Detecting False Data Injection Attacks Against Power System State Estimation with Fast Go-Decomposition Approach

Li, Boda; Ding, Tao; Huang, Can; Zhao, Junbo; Yang, Yongheng; Chen, Ying

Published in:
I E E E Transactions on Industrial Informatics

DOI (link to publication from Publisher):
10.1109/TII.2018.2875529

Publication date:
2019

Document Version
Accepted author manuscript, peer reviewed version

Detecting False Data Injection Attacks Against Power System State Estimation with Fast Go-Decomposition (GoDec) Approach

Boda Li, *Student Member*, *IEEE*, Tao Ding, *Member, IEEE*, Can Huang, *Senior Member, IEEE*, Jubo Zhao, *Member, IEEE*, Yongheng Yang, *Senior Member, IEEE*, Ying Chen, *Member, IEEE*

***Abstract*—State estimation is a fundamental function in modern energy management system (EMS), but its results may be vulnerable to false data injection attacks (FDIA). FDIA is able to change the estimation results without being detected by the traditional bad data detection algorithms. In this paper, we propose an accurate and computational attractive approach for FDIA detection. We first rely on the low rank characteristic of the measurement matrix and the sparsity of the attack matrix to reformulate the FDIA detection as a matrix separation problem. Then, four algorithms that solve for this problem are presented and compared, including the traditional Augmented Lagrange Multipliers (ALM), double-noise-dual-problem ALM (DNDP-ALM), the Low Rank Matrix Factorization (LMaFit) and the proposed new "Go Decomposition (GoDec)". Numerical simulation results show that our GoDec outperforms the other three alternatives and demonstrates a much higher computational efficiency. Furthermore, GoDec is shown to be able to handle measurement noise and applicable for large-scale attacks.**

***Index Terms*—Cyber security, false data injection attacks, matrix separation, smart grid, state estimation.**

I. Introduction

POWER system static state estimation (SE) plays an important role in energy management systems (EMS). It provides accurate and reliable state estimates for various EMS functions, such as optimal power flow and contingency analysis [1]-[5]. Typically, SE makes use of a set of redundant measurements to filter out incorrect measurements and find reliable state estimates. After that, the normalized residual based statistical test is performed to detect bad data. The latter can be induced by unintentional and intentional reasons (e.g., device malfunctions and cyber-attacks) [6]-[10]. Among them, false data injection attacks (FDIA) is one of the main challenges as it can bypass the traditional bad data detectors [3].

The FDIA of power system static state estimator was initiated by Liu. *et al*. Following that work, several other works have been carried out. For instance, two security indexes to quantify the threat of FDIA on power grid are proposed in [4]. Gabriela Hug *et al*. extended their work to AC model [5]. In addition, the potential financial loss caused by FDIA is investigated. Reference [6] investigates the finance benefits profited by attacker in an attacked market while [7] analyzes the impact of FDIA on real-time electric market operations.

To secure the state estimation results, several FDIA detection methods have been proposed [10-22]. A new $\mathcal{L}_\infty$ norm detector softening the influences of FDIA is presented in [10]. A generalized likelihood ratio detector incorporating historical data is proposed in [11]. In [12], the short-term state forecasting-aided detection approach that checks the statistical property of the historical data and the received measurements is proposed. Machine learning–based detection approaches are proposed in [13]. The evaluation index using transmission line real and reactive power measurement residuals is presented to identify FDIA. Reference [14] takes the measurement residual based on active and reactive power flow measurements as an evaluation index to identify the false data. A security mechanism based on a multi-agent filtering scheme with a trust-based mechanism is proposed in [15]. Phasor measurement units (PMU) are used in state estimation to determine fault location and ensure the correctness of measurements according to [16-17]. Hence, reference [18] demonstrates the benefits of deploying a limited number of secure PMUs to defend the attack, and references [19-20] utilize a variant Steiner tree and a heuristic algorithm to determine the positions and minimum number of PMUs respectively. Also, new detection approach using D-FACTS (Distributed Flexible AC Transmission System) is investigated in references [21-22] as well.

It should be noted that the measurement matrix is typically low rank and the attack matrix is sparse. As a result, the FDIA detection problem can be transformed into a matrix separation problem, which has been solved by the Augmented Lagrange

This work was supported in part by National Key Research and Development Program of China (2016YFB0901900), in part by National Natural Science Foundation of China (Grant 51607137) and in part by China Postdoctoral Science Foundation (2017T100748). This work was also performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 with Release Number LLNL-JRNL-747165.

B. Li and T. Ding are both with the State Key Laboratory of Electrical Insulation and Power Equipment, Department of Electrical Engineering, Xi'an Jiaotong University, Xi'an, Shaanxi, 710049, China, and the Department of Electrical Engineering, Tsinghua University, Beijing, 100084, China. T. Ding is the correspondence author (tding15@mail.xjtu.edu.cn)

C. Huang is with the Lawrence Livermore National Laboratory, Livermore, CA, USA.

Y. Yang is with the Department of Energy Technology, Aalborg University, Aalborg DK-9220, Denmark.

J. Zhao is with the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Falls Church, VA 22043, USA.

Y. Chen is with the Department of Electrical Engineering, Tsinghua University, Beijing, 100084, China.

Multipliers (ALM) and the Low Rank Matrix Factorization (LMaFit) approaches [23-24], respectively. As a promotion of the ALM method, double-noise-dual-problem ALM (DNDP-ALM) in reference [32] can also solve the matrix separation problem. However, the computational efficiencies of ALM and DNDP-ALM are not satisfactory, which limit their practical value. By contrast, although LMaFit has good computational efficiency, it obtains quite low statistical detection accuracy of the FDIA. To achieve a better balance between computational efficiency and detection accuracy, this paper proposes a new Go Decomposition (GoDec) approach, which has the following salient features:

(i) In the same condition, when there is no noise, GoDec has the similar computational efficiency as LMaFit while achieving higher accuracy of FDIA detection than the LMaFit; on the other hand, GoDec achieves the similar FDIA detection accuracy as ALM and DNDP-ALM while showing much higher computational efficiency.
(ii) The proposed GoDec is able to handle FDIA detection problems with noise, yielding more practical separation results than the ALM and the LMaFit. Compared with the DNDP-ALM which also considers noise in detection, GoDec has a higher precision and faster calculation speed.
(iii) GoDec is scalable to the large-scale attacks while ALM, DNDP-ALM and LMaFit have huge difficulties.

It should be noted that the proposed method is based on the DC power flow model for illustration and comparison with other methods, but this method can also be extended to the AC power flow model.

The rest of the paper is organized as follows. Section II presents the system model and explains the concept of FDIA. The problem of FDIA detection using matrix separation technique is formulated in Section III. Section IV displays the numerical results and the comparisons among other methods. Finally, Section V concludes the paper.

II. Preliminaries

A. Power System State Estimation

Power system static state estimator normally utilizes the measured measurements to infer the unknown state variables. The estimation model that relates measurements to state variables can be expressed as

$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{e} \tag{1}$$

where $\boldsymbol{z} \in \mathbb{R}^m$ and $\boldsymbol{x} \in \mathbb{R}^n$ denote the measurements and the state variables, respectively; $\boldsymbol{e}$ is the Gaussian noise with zero mean and covariance matrix $\boldsymbol{R}$, and $\boldsymbol{H} \in \mathbb{R}^{m\times n}$ is the Jacobian matrix.

In this paper, the DC model is employed to investigate the impact of FDIA on the power flow on the transmission system, where the voltage magnitudes of all buses are supposed to be 1 p.u.. Thus, $\boldsymbol{x}$ only contains the bus phase angles $\boldsymbol{\theta}$ and the measurements $\boldsymbol{z}$ consists of the active power flows $\boldsymbol{F}$ and power injections $\boldsymbol{P_{inj}}$. Define $\boldsymbol{z} = (z_1, z_2, \cdots, z_m)^{\mathrm{T}}$ and $\boldsymbol{\theta} = (\theta_1, \theta_2, \cdots, \theta_n)^{\mathrm{T}}$, we have:

$$\boldsymbol{F} = \boldsymbol{X^{-1}S\theta} \tag{2}$$

$$\boldsymbol{P_{inj}} = \boldsymbol{B\theta} \tag{3}$$

where $\boldsymbol{B}$ is the bus susceptance matrix of the system; $\boldsymbol{X}$ is the reactance matrix and $\boldsymbol{S}$ is the shift factor of line measurements. Hence, the measurements $\boldsymbol{z}$ and the Jacobian matrix $\boldsymbol{H}$ can be expressed as:

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{P_{inj}} \end{bmatrix} \tag{4}$$

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{X^{-1}S} \\ \boldsymbol{B} \end{bmatrix} \tag{5}$$

Suppose that the noise $\boldsymbol{e}$ in (1) is independent, thus, the covariance matrix $\boldsymbol{R}$ is a diagonal matrix. The state estimation problem above can be solved by weighted least square (WLS) estimator, yielding

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{z} \tag{6}$$

Consequently, the estimated measurements $\hat{\boldsymbol{z}}$ can be expressed as:

$$\hat{\boldsymbol{z}} = \boldsymbol{H}\hat{\boldsymbol{\theta}} = \boldsymbol{H}(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{z} = \boldsymbol{K}\boldsymbol{z} \tag{7}$$

where $\boldsymbol{K} = \boldsymbol{H}(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}$. Thus, the residuals of the measurements are defined as:

$$\boldsymbol{r} = \boldsymbol{z} - \hat{\boldsymbol{z}} = (\boldsymbol{I} - \boldsymbol{K})\boldsymbol{e} \tag{8}$$

Since the square of the $\mathcal{L}_2$ norm $\|\boldsymbol{r}\|_2^2$ follows the χ^2 distribution with the degree of freedom m − n, χ^2-test can be applied on the measurement residuals $\boldsymbol{r}$ for bad data detection. If $\|\boldsymbol{r}\|_2^2 > \tau^2$, then the bad data might exist, where τ is determined by a hypothesis test $\Pr\{\|\boldsymbol{r}\|_2^2 \geq \tau^2\} = \alpha$ with a significant level α.

