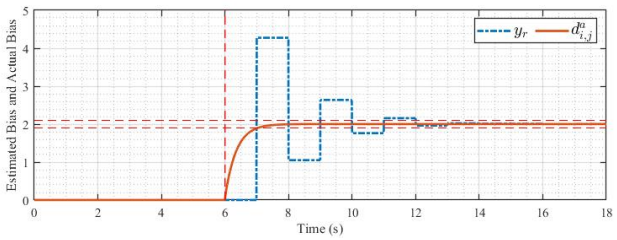


Fig. 6. This figure shows the estimated bias and actual bias under ZTS attack

1) Constant Attack: The bias vector is set as $\boldsymbol{\phi}_{i,j} = [0, 2, 0]^{\mathrm{T}}$. As shown in (a) of Fig. 7, under the attack, the average voltage is increasing rapidly, and all the load current deviate from their original values. The target of voltage balancing and current sharing fails. As shown in (b) of Fig. 7, when recovery scheme is activated at $t = 7$s, each DER voltage stops growing and becomes gentle gradually. And, all output current are close to each other again, current sharing has been recovered. Voltage balancing target is not completely recovered. However, there are some literature show the method to recover the steady state error with PI controller [16], we will not mention it in this paper.

2) ZTS Attack: The steady-state value of ZTS attack is set as $\boldsymbol{\phi}_{i,j} = [0, 2, 0]^{\mathrm{T}}$. As shown in (a) of Fig. 8, the phenomenon is similar to the impact of FDI attack. The increasing speed of average voltage is a little lower. As shown in (b) of Fig. 8, the effectiveness of recovery is a little worse than one under constant attack. However, it can finally recover the current balancing target either.

V. CONCLUSION

In this paper, we focus on hierarchical-control-based DCmGs and propose a distributed data recovery scheme against potential FDIAs in DCmGs. In particular, we study the impact of ZTS attacks and design the attacked data repair method with the calculated result. We further propose the MF-based slope calculator considering the voltage fluctuation in actual DER environment and the calculating methods of the executing period. Furthermore, the data recovery scheme only needs local information. Finally, we evaluate the performance of our proposed distributed data recovery scheme in both simulation and HIL experiments. In future work, we will consider distributed data recovery against other forms of cyber attacks in DCmGs.

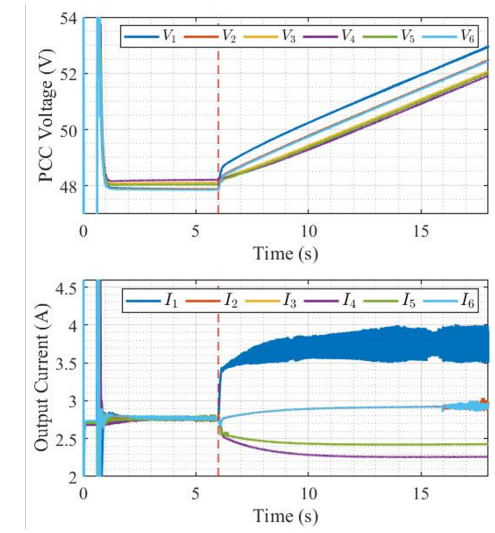
REFERENCES

[1] H. Lotfi and A. Khodaei, "Ac versus dc microgrid planning," *IEEE Transactions on Smart Grid*, vol. 8, no. 1, pp. 296–304, 2017.

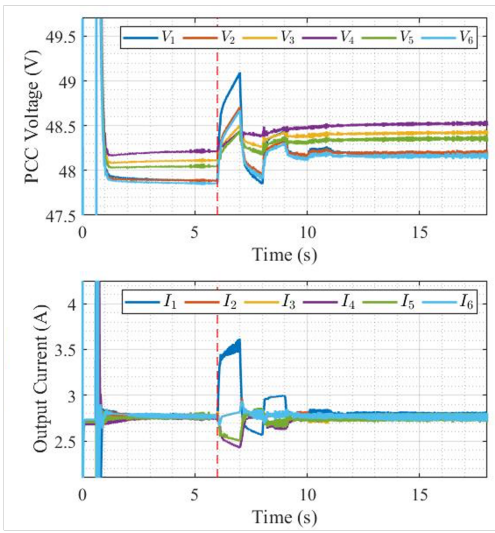
[2] G. Cezar, R. Rajagopal, and B. Zhang, "Stability of interconnected dc converters," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 9–14.

[3] T. Dragičević, X. Lu, J. C. Vasquez, and J. M. Guerrero, "Dc microgrids—part i: A review of control strategies and stabilization techniques," *IEEE Transactions on Power Electronics*, vol. 31, no. 7, pp. 4876–4891, 2016.

[4] M. Tucci, L. Meng, J. M. Guerrero, and G. Ferrari-Trecate, "Stable current sharing and voltage balancing in dc microgrids: A consensus-based secondary control layer," *Automatica*, vol. 95, pp. 1–13, 2018.

