# CONGO$^2$: Scalable Online Anomaly Detection and Localization in Power Electronics Networks

Jun Yu, Huimin Cheng, Jinan Zhang, Qi Li, Shushan Wu, Wenxuan Zhong, Jin Ye, WenZhan Song, Ping Ma

***Abstract*—Rapid and accurate detection and localization of electronic disturbances simultaneously are important for preventing its potential damages and determining potential remedies. Existing anomaly detection methods are severely limited by the low accuracy, the expensive computational cost and the need for highly trained personnel. There is an urgent need for a scalable online algorithm for in-field analysis of large-scale power electronics networks.**

**In this paper, we propose a fast and accurate algorithm for anomaly detection and localization of power electronics networks: stratified colored-node graph (CONGO$^2$). This algorithm hierarchically models the change of correlated waveforms and then correlated sensors using the colored-node graph. By aggregating the change of each sensor with its neighbors' inputs, we can spontaneously identify and localize the anomaly that cannot be detected by data collected from a single sensor.**

**As our proposed method only focuses on the changes within a short time frame, it is highly computational efficient and only needs small data storage. Thus, our method is ideal for online and reliable anomaly detection and localization of large-scale power electronic networks. Compared to existing anomaly detection methods, our method is entirely data-driven without training data, highly accurate and reliable for wide-spectrum anomalies detection, and more importantly, capable of both detection and localization. Thus, it is ideal for in-field deployment for large-scale power electronic networks. As illustrated by a distributed energy resources (DERs) power grid with 37-node, our method can effectively detect and localize various cyber and physical attacks.**

***Index Terms*—anomaly detection, anomaly localization, graph model**

## I. Introduction

POWER electronics are the building blocks of critical infrastructures, such as data centers, hospitals, and manufacturing systems. Unexpected power quality anomalies or disturbances incited by system faults or cyber attacks could cause exacerbated system-level unbalanced conditions, voltage sags, and harmonics that worsen device-level anomalies [1], [2], [3]. As illustrated in the mock example (Fig. 1), the attacker's malicious modification of controllers in power electronics converters would degrade both power electronics and the grid, leading to catastrophic failures and substantial economic losses. An accurate and timely anomaly detection and localization method that can protect the system is highly desirable to trigger the in-field implementation of the countermeasures.

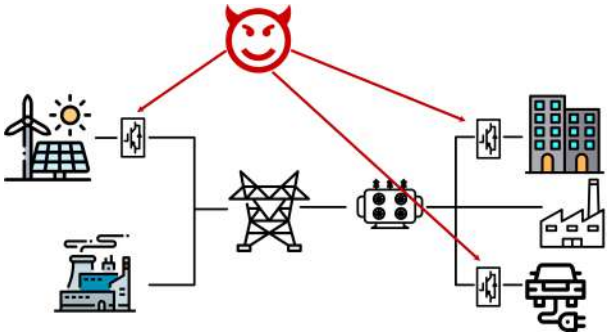

Fig. 1. Attacks threaten the security of the power electronics network.

For over a decade, physical-model-based methods have been one of the most popular tools in anomaly detection. However, despite many successful applications such as detecting attacks in the transmission grid [4], distribution grid [5] and DC microgrids [6], physical-model-based methods have some inherent limitations, such as the requirement of highly subjective domain-specific input and rigid physical assumptions in solving dynamic closed-loop systems [7], [8]. More seriously, most physical-model-based methods can only detect strong anomalies that affect the whole system and fail to detect subtle anomalies such as anomalous power electronics converters.

With the rapid development of high-throughput computing techniques and artificial intelligence, the computational cost of the large dynamic system has been dramatically reduced, offering the great possibility for data-driven methods in smart grid applications. Extensive studies have been published using data-driven approaches, ranging from supervised learning to unsupervised learning. For supervised learning, historical data are used to train a set of features that are predictive for system anomaly [9], [10], [11], [12]. Though the supervised learning methods are fairly effective in detecting power grid anomaly, they are severely limited by the cost to obtain gigantic and accurate training data sets, which is highly labor-intensive and time-consuming [13]. Moreover, the training set and testing set might be highly heterogeneous for wide-spectrum

This research is partially supported by U.S. National Science Foundation under grants DMS-1903226, DMS-1925066, DMS-2124493, SaTC-2019311 and ECCS-1946057, the U.S. National Institute of Health under grant R01GM122080, the U.S. Department of Energy DOE-EE0009026, the U.S. Department of Defense DOD-FA8571-21-C-0020, National Science Foundation of China under grants 12001042 and Beijing Institute of Technology research fund program for young scholars.