B. False Data Injection Attacks

Traditionally, bad data can be detected using Largest Normalized Residual (LNR) test. However, attack vectors constructed by the hacker are able to circumvent LNR test, imposing significant biases to the estimation results. Suppose that the attack vector is $\boldsymbol{a}$, the deviation of state variables caused by $\boldsymbol{a}$ is denoted as $\boldsymbol{c}$, then we have

$$\boldsymbol{a} = \boldsymbol{H}\boldsymbol{c} \tag{9}$$

Thus, the measurements collected by EMS can be expressed as:

$$\boldsymbol{z_a} = \boldsymbol{z_0} + \boldsymbol{a} = \boldsymbol{H}(\boldsymbol{\theta} + \boldsymbol{c}) + \boldsymbol{e} = \boldsymbol{H}\boldsymbol{\theta_a} + \boldsymbol{e} \tag{10}$$

where $\boldsymbol{z_a}$ is the malicious measurements and $\boldsymbol{\theta_a}$ corresponds to the result of state estimation using $\boldsymbol{z_a}$. The residual $\|\boldsymbol{r}\|_2$ in this situation is:

$$\begin{aligned} \|\boldsymbol{r}\|_2 &= \|\boldsymbol{z_a} - \boldsymbol{H}\boldsymbol{\theta_a}\|_2 = \|\boldsymbol{z_0} + \boldsymbol{a} - \boldsymbol{H}(\boldsymbol{\theta} + \boldsymbol{c})\|_2 \\ &= \|\boldsymbol{z_0} - \boldsymbol{H}\boldsymbol{\theta}\|_2 \end{aligned} \tag{11}$$

This means that the attack vector does not change the measurement residual and as a result it alters state variable from $\boldsymbol{\theta}$ to $\boldsymbol{\theta} + \boldsymbol{c}$ successfully without being detected.

In practice, [3] reveals that it is unlikely the hacker can attack all meters. Instead, he is limited to the access of limited resources used for compromising the meters persistently. On the

other hand, PMUs are widely used in the power system, which can provide accurate voltage angles and power flows. The utilization of PMUs leads to the decrease of meters that the attacker can compromise [18]. These reasons guarantee the sparsity of the attack vectors, and our research is based on this characteristic.

III. Problem Formulation and Solution

In this section, the problem formulation of FDIA detection is provided and the corresponding solutions are presented.

A. Problem Formulation

1) Basic Assumptions

Before we describe the problem, we first establish some basic assumptions. Below, we will provide three main assumptions and explain them respectively.

(i) The attacker can obtain the measurement matrix $\boldsymbol{H}$ of the power system.

The power system is a typical industrial control system (ICS), and an attacker must obtain enough information to successfully invade the grid and eventually cause load loss. In state estimation, the measurement matrix $\boldsymbol{H}$ is related to the topology of the grid. The attacker can obtain the network topology in a variety of ways, and based on this, $\boldsymbol{H}$ can be inferred. Researchers in [3] [23]-[25] have carried out corresponding researches on the basis of this assumption. Here, this assumption is only used to construct the attack data for simulation. It will not affect the modeling of the problem and solutions of detection methods.

(ii) The attacker's resources are limited.

Considering that the attacker's resources (personnel, attackable instrumentation, financial resources, etc.) are limited, we assume that the attacker can only corrupt a part of the data. Based on this assumption, the attack matrix composed of attack data for a period of time must have a sparse property.

(iii) The measurements and states change slowly in a steady state power system.

The power system is a continuously changing stable system. Under steady working conditions, the changes of various measurements and states in the system are very slow. Therefore, the data in the power grid changes little, or they are almost unchanged over a period of time. Under this assumption, the data matrix composed of the historical measurement vectors and the latest measurement vector will have a low-rank characteristic.

All three assumptions are closely related to the actual situation. Assumption (i) shows that it is feasible to construct the attack vector using the method provided in Section II-B. Assumption (ii) and (iii) indicate that the matrix formed by the attack data has a sparse property, and the matrix formed by the measurements has a low rank property. All these assumptions laid the foundation for subsequent research.

2) Basic Methodology of FDIA Detection

In the presence of FDIA, the attacked measurement at EMS includes a measurement component and an attack component as follows:

$$\boldsymbol{Z_a} = \boldsymbol{Z_0} + \boldsymbol{A} \tag{12}$$

where $\boldsymbol{Z_a} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_t\,] \in \mathbb{R}^{m\times t}$ denotes the measurement attacked at time $\boldsymbol{t}$; $\boldsymbol{Z_0} = [\boldsymbol{z}_{01}, \boldsymbol{z}_{02}, \cdots, \boldsymbol{z}_{0t}\,] \in \mathbb{R}^{m\times t}$ and $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t\,] \in \mathbb{R}^{m\times t}$ denote the measurement component and the attack component, respectively; $\boldsymbol{z}_j$ and $\boldsymbol{a}_j$ denote the measurement and the attack at time j, respectively.

Based on the assumptions given in the previous part, we can find that $\boldsymbol{Z_0}$ is a low rank matrix and $\boldsymbol{A}$ is a sparse matrix. This is because most of state variables change gradually (i.e. the intrinsic low-dimensional nature of power grid states) and most of attacks only affect a limited number of measurements (i.e. the sparse nature of FDI attacks). Matrix separation is a technique which is used for separating a matrix consisting of a low rank matrix and a sparse matrix [26]. In detection problem, $\boldsymbol{Z_a}$ can be regarded as the original matrix, and $\boldsymbol{Z_0}$ and $\boldsymbol{A}$ can be regarded as its low-rank components and sparse components, respectively. Thus, the FDIA detection problem can be viewed as a matrix separation problem and expressed as follows:

$$\min_{\boldsymbol{Z_0},\boldsymbol{A}} \operatorname{rank}(\boldsymbol{Z_0}) + \|\boldsymbol{A}\|_0\,, \quad s.t. \quad \boldsymbol{Z_a} = \boldsymbol{Z_0} + \boldsymbol{A} \tag{13}$$

where $\operatorname{rank}(\boldsymbol{Z_0})$ means the rank of $\boldsymbol{Z_0}$ and $\|\boldsymbol{A}\|_0$ means the number of the nonzero entries of $\boldsymbol{A}$.

So far, we have transformed the FDIA detection problem into a matrix separation problem. In order to facilitate the reader to understand our detection process, we have drawn a flowchart as follows.

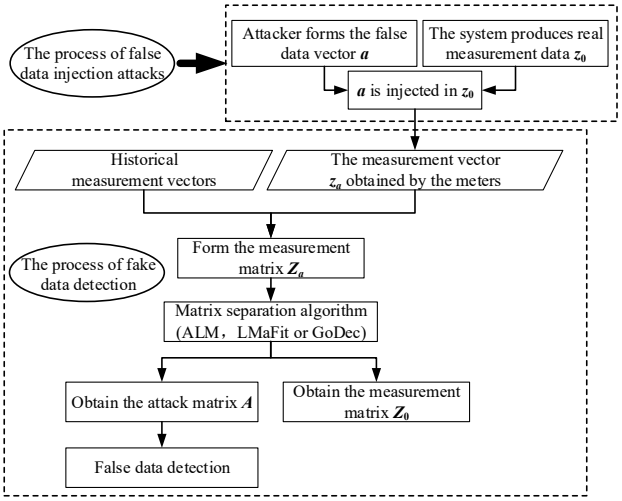


Fig. 1. The process of FDIA and FDIA detection

This flowchart describes the process of FDI attacks and the process of detecting FDIA. When the attacker attacks the system, the attack data $\boldsymbol{a}$ is injected into the normal measurement data $\boldsymbol{z}_0$. At this point, the data collected by the meter is $\boldsymbol{z_a}$.

When the system begins to detect the attack, it combines the current measurements with the historical measurements to form the measurement matrix $\boldsymbol{Z_a}$. After that, matrix separation is performed on $\boldsymbol{Z_a}$. If the separated $\boldsymbol{A}$ matrix is not an empty matrix, then the location of the attack and the magnitude of the attack can be determined based on the location of the non-zero elements in $\boldsymbol{A}$.

The most important section of the entire detection process is the matrix separation operation. The low-rank and sparse ma-

trix separation problem above characterize the low rank property of the measurement matrix and the sparse property of the attack matrix. However, this optimization problem is generally non-deterministic polynomial-time hard and difficult to get a global optimum [29]. To address that, three approaches including the ALM-based methods, the LMaFit and the proposed GoDec are presented and discussed.

B. ALM-based Solution

The ALM method to solve the matrix separation problem was first proposed by Lin. *et al*. This method is widely used in the engineering field and is constantly being developed by many other researchers.

Below, we will introduce the ALM-based solution in two parts. First, the traditional ALM method is introduced. Secondly, we introduce the latest method named DNDP-ALM developed from that traditional ALM method. In section IV, we tested these two methods and compared their performances.

1) Traditional ALM method

In this approach, the matrix separation problem (13) is reformulated as a convex optimization problem (14), in which $\mathrm{rank}(\boldsymbol{Z_0})$ and $\|\boldsymbol{A}\|_{\boldsymbol{0}}$ are replaced by their convex relaxation $\|\boldsymbol{Z_0}\|_*$ and $\|\boldsymbol{A}\|_1$, respectively [26].

$$\min_{\boldsymbol{Z_0},\boldsymbol{A}} \|\boldsymbol{Z_0}\|_* + \lambda\|\boldsymbol{A}\|_1, \quad s.t. \quad \boldsymbol{Z_a} = \boldsymbol{Z_0} + \boldsymbol{A} \tag{14}$$

where $\|\cdot\|_*$ represents the nuclear norm defined as the sum of all singular values of the matrix and $\|\cdot\|_1$ represents the $\mathcal{L}_1$ norm defined as the sum of absolute values of all entries of the matrix; λ is a positive weighting factor, which is usually set to $1/\sqrt{\max(m,t)}$ with $\boldsymbol{Z_a}$ dimensions m and t.

To solve the problem (14), ALM can be used and the augmented Lagrange function can be written as:

$$L(\boldsymbol{Z_0},\boldsymbol{A},\boldsymbol{Y},\mu) = \|\boldsymbol{Z_0}\|_* + \lambda\|\boldsymbol{A}\|_1 + \langle \boldsymbol{Y}, \boldsymbol{Z_a} - \boldsymbol{Z_0} - \boldsymbol{A}\rangle + \frac{\mu}{2}\|\boldsymbol{Z_a} - \boldsymbol{Z_0} - \boldsymbol{A}\|_F^2 \tag{15}$$

where $\boldsymbol{Y}$ is the Lagrange multiplier; μ is a positive scalar, and $\langle\cdot|\cdot\rangle$ is the inner product.