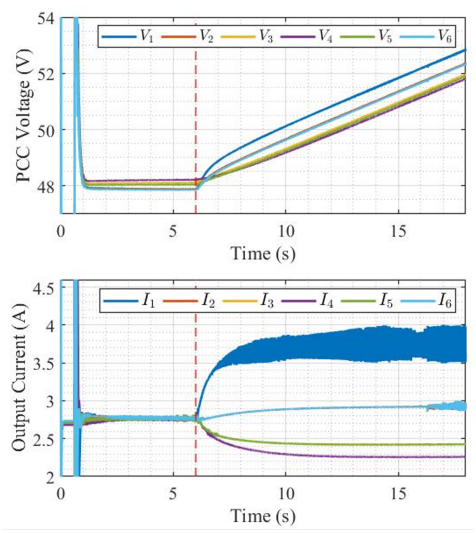


(a) Attack Impact

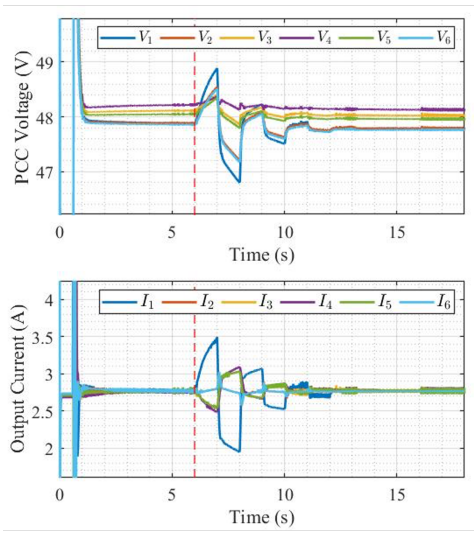


(b) Recovery Effectiveness

Fig. 7. This figure shows the impact of constant attack to DCmG and recovery effectiveness



(a) Attack Impact



(b) Recovery Effectiveness

Fig. 8. This figure shows the impact of ZTS attack to DCmG and recovery effectiveness

[5] O. A. Beg, T. T. Johnson, and A. Davoudi, "Detection of false-data injection attacks in cyber-physical dc microgrids," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2693–2703, 2017.

[6] H. Zhang, W. Meng, J. Qi, X. Wang, and W. X. Zheng, "Distributed load sharing under false data injection attack in an inverter-based microgrid," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 2, pp. 1543–1551, 2019.

[7] Y. Li, P. Zhang, L. Zhang, and B. Wang, "Active synchronous detection of deception attacks in microgrid control systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 1, pp. 373–375, 2017.

[8] A. J. Gallo, M. S. Turan, F. Boem, T. Parisini, and G. Ferrari-Trecate, "A distributed cyber-attack detection scheme with application to dc microgrids," *IEEE Transactions on Automatic Control*, vol. 65, no. 9, pp. 3800–3815, 2020.

[9] S. Abhinav, H. Modares, F. L. Lewis, F. Ferrese, and A. Davoudi, "Synchrony in networked microgrids under attacks," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6731–6741, 2018.

[10] S. Sahoo, S. Mishra, J. C.-H. Peng, and T. Dragičević, "A stealth cyber-attack detection strategy for dc microgrids," *IEEE Transactions on Power Electronics*, vol. 34, no. 8, pp. 8162–8174, 2019.

[11] A. O. Aluko, R. P. Carpanen, D. G. Dorrell, and E. E. Ojo, "Vulnerability analysis of false data injection attacks on the frequency stability of isolated microgrids," in *2021 Southern African Universities Power Engineering Conference/Robotics and Mechatronics/Pattern Recognition Association of South Africa (SAUPEC/RobMech/PRASA)*, 2021, pp. 1–6.

[12] H. Zhang, W. Meng, J. Qi, X. Wang, and W. X. Zheng, "False data injection attacks on inverter-based microgrid in autonomous mode," in *Distributed control methods and cyber security issues in microgrids*, 2020, pp. 125–146.

[13] M. Liu, Z. Jin, J. Xia, M. Sun, R. Deng, and P. Cheng, "Demo abstract: A hil emulator-based cyber security testbed for dc microgrids," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2021, pp. 1–2.

[14] P. Nahata, R. Soloperto, M. Tucci, A. Martinelli, and G. Ferrari-Trecate, "A passivity-based approach to voltage stabilization in dc microgrids with zip loads," *Automatica*, vol. 113, p. 108770, 2020.

[15] M. Liu, C. Zhao, Z. Zhang, R. Deng, P. Cheng, and J. Chen, "Converter-based moving target defense against deception attacks in dc microgrids," *IEEE Transactions on Smart Grid*, pp. 1–1, 2021.

[16] M. Liu, C. Zhao, R. Deng, P. Cheng, and J. Chen, "False data injection attacks and the distributed countermeasure in dc microgrids," *IEEE Transactions on Control of Network Systems*, pp. 1–12, 2022.