Jun Yu is with School of Mathematics and Statistics, and key laboratory of mathematical theory and computation in information security, Beijing Institute of Technology. yujunbeta@bit.edu.cn

Huimin Cheng, Shushan Wu, Wenxuan Zhong, and Ping Ma are with Department of Statistics, University of Georgia. huimincheng@uga.edu, shushanwu@uga.edu, wenxuan@uga.edu, pingma@uga.edu

Jinan Zhang, Qi Li, Jin Ye, WenZhan Song are with College of Engineering, University of Georgia. jinan.zhang@uga.edu, qi.li@uga.edu, jin.ye@uga.edu, wsong@uga.edu

attacks, which hinders the method's broad in-field application. Unsupervised learning, on the other hand, does not require the training set. They can automatically detect the disturbances if the disturbances change the streaming waveforms (six-dimensional time-series: three-phase voltages and three-phase currents) that are collected by each sensor in power grids. Although this approach provides a great promise to detect and even localize the anomaly in the power grid, the delivery of this promise has not yet been fully materialized, because there is a lack of effective and efficient computational algorithm for handling these large-scale networks. Existing methods mainly rely on change-point detection algorithms such as CUSUM and Hotelling $T^2$ [14], [15], which either assume that the time-series/waveforms are entirely independent or connected by unknown relationships that need to be estimated. The independence assumption implies that all the sensor are disconnected and that the six waveforms collected by each sensor are independent. Thus, a change-point detection method can be applied to each waveform individually to identify and even localize the sensor that has been attacked [16]. Despite the simplicity and accessibility, the anomaly detection methods based on independence assumptions can be very inaccurate for power grid as they ignore the connectivity between sensors and the physical dependence between the voltages and the currents. An alternative approach is to assume an unknown relationship between different waveforms[8]. This approach, however, is highly susceptible to the curse of dimensionality, which refers to various difficulties a large number of nodes can cause to parameter estimation and computation. Thus, it cannot be used for large-scale power grids.

Following the unsupervised anomaly detection approaches, in this article, we proposed a colored-node graph (CONGO) algorithm to integrate the sensor connectivity information and voltages-currents dependent information for a large-scale power grid. We first build a six-node network at each time point to incorporate the three-phase voltages and three-phase currents dependence and mark the waveform change of each node in a particular time frame by different colors. This is our physical-law-based CONGO. The wave change in the physical-law-based CONGO can be quantified using a Phase Change (PC) score, whose value is calculated by aggregating the changes of the nodes' value and edges' alteration. The score essentially measures the graphs' distance on Krylov subspace. We then build a power-grid-based CONGO using the sensor connectivity and mark the color of each sensor in a given time frame by their PC score. As our method uses the physical-law-based CONGO within the power-grid based-CONGO, we referred to our method as CONGO$^2$. The CONGO$^2$ provides a rich and flexible framework to address the limitations of anomaly detection in the power grid. By incorporating the connectivity and dependent information between sensors and different waveforms, our method effectively alleviated the problems of all the existing data-driven approaches. It is worth noting that our method is highly computational efficient and only uses data within a given time frame. Thus, it is a truly scalable online algorithm for a short time frame.

In addition to detection, our method can simultaneously localize the anomalies, a key feature triggering a protection plan and providing timely guidance for reparation. Existing works for anomaly localization was focused on localizing harmonic sources that are mainly generated by power electronics converters [17], [18]. This approach needs manually manipulate the power electronic converters to generate different harmonic sources. A data-driven approach is still lacking. Motivated by the stochastic nearest neighborhood method [19], [20], eigen-equation compression method [21], PCA-based method [22], and k-subgraph partition method [23] proposed to localize the anomaly of a node in a network, we proposed modifying each sensor's PC score by leveraging its neighbors' PC scores. Compared to a single sensor's PC score, the new score is more sensitive to the subtle attacks that cannot be detected by a single sensor, as it aggregates the signals by integrating its neighbours that are also affected.

The contributions and innovations of our work are summarized as follows.

1) To the best of our knowledge, CONGO$^2$ is the first unsupervised graph-based learning framework for anomaly detection and localization in power networks in real-time.
2) CONGO$^2$ builds graphs based on Krylov subspace distance, which is theoretically guaranteed to separate the high-frequency disturbances from the actual signal and thus reduces the chance of false alarms. Moreover, it considers the topological information of the graph. Thus, CONGO$^2$ can determine the anomaly location with high accuracy.
3) The PC score defined in Section III-D can be easily computed, and anomaly detection is based on relative changes between the current PC score and its exponential moving average. Thus it enables real-time anomaly detection. Furthermore, only the local topological information of the power grid is required. Thus CONGO$^2$ is scalable to large-scale power grid analytics, as stated in Section III-E.
4) Experiments and evaluations were conducted in a system of 37-node power grids with DERs under different cyber and physical attack scenarios. The state-of-the-art performance validates the effectiveness of CONGO$^2$.

## II. Algorithm Design

In this section, we will first introduce the CONGO$^2$ method which provides a new tool for visualizing and analyzing the power grids. We will then show how to detect and localize the anomalies using CONGO$^2$.

### A. Problem formulation

Let $\vec{\mathbf{x}}_1(t), \ldots, \vec{\mathbf{x}}_m(t)$ be observed waveform data for $m$ sensors at time $t$, where $\vec{\mathbf{x}}_i(t) = (x_{i1}(t), x_{i2}(t), \ldots, x_{i6}(t))$ is a six-dimensional vector recording the three-phase currents and three-phase voltages for sensor $i$, where $i = 1, \ldots, m$, $t = 1, \ldots, T$, and $T$ is the total number of time points. When the power grid is normally working, the observed waveforms of all sensors show stable waveform patterns. However, once an anomaly occurs, the observed waveforms of different sensors may deviate from their stable normal pattern.