Mathematically, ALM requires singular value decomposition (SVD), which may limit its computing speed and scalability. Interestingly, both Exact ALM (EALM) [23] and Inexact ALM (IALM) [27] algorithms have been used for the FDIA detection. Generally, IALM has a higher computational efficiency than EALM as it reduces the number of SVD as well as the time of SVD computation. The IALM algorithm is briefly depicted in TABLE I, where $\mathcal{S}_\tau\{x\}$ is defined in (16).

$$\mathcal{S}_\tau\{x\} = \mathrm{sgn}(x)\max(|x| - \tau, 0) \tag{16}$$

TABLE I FLOWCHART OF INEXACT ALM METHOD

Algorithm 1 Inexact ALM

Input: $\boldsymbol{Z_a} \in \mathbb{R}^{m\times t}$; $\lambda = 1/\sqrt{\max(m,t)}$;
Initialize: $\boldsymbol{Y}_{[0]} = 0$; $\boldsymbol{Z_0}_{[0]} = 0$; $\boldsymbol{A}_{[0]} = 0$; $\mu_{[0]} > 0$; $\alpha > 0$; $k = 0$;
while not converge **do**
//solve $\boldsymbol{Z_0}_{[k+1]} = \arg\min_{\boldsymbol{Z_0}} L(\boldsymbol{Z_0}, \boldsymbol{A}_{[k]}, \mu_{[k]}, \boldsymbol{Y}_{[k]})$
$(\boldsymbol{U},\boldsymbol{S},\boldsymbol{V})^{\mathrm{T}} = \mathrm{svd}\big(\boldsymbol{Z_a} - \boldsymbol{A}_{[k]} + \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]}\big)$
//obtain $[\boldsymbol{U},\boldsymbol{S},\boldsymbol{V}]$
$\boldsymbol{Z_0}_{[k+1]} = \boldsymbol{U}\mathcal{S}_{\mu_{[k]}^{-1}}\{\boldsymbol{S}\}\boldsymbol{V}^{\mathrm{T}}$
//solve $\boldsymbol{A}_{[k+1]} = \arg\min_{\boldsymbol{Z_0}} L(\boldsymbol{Z_0}_{[k+1]}, \boldsymbol{A}, \mu_{[k]}, \boldsymbol{Y}_{[k]})$
$\boldsymbol{A}_{[k+1]} = \mathcal{S}_{\lambda\mu_{[k]}^{-1}}\big\{\boldsymbol{Z_a} - \boldsymbol{Z_0}_{[k+1]} + \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]}\big\}$
$\boldsymbol{Y}_{[k+1]} = \boldsymbol{Y}_{[k]} + \mu_{[k]}\big(\boldsymbol{Z_a} - \boldsymbol{Z_0}_{[k+1]} - \boldsymbol{A}_{[k+1]}\big)$
$\mu_{[k+1]} = \alpha\mu_{[k]}$
$k = k + 1$
end while
Return: $\boldsymbol{Z_0}_{[k]}$; $\boldsymbol{A}_{[k]}$;
Output: $\boldsymbol{Z_0}_{[k]}$; $\boldsymbol{A}_{[k]}$;

2) Improved ALM-based Solution

Due to the high accuracy of traditional ALM method, it is widely used in various fields. But considering that traditional ALM is based on equation (12), it does not take measurement noise into account, which limits its scope of use. DNDP-ALM proposed in 2017 [32] improved the original optimization problem and incorporated noise into the constraints.

DNDP-ALM is an improvement on the original method, and there is not much difference in the solution process. Here, we briefly introduce this method and give the specific process of solving matrix separation problem.

The optimization problem that DNDP-ALM needs to solve is developed from equation (14). Here, we define the measurement noise matrix as $\boldsymbol{N}$. Thus, the following convex optimization problem can be represented as:

$$\min_{\boldsymbol{Z_0},\boldsymbol{A}} \|\boldsymbol{Z_0}\|_* + \lambda\|\boldsymbol{A}\|_1 + \beta\|\boldsymbol{N}\|_F, s.t. \boldsymbol{Z_a} = \boldsymbol{Z_0} + \boldsymbol{A} + \boldsymbol{N} \tag{17}$$

where β is a positive weighting factor, and $\|\cdot\|_F$ denotes the Frobenius norm of matrix $\boldsymbol{N}$. Problem (17) can also be solved by ALM. The augmented Lagrange function can be written as:

$$L(\boldsymbol{Z_0},\boldsymbol{A},\boldsymbol{Y},\mu) = \|\boldsymbol{Z_0}\|_* + \lambda\|\boldsymbol{A}\|_1 + \beta\|\boldsymbol{N}\|_F + \langle \boldsymbol{Y}, \boldsymbol{Z_a} - \boldsymbol{Z_0} - \boldsymbol{A} - \boldsymbol{N}\rangle + \frac{\mu}{2}\|\boldsymbol{Z_a} - \boldsymbol{Z_0} - \boldsymbol{A} - \boldsymbol{N}\|_F^2 \tag{18}$$

DNDP modifies the objective function and constraints of the original optimization problem, so that the new method can take the noise into consideration. The flowchart of DNDP-ALM is provided in TABLE II.

TABLE II FLOWCHART OF DNDP-ALM METHOD

Algorithm 2 DNDP-ALM

Input: $\boldsymbol{Z_a} \in \mathbb{R}^{m\times t}$; $\lambda = 1/\sqrt{\max(m,t)}$;
Initialize: $\boldsymbol{Y}_{[0]} = 0; \boldsymbol{Z_0}_{[0]} = 0$; $\boldsymbol{A}_{[0]} = 0; \boldsymbol{N}_{[0]} = 0$;
$\alpha > 0$; $\beta > 0$; $\mu_{[0]} > 0$; $k = 0$;
while $\boldsymbol{Z_0}, \boldsymbol{A}, \boldsymbol{N}$ not converge **do**
$\boldsymbol{Z_0}^0_{[k+1]} = \boldsymbol{Z_0}_{[k]}, \boldsymbol{A}^0_{[k+1]} = \boldsymbol{A}_{[k]}, \boldsymbol{N}^0_{[k+1]} = \boldsymbol{N}_{[k]}, j = 0$
while $\boldsymbol{Z_0}_{[k+1]}, \boldsymbol{A}_{[k+1]}, \boldsymbol{N}_{[k+1]}$not converge **do**
$(\boldsymbol{U},\boldsymbol{S},\boldsymbol{V})^{\mathrm{T}} = \mathrm{svd}\big(\boldsymbol{Z_a} - \boldsymbol{A}^j_{[k+1]} - \boldsymbol{N}^j_{[k+1]} + \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]}\big)$
$\boldsymbol{Z_0}^{j+1}_{[k+1]} = \boldsymbol{U}\mathcal{S}_{\mu_{[k]}^{-1}}\{\boldsymbol{S}\}\boldsymbol{V}^{\mathrm{T}}$
$\boldsymbol{A}^{j+1}_{[k+1]} = \mathcal{S}_{\lambda\mu_{[k]}^{-1}}\big\{\boldsymbol{Z_a} - \boldsymbol{Z_0}^{j}_{[k+1]} - \boldsymbol{N}^j_{[k+1]} + \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]}\big\}$
$(\boldsymbol{M},\boldsymbol{Z},\boldsymbol{N})^{\mathrm{T}} = \mathrm{svd}\big(\boldsymbol{Z_a} + \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]}\big)$

$\boldsymbol{N}_{[k+1]}^{j+1} = \boldsymbol{Z_a} + \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]} - \boldsymbol{M}\mathcal{S}_{\beta}\{\boldsymbol{Z}\}\boldsymbol{N}^{\mathrm{T}}$
$j = j + 1$
end while
$\boldsymbol{Y}_{[k+1]} = \boldsymbol{Y}_{[k]} + \mu_{[k]}\left(\boldsymbol{Z_a} - \boldsymbol{Z_{0[k+1]}} - \boldsymbol{A}_{[k+1]} - \boldsymbol{N}_{[k+1]}\right)$
$\mu_{[k+1]} = \alpha\mu_{[k]}$
$k = k + 1$
end while
Return: $\boldsymbol{Z_{0[k]}}$; $\boldsymbol{A}_{[k]}$;

Output: $\boldsymbol{Z_{0[k]}}$; $\boldsymbol{A}_{[k]}$;

C. LMaFit-based Solution

For the LMaFit approach, the matrix separation problem (13) is converted into the following optimization problem:

$$\min_{\mathbf{U},\mathbf{V},\boldsymbol{Z_0}} \|\boldsymbol{Z_a} - \boldsymbol{Z_0}\|_1, \quad s.t. \quad \boldsymbol{UV} - \boldsymbol{Z_0} = \boldsymbol{0} \tag{19}$$

where $\boldsymbol{Z_0}$ is represented by a product of $\mathbf{U} \in \mathbb{R}^{m\times r}$ and $\mathbf{V} \in \mathbb{R}^{r\times n}$, and r represents the initial rank estimate [28].

This problem can be solved with ALM as well. The augmented Lagrange function is expressed as:

$$L(\boldsymbol{U},\boldsymbol{V},\boldsymbol{Z_0},\boldsymbol{Y},\mu) = \|\mathcal{P}_{\boldsymbol{\Omega}}(\boldsymbol{Z_a} - \boldsymbol{Z_0})\|_1 + \langle \boldsymbol{Y}, \boldsymbol{UV} - \boldsymbol{Z_0}\rangle + \frac{\mu}{2}\|\boldsymbol{UV} - \boldsymbol{Z_0}\|_2^2 \tag{20}$$

Here, the general idea is to factorize the measurement matrix $\boldsymbol{Z_0}$ into the product of two low-rank matrices, instead of minimizing the nuclear norm of $\boldsymbol{Z_0}$. In such a way, SVD is avoided, and the speed and scalability of the algorithms is improved. The flowchart of LMaFit is presented in TABLE III.