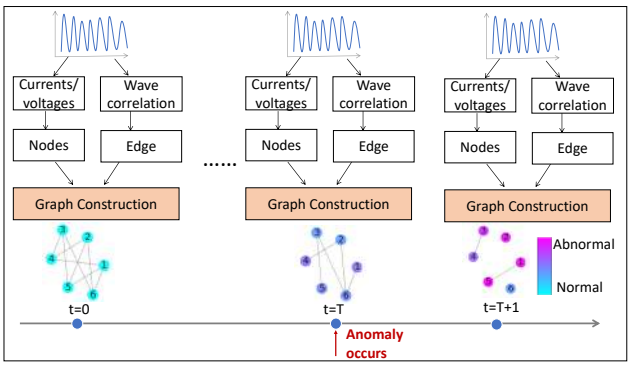


Fig. 2. Flowchart of CONGO. At each time $t$, we construct a colored-node graph with six nodes, where nodes 1-3 represent three-phase currents, and nodes 4-6 represent three-phase voltages.

In this paper, we aim to detect an anomaly as soon as it occurs and identify the anomaly location as accurately as possible in the power grid. The key to the success of anomaly detection is to "raise a flag" as soon as a waveform of a senor deviates from its normal pattern.

### B. CONGO of one sensor

In the normal situation, the transmissions, loads, generators, and the grid network decide both voltage and current. However, when there is an anomaly, e.g., a sensor feedback attack occurs in DER, the controller generates a wrong voltage reference. The voltage then deviates from its normal pattern. This deviation may lead to a high harmonics current appearance. Therefore, the performance of the voltages and currents in the whole grid may abruptly change when attacks occur.

To model the waveform data of a sensor in a three-phase electric power grid, we first model the six-dimensional waveform data of each sensor with a dynamic colored-node graph (CONGO), which has been shown to be a powerful tool to study complex systems [24], [25]. Fig. 2 shows the workflow of CONGO. Specifically, at each time point $t$ of sensor $i$, we construct a graph $G_i(t)$, consisting of six nodes representing the three-phase currents (nodes 1-3) and the three-phase voltages (nodes 4-6). The color of a node $j$ $(j = 1, \ldots, 6)$ in sensor $i$ $(i = 1, \ldots, m)$ is determined by the score $b_{ij}(t)$, which is defined as the squared Krylov subspace distance between two consecutive $x_{ij}(t)$ and $x_{ij}(t-1)$. We employ a cool-warm color-coding system where the cool color (i.e., blue) represents small change (i.e., small $b_{ij}(t)$ ) whereas warm color (i.e., red) indicates large change (i.e., large $b_{ij}(t)$). In graph $G_i(t)$, we draw a weighted edge between currents and voltages. The edge weight between two nodes $j$ and $l$ is $w_{jl}(i,t)$, which is defined as $\exp(-Kry^2(x_{ij}(t), x_{il}(t)))$, where $Kry$ is the Krylov subspace distance, $j = 1, 2, 3; l = 4, 5, 6$.

We now present the details of how we employ the Krylov subspace distance [26] to calculate $b_{ij}(t)$. We define the trajectory matrix $\boldsymbol{X}_{ij}(t)$ for $h$ consecutive data points $x_{ij}(t-h), \ldots, x_{ij}(t)$ as follows,

$$\begin{bmatrix} x_{ij}(t-h) & x_{ij}(t-h+1) & \ldots & x_{ij}(t-h+k-1) \\ x_{ij}(t-h+1) & x_{ij}(t-h+2) & \ldots & x_{ij}(t-h+k) \\ \vdots & \vdots & & \vdots \\ x_{ij}(t-h+k'-1) & x_{ij}(t-h+k') & \ldots & x_{ij}(t) \end{bmatrix}, \quad (1)$$

where $k'$ and $k$ are the pre-specified number of rows and columns, respectively, and $h = k + k' - 2$ is the window size. Analogously, we define the trajectory matrix $\boldsymbol{X}_{ij}(t-1)$ at time $t-1$. Let $\mathcal{X}_{ij}(t-1)$ and $\mathcal{X}_{ij}(t)$ be the row spaces spanned by $\boldsymbol{X}_{ij}(t-1)$ and $\boldsymbol{X}_{ij}(t)$. Let $b_{ij}(t)$ be the squared Krylov distance between two subspaces $\mathcal{X}_{ij}(t-1)$ and $\mathcal{X}_{ij}(t)$,

$$b_{ij}(t) = Kry^2(x_{ij}(t), x_{ij}(t-1)) = \min_{\substack{\|\boldsymbol{s}\|=1, \\ \boldsymbol{s} \in \mathcal{X}_{ij}(t-1)}} \|(P_1 - P_2)\boldsymbol{s}\|^2, \quad (2)$$

where $P_1$ and $P_2$ are the projection operators onto $\mathcal{X}_{ij}(t-1)$ and $\mathcal{X}_{ij}(t)$, respectively. Analogously, we apply the Krylov subspace distance to calculate the edge weight $w_{jl}(i,t)$ between node $j$ and node $l$ in graph $G_i(t)$. In this paper, we opt to use the Krylov distance because the Krylov subspace is theoretically guaranteed to separate the high-frequency disturbances from the actual signal and thus reduces the chance of false alarms [27].

The CONGO provides a friendly visualization method to examine the AC circuits condition by considering both the waveform of each sensor and the interactions between sensors. A simple example of CONGO is shown in Fig. 3. As we can see, the color of nodes 1-3 turns red at $t = 0.21$ because the waveform of nodes 1-3 (i.e., the red/blue/green current curves) have dramatic changes when evolving from 0.20 s to 0.21 s. The color of nodes 4-6 turns dark blue at $t = 0.21$ s because the waveform of nodes 4-6 (i.e., the red/blue/green voltage curves) has slight changes. In this example, for visualization purposes, we delete edges with weights less than 0.8. It is observed that all edges disappear at $t = 0.21$ s, indicating that the relationship between currents and voltages has dramatic change. All aforementioned observations suggest that an anomaly occurs between 0.20 s and 0.21 s with high probability.

### C. $CONGO^2$ of multiple sensors in the power grid

To model the relationship of waveforms of multiple sensors in the power grid, we develop the colored-node graph square model (CONGO$^2$), which applies CONGO twice. In the CONGO$^2$, we construct an undirected colored-node graph $G^2(t)$ at each time $t, t = 1, \ldots, T$. Each node in $G^2(t)$ represents a sensor. The color of sensor $i$ of $G^2(t)$ is determined by the distance $dis_i(t)$ between $G_i(t)$ and $G_i(t-1)$. We define $dis_i(t)$ as follows,

$$dis_i(t) = 0.5||B_i(t) - B_i(t-1)||_2^2 + 0.5||A_i(t) - A_i(t-1)||_F^2 \quad (3)$$

where $B_i(t) = (b_{i1}(t), \ldots, b_{i6}(t))$ is the vector of the scores of nodes in graph $G_i(t)$, $A_i(t)$ is the adjacency matrix of graph $G_i(t)$, $||\cdot||_2$ is $L_2$ distance, and $||\cdot||_F$ is matrix Frobenius norm. We again employ a cool-warm color-coding system where the cool color (i.e., blue) represents small changes (i.e., small $dis_i(t)$) whereas warm color (i.e., red) indicates large changes (i.e., large $dis_i(t)$).

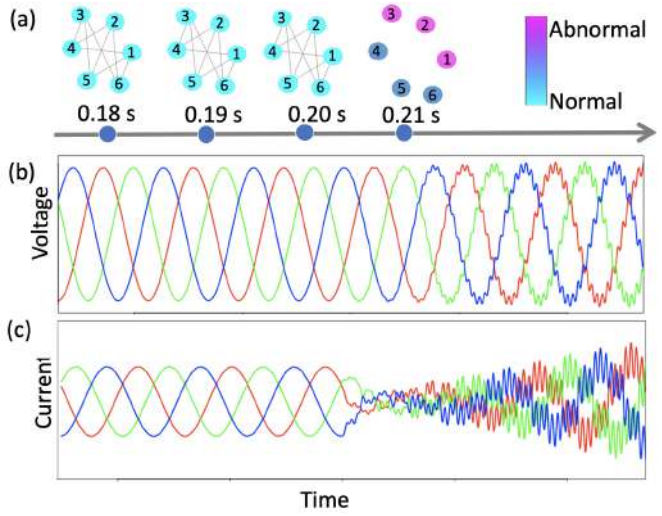


Fig. 3. An example of CONGO visualization. (a) is the graph representation of the signals at time $t$. A visualization of waveforms of the voltage and the current are given in (b) and (c). In this example, the anomaly occurs at 0.20 s.

To construct edges of $G^2(t)$, we first leverage the topological information of the smart grid to decide whether we draw an edge between two sensors. In particular, we draw an edge between sensor $i$ and sensor $j$ in $G^2(t)$ if and only if these two sensors satisfy the following three conditions. (i) They are connected in the power grid. (ii) They are adjacent, i.e., there is no other sensor placed between $G_i(t)$ and $G_j(t)$. (iii) They should not lie in different branches. To better illustrate the aforementioned three conditions, we take the simulated IEEE 37-node distributed power grid in Section III-B as an example. The topology of this 37-node power grid is shown in Fig. 8 (a). In Fig. 8 (a), we observe that sensor 01 and sensor 02 satisfy conditions (i) and (ii), but they do not satisfy conditions (iii) since they lie in different branches. Thus we do not draw an edge between sensor 01 and sensor 02.

If there is an edge between sensor $i$ and $j$, we assign a weight $S_{ij}(t)$ to the edge connecting $i$ and $j$ in $G^2(t)$. In this paper, we define $S_{ij}(t)$ as

$$S_{ij}(t) = \exp\{-0.5||B_i(t) - B_j(t)||_2^2 - 0.5||A_i(t) - A_j(t)||_F^2\}, \quad (4)$$

where the exponential transformation is used to make the score between zero to one. Note that $S_{ij}(t) = 0$ if there is no edge between sensor $i$ and $j$. Intuitively, $S_{ij}(t)$ measures the similarity between AC circuits conditions of sensor $i$ and $j$. Over time, the edge weight between two sensors is stable when the power grid works normally. When an anomaly occurs, the edge weight of sensors located close to the anomaly is significantly different from the edge weight of other sensors located away from the anomaly. We employ the developed CONGO$^2$ to localize the anomaly, as we will show in Section II-D.

### *D. Anomaly detection and localization using CONGO$^2$*

We first present how to employ CONGO$^2$ for anomaly detection. In particular, we propose the following phase change score (abbreviated as PC),

$$PC(t) := m^{-1} \sum_{i=1}^{m} \{0.5 dis_i(t) + 0.5 \Delta deg_i(t)\}, \quad (5)$$

where $dis_i(t)$ is defined in Eq. (3), $\Delta deg_i(t) = |\sum_{S_{ij}(t)\neq 0} S_{ij}(t) - \sum_{S_{ij}(t)\neq 0} S_{ij}(t-1)|/D_i$, here $D_i$ is the number of edges connecting to node $i$. The first term $dis_i(t)$ measures the difference between $G_i(t)$ and $G_i(t-1)$ of sensor $i$. The second term $\Delta deg_i(t)$ measures the weight difference of all edges connecting sensor $i$ between $t$ and $t-1$.