TABLE III FLOWCHART OF LOW RANK MATRIX FITTING

Algorithm 3 Low-rank Matrix Fitting

Input: $\boldsymbol{Z_a} \in \mathbb{R}^{m\times t}$; initial rank estimation r;
Initialize: $\boldsymbol{U} \in \mathbb{R}^{m\times r}$; $\boldsymbol{V} \in \mathbb{R}^{r\times t}$; $\boldsymbol{Z_{0[0]}} = \boldsymbol{U} * \boldsymbol{V}$; $\boldsymbol{Y}_{[0]} = 0$; $\mu_{[0]} > 0; \alpha > 0; k = 0$;
while not converge **do**
$\boldsymbol{U}_{[k+1]} = \left(\boldsymbol{Z_0} - \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]}\right)\boldsymbol{V}^{T}(\boldsymbol{V}\boldsymbol{V}^{T})^{-1}$;
$\boldsymbol{V}_{[k+1]} = (\boldsymbol{U}^{T}\boldsymbol{U})^{-1}\boldsymbol{U}^{T}\left(\boldsymbol{Z_0} - \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]}\right)$;
$\boldsymbol{Z_{0[k+1]}} = S_{\mu_{[k]}^{-1}}\left\{\boldsymbol{U}_{[k+1]}\boldsymbol{V}_{[k+1]} - \boldsymbol{Z_a} + \mu_{[k]}^{-1}\boldsymbol{Y}_{[k]}\right\}$;
$\boldsymbol{Y}_{[k+1]} = \boldsymbol{Y}_{[k]} + \boldsymbol{u}_{[k]}\left(\boldsymbol{U}_{[k+1]}\boldsymbol{V}_{[k+1]} - \boldsymbol{Z_{0[k+1]}}\right)$;
$\mu_{[k+1]} = \alpha\mu_{[k]}$;
$k = k + 1$;
possibly re-estimate r, and adjust sizes of the iterates
end while
Return: $\boldsymbol{Z_{0[k]}}$;

Output: $\boldsymbol{Z_{0[k]}}$; $\boldsymbol{Z_a} - \boldsymbol{Z_{0[k]}}$;

D. Proposed GoDec Solution

In the ALM algorithm, the SVD at each iteration is laborious for high-dimensional matrices. While for LMaFit algorithm, SVD is replaced by low-rank matrix factorization, yielding better computational efficiency than ALM However, both ALM and LMaFit don't take the measurement noise into account, yielding biased state estimation results. In addition, the convergence of LMaFit is not guaranteed due to the non-convex nature of the LMaFit problem. Furthermore, both ALM and LMaFit are not able to handle large-scale attacks.

To address these problems, a new algorithm called "Go Decomposition" (GoDec) is proposed in this paper. The general idea is to replace the SVD with Bilateral Random Projections (BRP). In addition, the measurement noise is considered in GoDec, which is ignored by both ALM and LMaFit.

The error of BRP based approximation approaches to the error of SVD approximation under general conditions, but the computing burden of BRP is much less than SVD.

First, the attacked measurement is represented as:

$$\boldsymbol{Z_a} = \boldsymbol{Z_0} + \boldsymbol{A} + \boldsymbol{N}, \ \mathrm{rank}(\boldsymbol{Z_0}) \le r, \mathrm{card}(\boldsymbol{A}) \le p \tag{21}$$

where $\mathrm{card}(\cdot)$ means the number of nonzero entries in the matrix.

The matrix separation problem is transformed into the following optimization problem:

$$\min_{\boldsymbol{Z_0},\boldsymbol{A}} \|\boldsymbol{Z_a} - \boldsymbol{Z_0} - \boldsymbol{A}\|_F^2 \ \ \mathrm{s.t.}\, \mathrm{rank}(\boldsymbol{Z_0}) \le r, \mathrm{card}(\boldsymbol{A}) \le p \tag{22}$$

The measurements $\boldsymbol{Z_0}$ and the attack matrix $\boldsymbol{A}$ can be separated by alternatively solving the following two sub-problems until convergence. Note that the two sub-problems have non-convex constraints, while their global solutions $\boldsymbol{Z_{0[k]}}$ and $\boldsymbol{A}_{[k]}$ can be guaranteed [29].

$$\boldsymbol{Z_{0[k]}} = \arg\min_{\mathrm{rank}(\boldsymbol{Z_0})\le r} \left\|\boldsymbol{Z_a} - \boldsymbol{Z_0} - \boldsymbol{A}_{[k-1]}\right\|_F^2 \tag{23}$$

$$\boldsymbol{A}_{[k]} = \arg\min_{\mathrm{card}(\boldsymbol{A})\le p} \left\|\boldsymbol{Z_a} - \boldsymbol{Z_{0[k]}} - \boldsymbol{A}\right\|_F^2 \tag{24}$$

Second, BRP is used to replace SVD for low-rank approximation to reduce time cost [30], In original GoDec algorithms, the main computation task is to update $\boldsymbol{Z_{0[k]}}$.

For the sub-problem (23), we mainly work on the low-rank approximation with BRP. Let $\boldsymbol{Z_0'} = \boldsymbol{Z_0} + \boldsymbol{N}$ and $\boldsymbol{Z_0'} \in \mathbb{R}^{m\times t}$.

$$\widetilde{\boldsymbol{Z_0'}} = \left[(\boldsymbol{Z_a} - \boldsymbol{A}_{[k-1]})(\boldsymbol{Z_a} - \boldsymbol{A}_{[k-1]})^{\mathrm{T}}\right]^{q}(\boldsymbol{Z_a} - \boldsymbol{A}_{[k-1]}) \tag{25}$$

where q is a positive parameter.

Then, the BRP of $\widetilde{\boldsymbol{Z_0'}}$ is:

$$\boldsymbol{Y_1} = \widetilde{\boldsymbol{Z_0'}}\boldsymbol{S_1}, \boldsymbol{Y_2} = \widetilde{\boldsymbol{Z_0'}}^{\mathrm{T}}\boldsymbol{S_2} \tag{26}$$

where $\boldsymbol{Y_1} \in \mathbb{R}^{m\times r}$ is left random projection of $\widetilde{\boldsymbol{Z_0'}}$; $\boldsymbol{Y_2} \in \mathbb{R}^{t\times r}$ is the right random projection, and r represents the rank of measurement matrix. $\boldsymbol{S_1} \in \mathbb{R}^{t\times r}$ is an independent Gaussian random matrix and $\boldsymbol{S_2} \in \mathbb{R}^{m\times r}$ is a matrix updated by $\boldsymbol{Y_1}$as

$$\boldsymbol{S_2} = \boldsymbol{Y_1} \tag{27}$$

With BRR, the r rank approximation of $\widetilde{\boldsymbol{Z_0'}}$ is

$$\widetilde{\boldsymbol{Z_r}} = \boldsymbol{Y_1}(\boldsymbol{S_2}^{\mathrm{T}}\boldsymbol{Y_1})^{-1}\boldsymbol{Y_2}^{\mathrm{T}} \tag{28}$$

In order to obtain the approximation of $\boldsymbol{Z_0'}$ with the rank r, we calculate the QR decomposition of $\boldsymbol{Y_1}$ and $\boldsymbol{Y_2}$, i.e.:

$$\boldsymbol{Y_1} = \boldsymbol{Q_1}\boldsymbol{R_1}, \boldsymbol{Y_2} = \boldsymbol{Q_2}\boldsymbol{R_2} \tag{29}$$

Then the low-rank approximation of $\boldsymbol{Z_0'}$ is given by:

$$\boldsymbol{Z_0} = (\widetilde{\boldsymbol{Z_r}})^{\frac{1}{2q+1}} = \boldsymbol{Q_1}\left[\boldsymbol{R_1}(\boldsymbol{S_2^{\mathrm{T}}Y_1})^{-1}\boldsymbol{R_2^{\mathrm{T}}}\right]^{\frac{1}{2q+1}}\boldsymbol{Q_2^{\mathrm{T}}} \quad (30)$$

In the calculation process above, GoDec can increase q to reduce the error of BRP. For the sub-problem (24), we take the first p largest elements of $\left|\boldsymbol{Z_a} - \boldsymbol{Z_{0[k]}}\right|$, and assign those values to $\boldsymbol{A}_{[k]}$ in the same position:

$$\boldsymbol{A}_{[k]} = \mathcal{P}_{\Omega}(\boldsymbol{Z_a} - \boldsymbol{Z_{0[k]}}) \quad (31)$$

Finally, we adopt a positive scalar ϵ to check the convergence of the algorithm shown in (32). The overall flowchart of GoDec is presented in TABLE IV.

$$\left\|\boldsymbol{Z_a} - \boldsymbol{Z_{0[k]}} - \boldsymbol{A}_{[k]}\right\|_F^2 / \|\boldsymbol{Z_a}\|_F^2 < \epsilon \quad (32)$$

TABLE IV FLOWCHART OF FAST GO DECOMPOSITION

Algorithm 4 Fast Go Decomposition

Input: $\boldsymbol{Z_a}$, r, p, ϵ, q

Initialize: $\boldsymbol{Z_{0[0]}} = \boldsymbol{Z_a}$, $\boldsymbol{A}_{[0]} = \boldsymbol{0}$

while $\left\|\boldsymbol{Z_a} - \boldsymbol{Z_{0[k]}} - \boldsymbol{A}_{[k]}\right\|_F^2 / \|\boldsymbol{Z_a}\|_F^2 > \epsilon$, **do**

$\widetilde{\boldsymbol{Z_0}} = \left[(\boldsymbol{Z_a} - \boldsymbol{A}_{[k-1]})(\boldsymbol{Z_a} - \boldsymbol{A}_{[k-1]})^{\mathrm{T}}\right]^q(\boldsymbol{Z_a} - \boldsymbol{A}_{[k-1]})$;

$\boldsymbol{Y_1} = \widetilde{\boldsymbol{Z_0}}\boldsymbol{S_1}, \boldsymbol{S_2} = \boldsymbol{Y_1}$;

$\boldsymbol{Y_2} = \widetilde{\boldsymbol{Z_0}}^{\mathrm{T}}\boldsymbol{Y_1} = \boldsymbol{Q_2}\boldsymbol{R_2}, \boldsymbol{Y_1} = \widetilde{\boldsymbol{Z_0}}\boldsymbol{Y_2} = \boldsymbol{Q_1}\boldsymbol{R_1}$;

If $\mathrm{rank}\left(\boldsymbol{S_2^{\mathrm{T}}Y_1}\right) < r$

then $r =$ $\mathrm{rank}\left(\boldsymbol{S_2^{\mathrm{T}}Y_1}\right)$, go to the first step;

end

$\boldsymbol{Z_{0[k]}} = \boldsymbol{Q_1}\left[\boldsymbol{R_1}(\boldsymbol{S_2^{\mathrm{T}}Y_1})^{-1}\boldsymbol{R_2^{\mathrm{T}}}\right]^{1/(2q+1)}\boldsymbol{Q_2^{\mathrm{T}}}$;

$\boldsymbol{A}_{[k]} = \mathcal{P}_{\Omega}(\boldsymbol{Z_a} - \boldsymbol{Z_{0[k]}})$, Ω is the nonzero subset of the first p largest entries of $\left|\boldsymbol{Z_a} - \boldsymbol{Z_{0[k]}}\right|$;

end while

Output: $\boldsymbol{Z_0}$, $\boldsymbol{A}$, $\boldsymbol{Z_a} - \boldsymbol{Z_0}$

Although the derivations of all three methods are based on the feature that the attack matrix is sparse, some scholars expand the scope of application when the attack matrix is not sparse. Arvind Ganesh *et al* [31] analyzed the application in dense error correction by improved weighting parameter λ in (14) slightly, while the performances between ALM and LMaFit are compared in [28] when the matrix is not sparse. As a result, our proposed approach can be applied to dense error problem as well. In fact, in the presence of large-scale attacks the attack matrix can be treated as "dense error", which can be handled by the proposed approach (see the results below).