Apparently, the system has phase change if there is an abrupt change in PC. To determine whether a change is abrupt, we recommend using the relative changes between the current $PC(t)$ and its exponential moving average $\mathrm{EMAPC}(t)$. If $PC(t)/\mathrm{EMAPC}(t)$ is greater than a prespecified threshold, we conclude that the system has phase change. Here the $\mathrm{EMAPC}(t)$ is recursively defined as follows,

$$\mathrm{EMAPC}(t) = \begin{cases} PC(1), & t = 1 \\ 0.2PC(t) + 0.8\mathrm{EMAPC}(t-1), & t > 1. \end{cases}$$

The above coefficients 0.2 and 0.8 are commonly used in the statistical analysis [28].

We then present how to employ CONGO$^2$ for anomaly localization. In particular, we define the contribution of sensor $i$ at time $t$ as

$$cPC_i(t) = \{0.5 dis_i(t) + 0.5 \Delta deg_i(t)\}/(m \cdot PC(t)), \quad (6)$$

where $PC(t)$ is defined in Eq. (5). Note that when the system works under normal conditions, all sensors usually contribute equally to PC. However, when an anomaly occurs to the system, the sensor close to the anomaly will contribute more than others. Therefore, we localize the anomaly by observing the contribution of each sensor.

Specifically, we first identify candidate anomaly sensors whose $cPC_i(t)$ are higher than $\rho/m$, where $\rho(> 1)$ is a pre-specified parameter. After finding the candidate anomaly sensors, we propose to localize the anomaly through the following criteria.

- **Criterion 1**. If the candidate anomaly sensors cover half of the branches, the anomaly is localized at the trunk, e.g., the red line in Fig. 8 (a). One example of such anomaly is the three-phase short circuit.
- **Criterion 2**. If all the candidate anomaly sensors are in the same branch, the anomaly is localized at this branch.
- **Criterion 3**. Suppose sensors $i$ and $j$ are two candidate anomaly sensors with the highest $cPC(t)$. If $cPC_i(t)/cPC_j(t) \leq 1.5$, and sensor $i$ lies in the trunk and sensor $j$ lies in one branch, the anomaly is located at the branch of $j$. Criterion 3 is reasonable since the trunk sensor usually receives more influence than the sensors placed in the branches. For example, in Fig. 8 (a), the sensor PCC is connected to a power grid. When the anomaly happens, sensor PCC also receives feedback from the power grid.
- **Criterion 4**. If all of the aforementioned criteria are not satisfied, we rank the candidate anomaly sensors in the decreasing order of the $cPC_i(t)$. The anomaly is

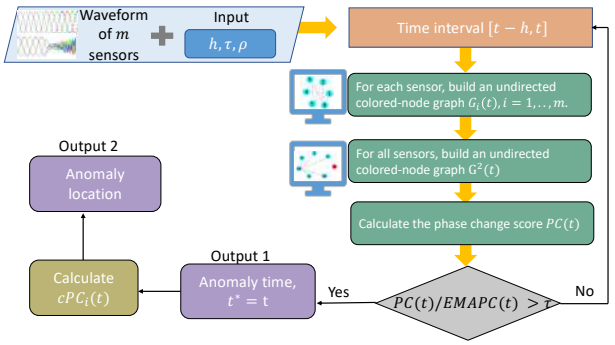


Fig. 4. Flowchart of CONGO$^2$ Algorithm. The inputs include the waveform data of $m$ sensors, the length of the time interval $h$, and the predefined parameters $\tau$ and $\rho$ ($\rho > 1$ ). CONGO$^2$ outputs when the anomaly occurs and where the anomaly occurs.

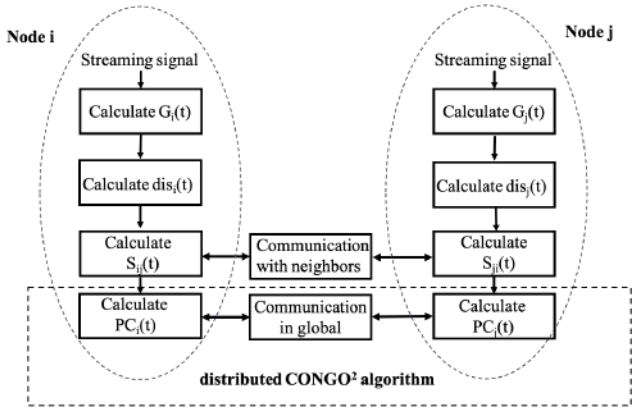


Fig. 5. Distributed implementation design

localized by sequentially checking these sensors.