IV. Numerical results

In this section, numerical simulations are performed to evaluate the performances of the four approaches. Specifically, they are assessed from three aspects, namely, detection accuracy with measurement noise, computational efficiency and scalability for large-scale attacks. All the tests are conducted on the IEEE 118-bus test system. The method of performing attacks can be found in [11] and [25]. In this paper, we focus on FDIA detection, and the attacked meters are selected randomly. Suppose that in a continuous time period T, EMS collected 150 groups of measurements from different snapshots. Hence, the measurement matrix used for simulation is $\boldsymbol{Z_a} \in \mathbb{R}^{303\times150}$. Besides, in this part, the noise in the measurements is Gaussian noise. The rank estimation r in equation (21) is set as $0.05m$, where m is the number of matrix's columns, and p in equation (21) is fixed as $0.05mn$, where n is the number of matrix's rows.

A. Computational Accuracy

The attack matrix formed by attackers contains two kinds of information. One is the value of false injection data, and the other is the location where attackers conduct injection. Below, we will discuss computational accuracy from both numerical detection accuracy and location detection accuracy separately, to illustrate the superiority of our algorithm.

1) Numerical Detection Accuracy

First of all, we will analyze the numerical detection accuracy. To quantify the accuracy of each approach for detecting the data injected by attacker, two metrics are used, including:

(1) the relative reconstruction error $\boldsymbol{\delta}$ for state variables $\boldsymbol{\theta}$

$$\boldsymbol{\delta} = \left(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)./|\boldsymbol{\theta}| \quad (33)$$

where $\widehat{\boldsymbol{\theta}}$ is the result of state estimation using $\boldsymbol{Z_0}$ from matrix separation, $|\boldsymbol{\theta}|$ calculates the absolute value of all entries in $\boldsymbol{\theta}$.

(2) the mean absolute error ε for attack matrix $\boldsymbol{A}$:

$$\varepsilon = \frac{\sum_1^{mn}|\boldsymbol{A}-\boldsymbol{A'}|}{a\times b} \quad (34)$$

where $\boldsymbol{A'}$ is the matrix recovered from the algorithm; $\boldsymbol{A}$ is the attack matrix we construct, and a and b stand for the number of columns and rows of $\boldsymbol{A}$.

First, we make an intuitive comparison of the four algorithms with a series of gray scale images. The original attack matrix constructed with equation (9) is given in Fig. 2, and the attack matrices separated by the main four algorithms are shown in Figs. 3 (a)-(d) and Figs. 4 (a)-(d). The different gray scale in Figs. 2 to 4 represents the different magnitudes of attacks which are generated by us or separated by algorithms. The x-axis represents the sampling time t and y-axis represents the measurements in the measurements matrix.

Note that the measurement matrices in Fig. 3 are without noise while those shown in Fig. 4 are with 5% noise (The gray scale image of the separated results with 10% measurement noise are shown in the appendix.). It is observed from Fig. 3 that four algorithms detect and identify the FDIA with different accuracy: (1) ALM detects some of the FDIA but it could not identify the magnitudes of the false data; (2) DNDP-ALM has significantly improved the performance of detection attacks compared with ALM, but there are still some attacks that cannot be detected; (3) LMaFit detects most of the attacks but performs poorly on identifying the magnitudes; and (4) GoDec is able to detect the attacks and identify the magnitudes simultaneously. In addition, it is observed from Fig.4 that ALM and LMaFit are sensitive to measurement noise and produce poor results, which is not the case for our proposed approach. In addition, even considering the disturbance of noise in the measurements, the detection accuracy of DNDP-ALM is not as high as GoDec.

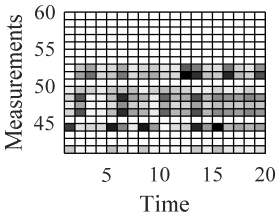
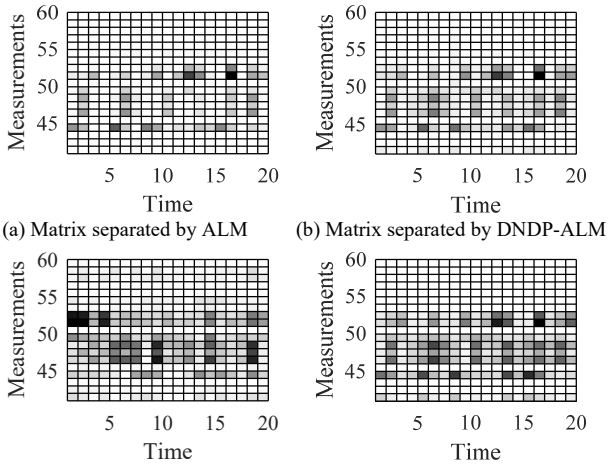


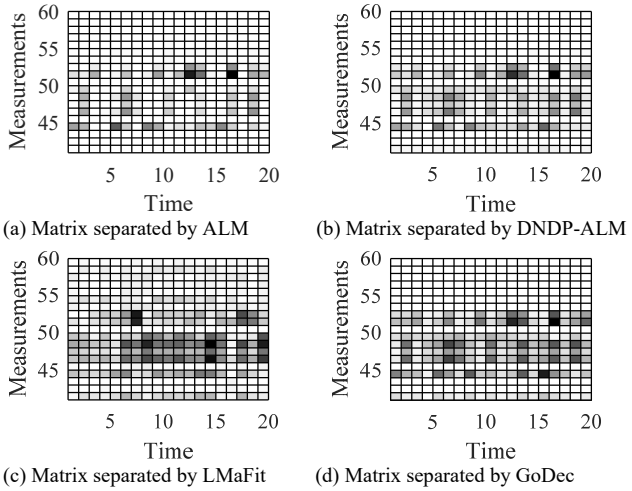
Fig. 2. The original attack matrix constructed with equation (9)



(a) Matrix separated by ALM (b) Matrix separated by DNDP-ALM

(c) Matrix separated by LMaFit (d) Matrix separated by GoDec

Fig. 3. The comparison among matrices separated by different algorithms (the measurements aren't disturbed by noise)



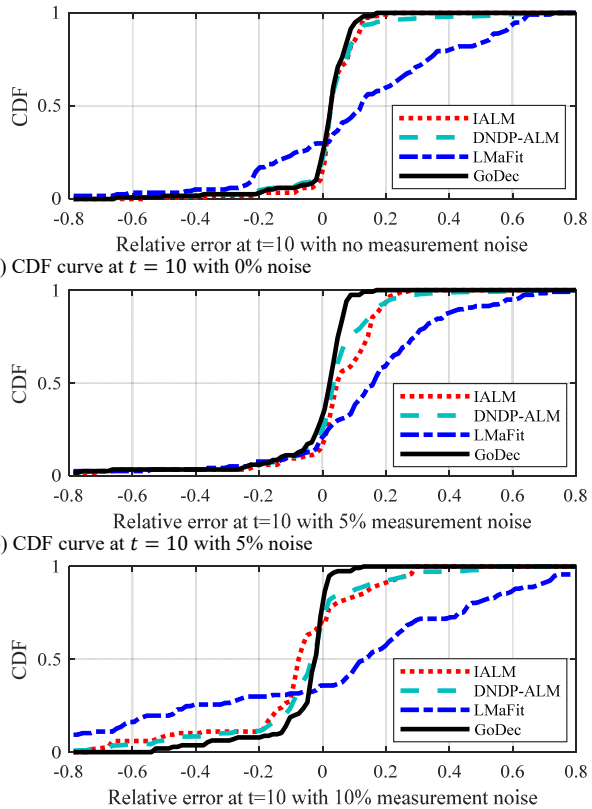
(a) Matrix separated by ALM (b) Matrix separated by DNDP-ALM

(c) Matrix separated by LMaFit (d) Matrix separated by GoDec

Fig. 4. The comparison among original attack matrix and matrices separated by different algorithms (the measurements are disturbed by 5% noise)

Next, we make use of indictors ε and $\boldsymbol{\delta}$ to perform some quantitative analysis. Under different noise level, the mean absolute errors ε between the original attack matrices and the separated matrices are calculated and shown in TABLE V. The maximum relative error $\boldsymbol{\delta_{max}}$ is shown in TABLE VI. We find from these two tables that the estimation errors of all methods increase with the increase of noise level. However, ALM, DNDP-ALM and LMaFit show higher sensitivity to noise level than our GoDec. For example, with 0-10% noise, ALM, DNDP-ALM and GoDec's ε increase from 0.0612 to 0.1449, 0.0610 to 0.1349 and from 0.0602 to 0.1296, respectively, while LMaFit's ε are all above 0.2; and with no noise, ALM, DNDP-ALM and GoDec's $\boldsymbol{\delta_{max}}$% are 34.85%, 28.69% and 26.19%, respectively, while LMaFit's $\boldsymbol{\delta_{max}}$% reaches to 290.25%. Although DNDP-ALM can deal with noise-containing matrix separation problems, its performance at different noise levels is still inferior to GoDec. When the noise increases from 0 to 10%, the values of $\boldsymbol{\delta_{max}}$ and ε of GoDec are always lower than those of DNDP-ALM, which illustrates that GoDec performs better in dealing with false data injection detection problem with noise.

Furthermore, we provide the relative error of the voltage angle and show more details of error distribution of the separated results in Fig. 5. Specifically, we use relative error of state variables to compare the accuracy among all methods. Note that each column of $\boldsymbol{Z_0}$ corresponds to the result of state estimation in a certain time. The target of state estimation is to obtain the accurate state of system, which can be shown by the error distribution of the estimated measurements. Although we have compared the maximum and mean error



(a) CDF curve at $t = 10$ with 0% noise

(b) CDF curve at $t = 10$ with 5% noise

(c) CDF curve at $t = 10$ with 10% noise

Fig. 5. Power state reconstruction performance of four algorithms at specific time instant $t = 10$ with 0% noise, 5% noise and 10% noise.