Our proposed CONGO$^2$ algorithm for anomaly detection and localization is summarized in Fig. 4. It is imperative to note that the detection and localization are based on the relative changes in the spatial and temporal domain. Thus, our method is unsupervised and easy to be applied in online monitoring systems. We further present the computational cost of CONGO$^2$. Note that the calculation of $PC$ score involves the calculation of $\sum_{i=1}^{m} dis_i(t)$ and $\sum_{i=1}^{m} \Delta deg_i(t)$. Since calculating $\sum_{i=1}^{m} dis_i(t)$ requires $O(|V|)$ computational cost, and calculating $\sum_{i=1}^{m} \Delta deg_i(t)$ requires $O(|E|)$ computational cost, the computational cost of CONGO$^2$ is $O(|V|)+O(|E|)$, where $|V|$ is the number of nodes in $G^2(0)$, $|E|$ is the number of edges in $G^2(0)$. Note that $|V|$ is the number of sensors, which is the aforementioned $m$, and $|E|$ depends on the smart grid topology. When $G^2(0)$ is a sparse network, i.e., $|E|$ is of the order $O(|V|)$, the computational cost is only $O(|V|)$.

### *E. Distributed implementation of CONGO$^2$*

In this section, we show the distributed implementation of CONGO$^2$. Fig. 5 shows the design of distributed implementation. In particular, we take node $i$ as an example. First, we calculate the $dis_i(t)$ based on $G_i(t)$ locally. Second, to calculate the edge weight $S_{ij}(t)$ between node $i$ and its neighbor $j$, we transfer $A_i(t)$ and $B_i(t)$ between them accordingly. The transferred data could be compressed into a user datagram protocol (UDP) package with timestamp and broadcast to its neighbors. Lastly, for event detection and localization, the phase change score of the whole network $PC(t)$ and sensor's contribution $cPC_i(t)$ are needed, as defined in Eq. (5) and Eq. (6). Fig. 5 summarizes the distributed implementation design. Since the message size in each communication in the second layer graph is constant, and our method only requires each sensor to communicate with one-hop neighbors only, thus the communication cost (in bits) with neighbors is $O(|E|)$. The communication cost in global is $O(|V|)$. Therefore, the communication cost of our method is $O(|V|)+O(|E|)$, which is scalable for large smart grids.

To further reduce the communication cost, we also design a decentralized framework. As shown in Fig. 6(a), the proposed algorithm aggregates the information and manages cooperation between nodes. The compressed information of the node and its neighbor communicate in a tree structure. We choose the tree-based aggregation technique since it is faster to reach a consensus on a real-time system [29]. Our proposed decentralized framework works in the following steps. First, we broadcast node $i$'s information $G_i(t)$ to its neighbor node $j$, and calculate the $S_{ij}(t)$ for the edge between sensor $i$ and $j$ in $G^2(t)$. Second, we compute the $PC(t)$ in root node by aggregating $PC(t)$ of subtree in a bottom-up fashion in the spanning tree. Third, $PC(t)$ will be passed from the root node to the leaf nodes, and every node $i$ can calculate its $cPC_i(t)$. Finally, the root node collects all $cPC_i(t)$ for the event localization. Fig. 6 illustrates how each node communicates with neighbors and the communication in the whole network.

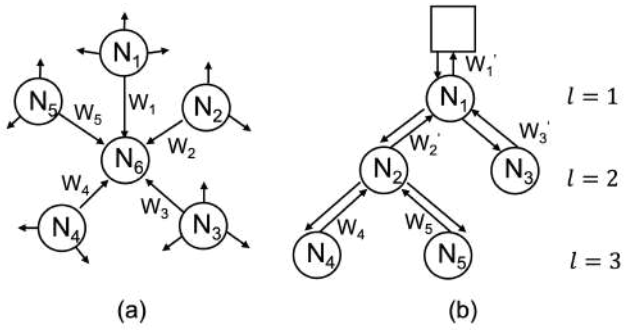


Fig. 6. Illustration of two ways of node communication in distributed design. The arrow indicates the direction of communication. (a) shows that the node broadcasts the information with its neighborhood. (b) shows the communication in global, where the square box stands for the root of the tree and $l$ stands for tree's level.

## III. Empirical Evaluations

To assess the performance of the proposed method, we carry out extensive empirical evaluations. In what follows, we will first present the threat model we use to generate simulated data and the experiment setup. We then present the results of our methods, and the comparison results with other existing methods.

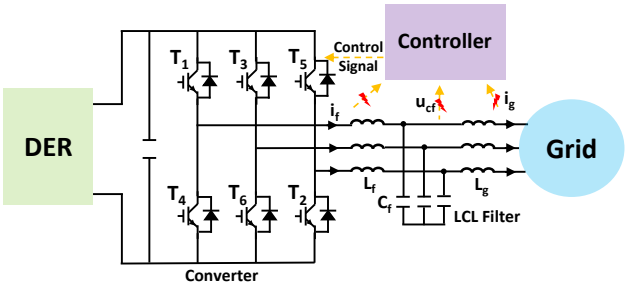


Fig. 7. Diagram of the grid-connected DER.

### A. Threat Model

With the emerging of IoT technology, power electronics networks evolve into more intelligent systems with high efficiency and expose more vulnerabilities to cyber attacks. Many studies for cyber physical security in power networks have been conducted. In [30], the authors summarize the assessment, detection, and mitigation methodology for cyber attacks in Photovoltaic (PV) farms. Cyber attacks in PV farms are discussed extensively, including firmware attacks, software attacks, and network attacks. Besides, attackers could design cyber attacks in the smart meter, state estimation, and control center using communication systems in the power electronics networks. For smart meter security, attackers could change the meter measurement to get monetary gains. Cyber attacks in the state estimation and control center can lead to the instability of the power electronics system by tempering the control signal. For example, [31] introduces false data injection attacks (FDIAs) that falsify the control signal of the energy management system (EMS) in the microgrid. In [32], FDIAs and DoS attacks are demonstrated in a Hardware-in-the-loop testbed, and their impact is analyzed. Many methodologies are proposed to address the cyber security in power electronics networks through information technology. In [33], a Software-Defined Networking-based architecture is developed to protect the microgrid operation from cyber attacks. Blockchain-based technology for cyber attacks defense in power electronics devices is discussed and explored [34]. Compared to the cyber attack compromising the software or communication link, there is a growing concern that sensor attacks pose on the power electronics operation. This sensor attack could falsify the measurement of the controller in power electronics devices. In [35], the authors demonstrate a noninvasive sensor attack in a power electronics converter. Although many researchers have started studying sensor attacks, they cannot mitigate them by using information-based technologies. Thus, we proposed a new detection and localization method for the sensor attack in power electronics networks.

The topology of a typically distributed energy resource (DER) is shown in Fig. 7. We use the converter as a common interface to transfer DC power in DER to the AC grid to generate the power grid. The LCL filter, including $L_f$, $C_f$, and $L_g$, is designed to eliminate the harmonics in voltages and currents. Generally, the PI controller is employed to drive switches (T1-T6) to achieve power conversion. The measurement $i_f$, $u_{cf}$, and $i_g$ works as the feedback, which affects the performance of the controller. The measurement can be denoted as

$$Y_0(t) = [i_f, u_{cf}, i_g]^T \tag{7}$$

As sensor attack compromises the measurement data in the converter, the data integrity in the DER controller is destroyed. Therefore, the sensor attack is defined as a data integrity attack (DIA) as follows,

$$Y_F(t) = \alpha Y_0(t) + \beta \tag{8}$$

where $Y_F$ is the compromised measurement that is the controller's input; $Y_0$ is the actual measurement; $\alpha$ is a multiplicative factor matrix that defines the weight of the attack; $\beta$ is the malicious modification of the signals.

### B. Experiment Setups

An IEEE 37-node distributed power grid is simulated in MATLAB. The grid topology is shown in Fig. 8 (a). Node 799 is modeled as a power grid with a rated voltage of 4.8 kV. The rest of the nodes are modeled as linear loads. Besides, several DERs are added. For every DER, a three-phase inverter is used to convert DC power to an AC power grid, and a 20 kVA power transformer is used to connect the inverter to the distributed power grid. Two types of DERs are modeled. One type is modeled as the current source inverter (CSI), which represents the PV farm. The other type is the voltage source inverter (VSI), which simulates battery, gas turbine, etc. Compared with the CSI, the VSI provides constant voltage and sustains the frequency for the whole grid. In Fig. 8 (a), DER A is modeled as VSI., and DERs B, C, D, E are CSIs. In Fig.8 (d), DC supply could represent a solar panel, a wind turbine, or other DERs. The $L_f, C_f$, and $L_g$ work as a filter,eliminating the harmonics in voltages and currents.

We test and evaluate our approach in the aforementioned 37-node power grid under various settings which have different cyber attacks, DERs generation and load capacity. The DIAs are designed in the converter in the power grid as presented in Section III-A. In the attacks scenarios, the compromised data of DIA is expressed as $Y_0 = [i_f, u_{cf}, i_g]^T$. The attacks may occur on the trunk or branches of the grid due to the different locations of DERs. Seven electrical waveform sensors are placed in this power grid (green bars in Fig. 8 (a)). The goal is to detect and localize anomalies on power electronics in power grids by monitoring electrical waveform in sensors.

TABLE I
LOAD CAPACITY AND GENERATION SETTING

| Power level | DERs generation (kW) | Total load (kW) | Power level | DERs generation (kW) | Total load (kW) |
|---|---|---|---|---|---|
| Setting 1 | 33.5 | 78 | Setting 2 | 33.5 | 58.5 |
| Setting 3 | 33.5 | 39 | Setting 4 | 16.75 | 78 |
| Setting 5 | 16.75 | 58.5 | Setting 6 | 16.75 | 39 |

By the simulation model Eqn. (8), we simulated typical cyber attacks on inverters and a three-phase short circuit fault in a transmission line, considering a variety of PV farm power generations and loads. Table I shows the total generation of DERs and total loads in different settings. Here, the impact from two types of power generation are considered; three kinds

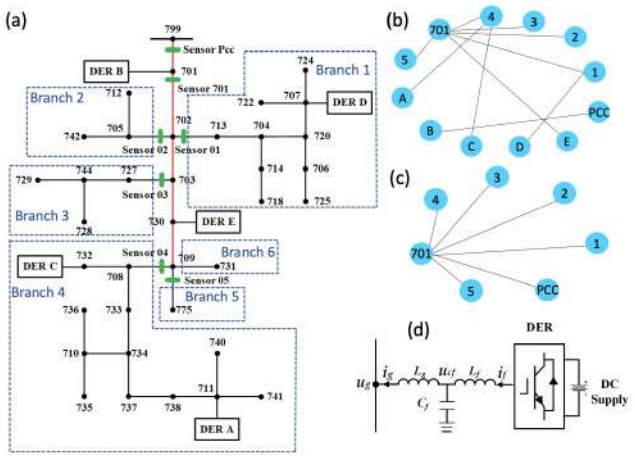


Fig. 8. (a) The 37-node power network. Green bars are sensors. The red line is the trunk, and other black lines are branches. The blue dash lines highlight the six branches, i.e., branches 1–6. DERs are inverters. Node 799 represents the power grid. (b) Construction of $G^2(t)$ for sensor 01–05, 701, DER A–E, and PCC. (c) Construction of $G^2(t)$ for sensor 01–05, 701, and PCC. (d) DERs model.