TABLE V THE MAXIMUM VALUE OF RELATIVE ERROR $\boldsymbol{\delta}$

Algorithm	IALM/EALM			DNDP-ALM			LMaFit			GoDec		
noise	0	5%	10%	0	5%	10%	0	5%	10%	0	5%	10%
δ_{max}	0.3485	0.4637	0.7471	0.2869	0.3299	0.6236	2.9025	3.1941	3.4059	0.2619	0.2755	0.5651

TABLE VI THE MEAN ABSOLUTE ERROR ε OF ATTACK MATRICES

Algorithm	IALM/EALM			DNDP-ALM			LMaFit			GoDec		
noise	0	5%	10%	0	5%	10%	0%	5%	10%	0	5%	10%
ε	0.0612	0.0646	0.1449	0.0610	0.0675	0.1349	0.2041	0.2131	0.3277	0.0602	0.0689	0.1296

among four methods in TABLE V and VI, it is still necessary to observe error distribution of system states with the purpose of finding out the effects of algorithms on all states. Following reference [23], we sort the error of separated results from large to small and plot their cumulative distribution functions in Fig.5. According to the results, we find that all algorithms have an ability in recovering the state variables from attacks. However, GoDec outperforms ALM, DNDP-ALM and LMaFit. The comparison results at other time instances are shown in the appendix, such as $t = 20$, 50, 100 and 150. In summary, four methods can explore the low-rank and sparse components in $\boldsymbol{Z_a}$, but GoDec has the best performance. It directly constrains the rank range of $\boldsymbol{Z_0}$ and cardinality range of $\boldsymbol{A}$ in optimization model as well, which leads to a higher precision of the separation results [29].

2) Location Detection Accuracy

Then, we analyze the detection accuracy of the injection location. In order to compare the performances, two indexes named true positive (TP) rate and5 false alarm (FA) rate are defined separately, as follows:

$$\mathrm{TP} = \frac{n_{sd}}{n_{attack}},\ \mathrm{FA} = \frac{n_{fr}}{n_{normal}} \tag{35}$$

where n_{attack} represents the number of locations where attackers inject data (i.e. the non-zero element in the attack matrix), n_{sd} represents the number of successful detections of injected data, n_{normal} represents the number of locations with no attack, and n_{fr} represents the number of false report of the attack-free locations.

The higher the TP of an approach, the higher the accuracy of the approach for detecting the location of false data injection; the lower the FA of a method, the less likely the method is to cause false positives to an attack. If FA is too high, the system will not be able to identify the location of the actual attack, which will affect the normal operation of the system. Under different noise level, the values of TP and FA of four approaches are provided in TABLE VII.

TABLE VII THE VALUE OF TP AND FA FOR FOUR APPROACHES

Algorithm	IALM/EALM			DNDP-ALM			LMaFit			GoDec		
noise	0	5%	10%	0	5%	10%	0	5%	10%	0	5%	10%
TP	94.37%	92.66%	89.43%	95.23%	93.58%	91.15%	98.36%	99.14%	98.14%	95.62%	93.97%	92.05%
FA	2.78%	5.41%	9.17%	1.79%	3.71%	6.03%	13.88%	20.22%	23.64%	1.42%	3.01%	4.82%

Based on the results, it can be clearly discovered that the ALM-based approaches (i.e. IALM and DNDP-ALM) and GoDec can accurately detect the locations of the false data injection attack at a lower FA value. Although LMaFit has a higher TP value, its FA value is too high. When using LMaFit, many locations where no false data injection occurs will also be detected as false data injection. Therefore, the practicality of LMaFit is further reduced.

Then we compare the ALM-based approaches with GoDec. In the case of no measurement noise, the accuracy of the three algorithms is close. When noise exists, we can find that detection accuracy of GoDec is higher and the FA value is slightly lower than DNDP-ALM under the same noise level.

Thus, combining the above discussion about numerical detection accuracy and location detection accuracy, we can conclude that GoDec has a higher computational accuracy compared with ALM and LMaFit, and it performs better than improved approach DNDP-ALM when there is noise in the measurements.

B. Computational Efficiency

In this part, the computational efficiency of the four solutions are compared and analyzed. First, the four algorithms are performed on a small measurement matrix (column $m = 100$). Then, they are tested on a series of larger matrices (column m increases from 100 to 2100 with an increment 200). The column dimensions correspond to the total number of sampling time. The corresponding computing times are recorded and showed in Fig. 6.

It is observed that 1) computing times of all four algorithms increase with the increase of the measurement matrix dimensions and 2) LMaFit and GoDec's need less than 10s for most cases, which show their capabilities for real-time applications. This is because they avoid computational demanding SVD procedures. In particular, LMaFit implements the rank estimation with the rank-revealing feature of QR factorization and GoDec factorizes the two random projections with QR decomposition. Thus, LMaFit and GoDec have similar speeds in solving the problem.

Furthermore, we find that 1) our GoDec algorithm is the most computational efficiency algorithms, 2) the IALM and DNDP-ALM algorithms have poor performances under high-dimensional measurement conditions. In both algorithms, SVD is used to solve the problem. During each iteration, SVD is used once in IALM, and is used twice in DNDP-ALM. In addition, DNDP also includes a double loop, which makes the time cost of solving the problem with the algorithm unacceptable. Specifically, IALM spends 100+ seconds when the matrix's column dimension exceeds 2000 and DNDP-ALM spends 1000+ seconds when the matrix's column dimension exceeds 1500.

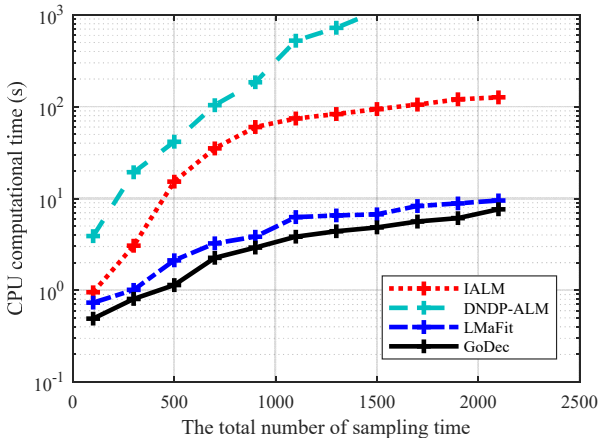


Fig. 6. CPU computation time for four algorithms

Considering the scan rate of SCADA measurements (typically few seconds or minutes), the detection results by IALM might be invalid and the EMS at the control center might have been attacked.

C. Performance Analysis vs. Attack Scales

Although the large-scale FIDA is unlike to happen in practice, it would be good to develop defense approaches that can work even in some extreme events, including intentional terrorist cyber and physical attacks scenarios.

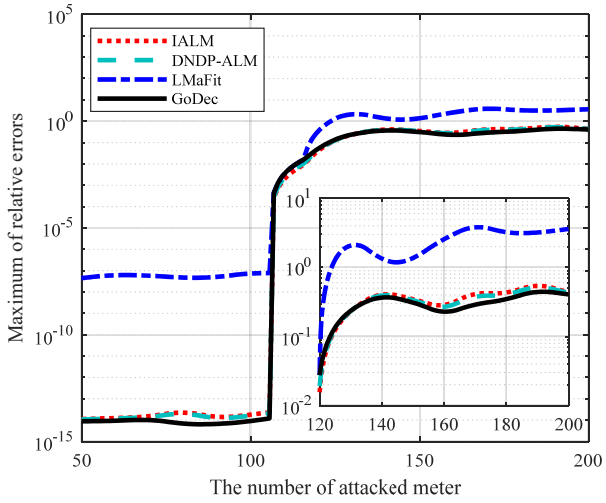
Note that [25] and [31] reveal that LMaFit becomes invalid when the sparse matrix dominates the low-rank one in magnitude. Thus, we investigate the performances of all algorithms when a large-scale attack appears. To analyze the influences on the performance of each algorithm with different scales of attacks, we define a mean relative error for state variables $\boldsymbol{\theta}$.

$$\bar{\delta} = \frac{\Sigma_n |\delta_i|}{k} \tag{33}$$

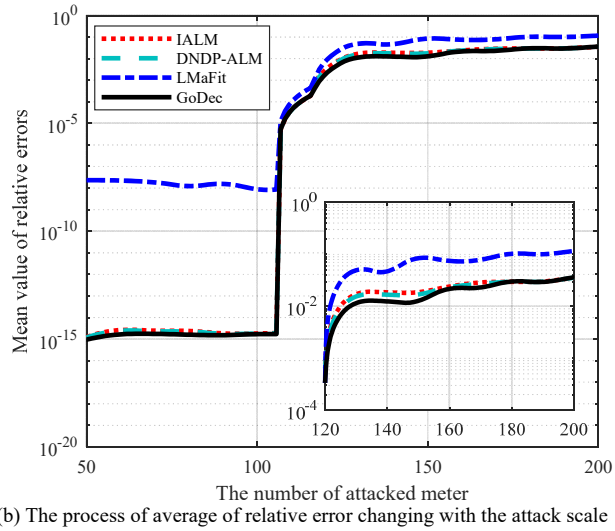
where δ_i is relative error of each voltage angle, and k is the number of attacked meters.

In process of test, k varies from 50 to 200, and increases 10 attacked meters each time. The results are shown in Fig. 5.

According to the Fig. 7, we see that when FDIA is in small-scale, ALM, DNDP-ALM and GoDec have better performances than LMaFit. With the increase of attack scale, the accuracies of all of algorithms decrease. The error of LMaFit becomes unacceptable when attack scale becomes large. In this situation, its maximum relative error δ_{max} is 402.5%. By contrast, the maximum relative error δ_{max} of IALM, DNDP-ALM and GoDec are just 57.87%, 54.69% and 42.18%, respectively. We conclude that LMaFit is unable to detect false data in large-scale, while IALM, DNDP-ALM and GoDec can achieve quite reasonable performance. In summary, GoDec performs well no matter the attack scale is large or small.



(a) The process of maximum relative error changing with the attack scale



(b) The process of average of relative error changing with the attack scale

Fig. 7. The maximum relative error and mean value of relative error with the increase of attack scale on IALM, LMaFit and GoDec

V. Conclusion

To detect FDIA in an efficient and computational attractive way, this paper proposes a new "Go Decomposition (GoDec)". We use the low rank feature of the measurement matrix and the sparsity of the attack matrix to reformulate the FDIA detection as a matrix separation problem. The latter is solved by four algorithms, namely the traditional Augmented Lagrange Multipliers (ALM), double-noise-dual-problem ALM (DNDP-ALM), the Low Rank Matrix Factorization (LMaFit) and the proposed new "Go Decomposition (GoDec)". We show that our GoDec outperforms the other three alternatives and demonstrates a much higher computational efficiency. Furthermore, GoDec is shown to be able to handle measurement noise and is applicable for large-scale attacks.

References

[1] Monticelli, A. "State Estimation in Electric Power Systems: A Generalized Approach." In *IEEE Power Engineering Review*, vol. 20, no. 4, pp. 51-53, 2002.

[2] A. Abur and A. G. Exposito, *Power System State Estimation: Theory and Implementation*. CRC Press, 2004.

[3] Liu, Yao, P. Ning, and M. K. Reiter. "False data injection attacks against state estimation in electric power grids." *ACM Conference on Computer and Communications Security, CCS 2009, Chicago, Illinois, Usa, November* DBLP, 2011:21-32.

[4] Sandberg, Henrik, A. Teixeira, and K. H. Johansson. "On Security Indices for State Estimators in Power Networks." *Preprints of the First Workshop on Secure Control Systems, CPSWEEK 2010, Stockholm, Sweden* 2010.

[5] G. Hug and J. A. Giampapa, "Vulnerability Assessment of AC State Estimation With Respect to False Data Injection Cyber-Attacks," in *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1362-1370, Sept. 2012.

[6] M. Esmalifalak, Z. Han and L. Song, "Effect of stealthy bad data injection on network congestion in market based power system," *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, Shanghai, 2012, pp. 2468-2472.

[7] L. Xie, Y. Mo and B. Sinopoli, "Integrity Data Attacks in Power Market Operations," in *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 659-666, Dec. 2011.

[8] J. Zhu, G. Zhang, T. Wang, J. Zhao, "Overview of Fraudulent Data Attack on Power System State Estimation and Defense Mechanism," in *Power System Technology*, vol. 40, no. 8, pp. 2406-2415, Aug. 2016.

[9] R. Deng, G. Xiao, R. Lu, H. Liang and A. V. Vasilakos, "False Data Injection on State Estimation in Power Systems—Attacks, Impacts, and Defense: A Survey," in *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 411-423, April 2017.

[10] O. Kosut, Liyan Jia, R. J. Thomas and Lang Tong, "Limiting false data attacks on power system state estimation," *2010 44th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, 2010, pp. 1-6.

[11] O. Kosut, L. Jia, R. J. Thomas and L. Tong, "Malicious Data Attacks on the Smart Grid," in *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 645-658, Dec. 2011.

[12] J. B. Zhao, G. X. Zhang, M. La Scala, Z. Dong, C. Chen, J. Wang, "Short-term state forecasting-aided method for detection of smart grid general false data injection attacks", *IEEE Trans. Smart Grid*, Vol. 8, no. 4, pp. 1580-1590, 2017.

[13] M. Esmalifalak; L. Liu; N. Nguyen; R. Zheng; Z. Han, "Detecting Stealthy False Data Injection Using Machine Learning in Smart Grid," in *IEEE Systems Journal*, vol.PP, no.99, pp.1-9

[14] K. C. Sou, H. Sandberg and K. H. Johansson, "Data Attack Isolation in Power Networks Using Secure Voltage Magnitude Measurements," in *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 14-28, Jan. 2014.

[15] I. Matei, J. S. Baras and V. Srinivasan, "Trust-based multi-agent filtering for increased Smart Grid security," *2012 20th Mediterranean Conference on Control & Automation (MED)*, Barcelona, 2012, pp. 716-721.

[16] G. Feng and A. Abur, "Fault Location Using Wide-Area Measurements and Sparse Estimation," in *IEEE Transactions on Power Systems*, vol. 31, no. 4, pp. 2938-2945, July 2016.

[17] A. Abur, "Use of PMUs in WLS and LAV based state estimation," *2015 IEEE Power & Energy Society General Meeting*, Denver, CO, 2015, pp. 1-1.

[18] J. B. Zhao, G. X. Zhang, R. Jabr, "Robust detection of cyber attacks on state estimators using phasor measurements", *IEEE Trans. Power Systems*, Vol. 32, no. 3, pp. 2468-2470, 2017.

[19] Giani, A, et al. "Smart Grid Data Integrity Attacks." *Smart Grid IEEE Transactions on* 4.3(2013):1244-1253.

[20] S. Bi and Y. J. Zhang, "Graphical Methods for Defense Against False-Data Injection Attacks on Power System State Estimation," in *IEEE Transactions on Smart Grid*, vol. 5, no. 3, pp. 1216-1227, May 2014.

[21] K. L. Morrow, E. Heine, K. M. Rogers, R. B. Bobba and T. J. Overbye, "Topology Perturbation for Detecting Malicious Data Injection," *2012 45th Hawaii International Conference on System Sciences*, Maui, HI, 2012, pp. 2104-2113.

[22] K. Kuntz, M. Smith, K. Wedeward and M. Collins, "Detecting, locating, & quantifying false data injections utilizing grid topology through optimized D-FACTS device placement," *2014 North American Power Symposium (NAPS)*, Pullman, WA, 2014, pp. 1-6.

[23] L. Liu, M. Esmalifalak, Q. Ding, V. A. Emesih and Z. Han, "Detecting False Data Injection Attacks on Power Grid by Sparse Optimization," in *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 612-621, March 2014.

[24] L. Liu, M. Esmalifalak and Z. Han, "Detection of false data injection in power grid exploiting low rank and sparsity," *2013 IEEE International Conference on Communications (ICC)*, Budapest, 2013, pp. 4461-4465.

[25] T. T. Kim and H. V. Poor, "Strategic Protection Against Data Injection Attacks on Power Grids," in *IEEE Transactions on Smart Grid*, vol. 2, no. 2, pp. 326-333, June 2011.

[26] Ganesh, Arvind, et al. "Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Matrices via Convex Optimization. "*Journal of the Acm* 58.3(2009):11.

[27] Anwar, Adnan, A. N. Mahmood, and M. Pickering. Data-Driven Stealthy Injection Attacks on Smart Grid with Incomplete Measurements. Intelligence and Security Informatics. 2016.

[28] Y. Shen, Z. Wen, and Y. Zhang. "Augmented Lagrangian alternating direction method for matrix separation based on low-rank factorization. "*Optimization Methods & Software* 29.2(2012):1-25.

[29] T. Zhou, and D. Tao. "GoDec: Randomized Low rank & Sparse Matrix Decomposition in Noisy Case. "*International Conference on Machine Learning, ICML 2011, Bellevue, Washington, Usa, June 28 - July* DBLP, 2011:33-40.

[30] M. Fazel, E. Candes, B. Recht and P. Parrilo, "Compressed sensing and robust recovery of low rank matrices," *2008 42nd Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, 2008, pp. 1043-1047.

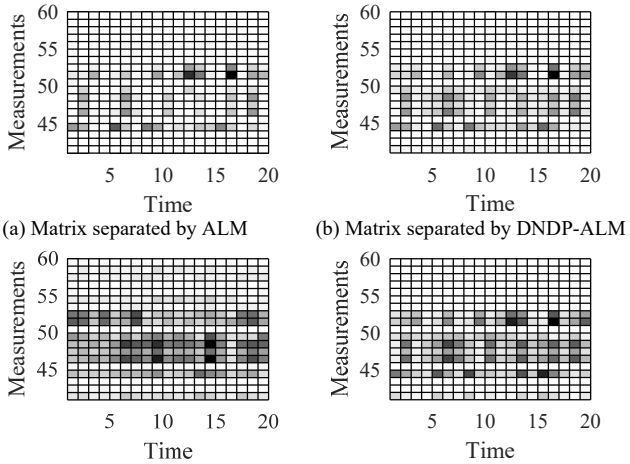
[31] A. Ganesh, J. Wright, X. Li, E. J. Candès and Y. Ma, "Dense error correction for low-rank matric34es via Principal Component Pursuit," *2010 IEEE International Symposium on Information Theory*, Austin, TX, 2010, pp. 1513-1517.

[32] D. Cheng, J. Yang, J. Wang, et al. "Double-noise-dual-problem approach to the augmented Lagrange multiplier method for robust principal component analysis." *Soft Computing*, vol. 21, no. 10, pp. 2723-2732, 2017.

APPENDIX

A. *Additional Gray Scale Image with 10% noise*

We only provide the gray scale images under 0% noise and 5% noise in Section V-A. To further illustrate the impact of noise on detection performance, we give the gray scale image for detection results with 10% measurements noise. The image is as follows:



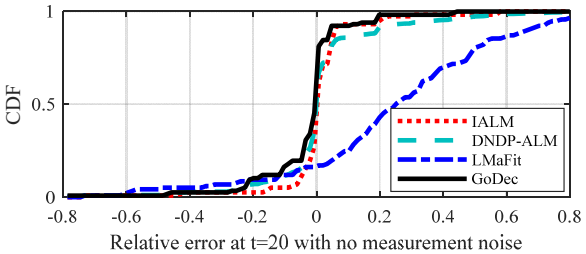
(a) Matrix separated by ALM (b) Matrix separated by DNDP-ALM

(c) Matrix separated by LMaFit (d) Matrix separated by GoDec

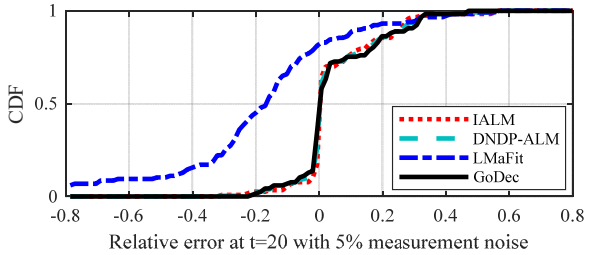
Fig. A-1. The comparison among original attack matrix and matrices separated by different algorithms (the measurements are disturbed by 10% noise)

B. *Additional CDF curve at different time instances*

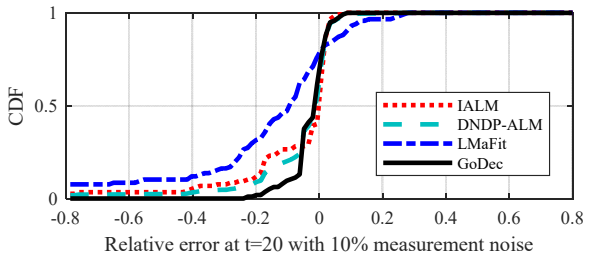
We just exhibit the error distribution at a specific time instance t=10. To give more comparison to illustrate the better performance of our method, we give the CDF curves among four methods in different time instances, such as t = 20, 50, 80, 100 and 150.The curves are as follows:



(a) CDF curve at $t = 20$ with 0% noise

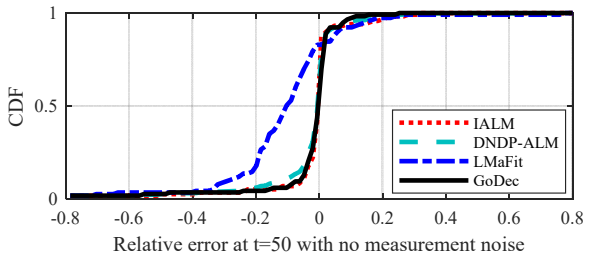


(b) CDF curve at $t = 20$ with 5% noise

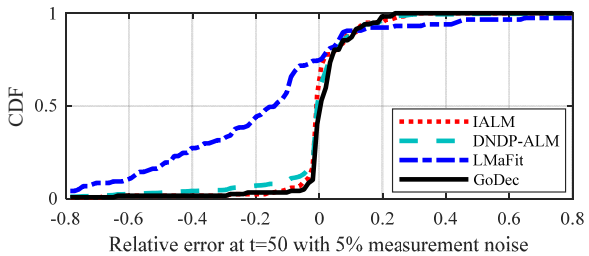


(c) CDF curve at $t = 20$ with 10% noise

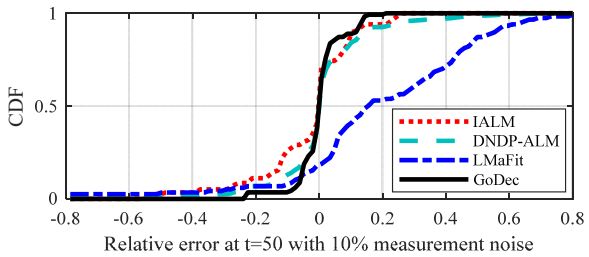
Fig.B-1. Power state reconstruction performance of four algorithms at specific time instant $t = 20$ with 0% noise, 5% noise and 10% noise.



(a) CDF curve at $t = 50$ with 0% noise

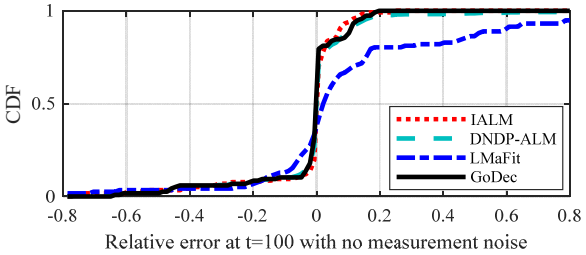


(b) CDF curve at $t = 50$ with 5% noise

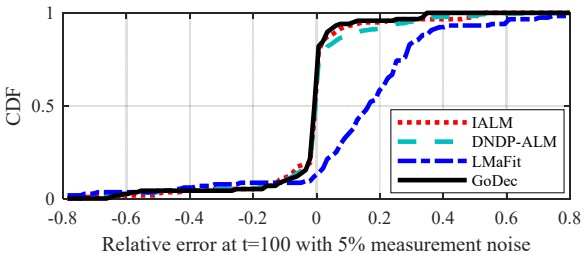


(c) CDF curve at $t = 50$ with 10% noise

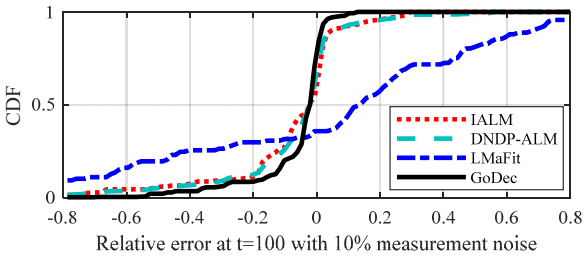
Fig. B-2. Power state reconstruction performance of four algorithms at specific time instant $t = 50$ with 0% noise, 5% noise and 10% noise.



(a) CDF curve at $t = 100$ with 0% noise

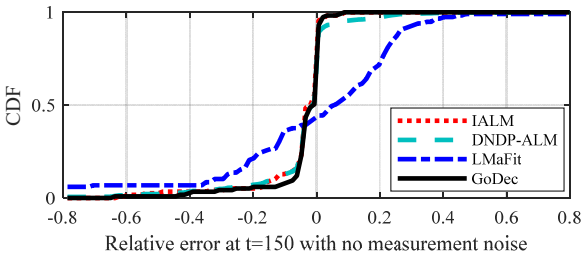


(b) CDF curve at $t = 100$ with 5% noise

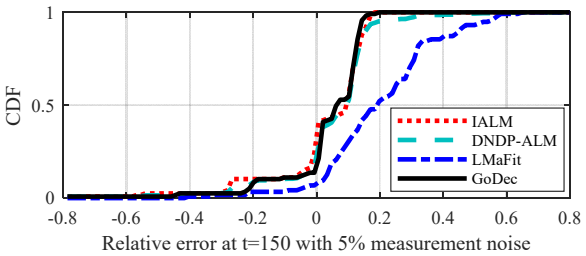


(c) CDF curve at $t = 100$ with 10% noise

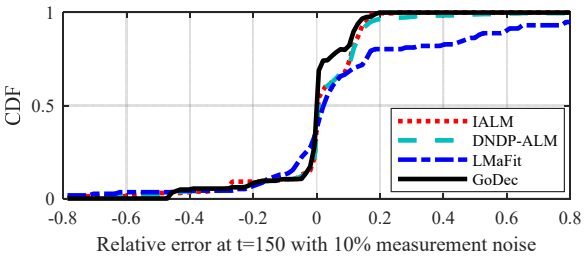
Fig.B-3. Power state reconstruction performance of three algorithms at specific time instant $t = 100$ with 0% noise, 5% noise and 10% noise.



(a) CDF curve at $t = 150$ with 0% noise



(b) CDF curve at $t = 150$ with 5% noise



(c) CDF curve at $t = 150$ with 10% noise

Fig. B-4. Power state reconstruction performance of four algorithms at specific time instant $t = 150$ with 0% noise, 5% noise and 10% noise.

Boda Li (S'17) received the B.E. degree in electrical engineering Xi'an Jiaotong University, Xi'an, China, in 2017. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Applied Electronic Technology, Tsinghua University. His research interests include cyber security of a smart grid and cyber-physical system.

Tao Ding (S'13–M'15) received the B.S.E.E. and M.S.E.E. degrees from Southeast University, Nanjing, China, in 2009 and 2012, respectively, and the Ph.D. degree from Tsinghua University, Beijing, China, in 2015. During 2013 and 2014, he was a Visiting Scholar in the Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN, USA. He is currently an Associate Professor in the State Key Laboratory of Electrical Insulation and Power Equipment, the School of Electrical Engineering, Xi'an Jiaotong University. His current research interests include electricity markets, power system economics and optimization methods, and power system planning and reliability evaluation. He has published more than 100 technical papers and authored by "Springer Theses" recognizing outstanding Ph.D. research around the world and across the physical sciences—*Power System Operation with Large Scale Stochastic Wind Power Integration*. He received the excellent master and doctoral dissertation from Southeast University and Tsinghua University, respectively, and Outstanding Graduate Award of Beijing City. Dr. Ding is an Editor of CSEE Journal of Power and Energy Systems, Energies, and a Guest Editor of IEEE Transactions on Sustainable energy, Applied Sciences.

Can Huang (GS'13–M'16) received the B.S.E.E degree from Hohai university, Nanjing, China, in 2008, the M.S.E.E. degree from Southeast University, Nanjing, in 2011, and the Ph.D. degree from The University of Tennessee, Knoxville, TN, USA, in 2016. From 2011 to 2012, he was with the State Grid Electric Power Research Institute (NARI Group Corporation), Nanjing. He is currently a Postdoctoral Research Staff Member with the Lawrence Livermore National Laboratory, Livermore, CA, USA. His current research interests include advanced control, computing, and communication for power and energy systems.

Junbo Zhao (M'18) is an assistant professor (research)at Virginia Tech. He received the Ph.D. degree from Bradley Department of Electrical and Computer Engineering at Virginia Tech in 2018. He did the summer internship at Pacific Northwest National Laboratory from May-August, 2017. He is now the chair of IEEE Task Force on Power System Dynamic State and Parameter Estimation, the secretary of the IEEE Working Group on State Estimation Algorithms and the IEEE Task Force on Synchrophasor Applications in Power System Operation and Control. He has written 2 book chapters, published more than 40 peer-reviewed journal and conference papers. His research interests lie in power system realtime monitoring, operations and cyber security that include state estimation, dynamics and stability, cyber attacks and countermeasures, big data analytics and robust statistics with applications in the smart grid. He serves as the Associate Editor of International Journal of Electrical Power & Energy Systems and IET Generation, Transmission & Distribution.

Yongheng Yang (S'11-M'15-SM'17) received the B.Eng. degree in electrical engineering and automation from Northwestern Polytechnical University, Shaanxi, China, in 2009 and the Ph.D. degree in electrical engineering from Aalborg University, Aalborg, Denmark, in 2014. He was a postgraduate student at Southeast University, China, from 2009 to 2011. In 2013, he was a Visiting Scholar at Texas A&M University, USA. Dr. Yang has been with the Department of Energy Technology, Aalborg University since 2014, first as a Postdoc researcher, then an Assistant Professor, and now an Associate Professor. He has been focusing on grid integration of renewable energies, power electronic converter design, analysis and control, and reliability in power electronics. Dr. Yang served as a Guest Associate Editor of IEEE Journal of Emerging and Selected Topics in Power Electronics and a Guest Editor of Applied Sciences. He is an Associate Editor of CPSS Transactions on Power Electronics and Applications.

Ying Chen (M'06) received his B.E. and Ph.D. degrees in electrical engineering from Tsinghua University, Beijing, China, in 2001 and 2006, respectively. He is currently an Associate Professor in the Department of Electrical Engineering and Applied Electronic Technology, Tsinghua University. His research interests include parallel and distributed computing, electromagnetic transient simulation, cyber-physical system modeling, and cyber security of smart grid.