TABLE II
SIMULATION SETTING AND ATTACK TARGET/FAULT

| Case | Targeted DER/fault | Power level setting | Anomaly Location |
|---|---|---|---|
| Case 1 | A | Setting 1 | Branch 4 |
| Case 2 | B | Setting 1 | Trunk |
| Case 3 | C | Setting 4 | Branch 4 |
| Case 4 | D | Setting 4 | Branch 1 |
| Case 5 | A | Setting 3 | Branch 4 |
| Case 6 | B | Setting 3 | Trunk |
| Case 7 | C | Setting 6 | Branch 4 |
| Case 8 | D | Setting 6 | Branch 1 |
| Case 9 | A | Setting 2 | Branch 4 |
| Case 10 | B | Setting 2 | Trunk |
| Case 11 | C | Setting 5 | Branch 4 |
| Case 12 | D | Setting 5 | Branch 1 |
| Case 13 | 730-DER E (fault) | Setting 6 | Trunk |

of loads capacity are modeled in the grid. The sampling rate is 20 kHz. Waveform data during one second is simulated for each setting. Under aforementioned different settings, we designed 13 cases, as shown in Table II. For case 1 to case 12, the anomaly is designed to falsify different controller sensors in DER. Case 13 simulates the three-phase short circuit fault at the transmission line (node 730 - DER E). In all cases, the cyber attacks begin at 0.2 s and end at 0.4 s. We collected the waveform data of seven sensors (i.e., sensor 01-05, 701, and PCC) and five DERs (i.e., A, B, C, D, and E). For each sensor or DER, we have 20,000 observations recording three-phase voltages and three-phase currents.

### C. Results of CONGO$^2$

We consider the following two scenarios. In scenario (A), we use all waveform data collected by the seven sensors (i.e., sensor 01–05, 701, and PCC) and the five DERs. In real-world applications, we sometimes do not have the waveform data of the device where the anomaly occurs. To evaluate the performance of our method in such a scenario, we consider the following scenario. In scenario (B), we only have the waveform data collected by the seven sensors, i.e., sensor 01–05, 701, and PCC, marked in green in Fig. 8 (a). Note that it is more difficult to detect and localize the anomalies in scenario (B). In this paper, we set $\tau = 1.2$ and $\rho = 1.2$. It is worth mentioning that our method is not sensitive to the specification of these two thresholds. More details of the sensitivity analysis refer to Section III-G.

TABLE III
DETECTION RESULTS VIA CONGO$^2$

| Case | Detection time | Localization in scenario (A) | Localization in scenario (B) |
|---|---|---|---|
| Case 1 | 0.2025 s | Branch 4 | Branch 4 |
| Case 2 | 0.2025 s | Trunk | Trunk |
| Case 3 | 0.2025 s | Branch 4 | Branch 4 |
| Case 4 | 0.2025 s | Branch 1 | Branch 1 |
| Case 5 | 0.2025 s | Branch 4 | Trunk |
| Case 6 | 0.2025 s | Trunk | Trunk |
| Case 7 | 0.2025 s | Branch 4 | Branch 4 |
| Case 8 | 0.2025 s | Branch 1 | Branch 1 |
| Case 9 | 0.2025 s | Branch 4 | Branch 4 |
| Case 10 | 0.2025 s | Trunk | Trunk |
| Case 11 | 0.2025 s | Branch 4 | Branch 4 |
| Case 12 | 0.2025 s | Branch 1 | Branch 1 |
| Case 13 | 0.2025 s | Trunk | Trunk |

Table III reports the results of CONGO$^2$ in the aforementioned 13 cases. From Table III, we have the following observations. First, our method detects anomaly at 0.2025 s for all cases. Note that the true anomaly occurs at 0.2 s. The delay of our method is only 0.0025 s. Second, in scenario (A), for all cases, our method correctly localizes the branch where the anomaly happens. Thus, in scenario (A), the accuracy is 100%. Third, in scenario (B), our method correctly localizes the branch where the anomaly happens for all cases except for case 5. In case 5, we inaccurately localize the anomaly in the trunk. This is because DER A meets the load consumption in the branch, which leads to a small power exchange between the branch (sensor 4) and the trunk. Thus, the detection result of case 5 in scenario (B) is not that strong compared with other power level setting and false localize the anomaly at the trunk. The accuracy in scenario (B) is 92%. More details of the typical examples can be found in Section III-G.

### D. Comparison results with other existing methods

First, we compare our method with the Hotelling $T^2$ chart, which is commonly used in multivariate statistical quality control [36]. Then, we use the Hotelling $T^2$ chart on instantaneous amplitudes of each sensor through the Hilbert transform on the data. We localize the anomaly branch to which the earliest anomaly sensors belong. Second, we compare the CUSUM method on each data dimension with our method, since we have six-dimensional data in each sensor. To make CUSUM more robust, the anomaly time detected by CUSUM is when at least three dimensions are detected as abnormal. When three or more sensors report the anomaly happens, we record the time as the detection time. Third, we make further comparison with a recently proposed method [37]

TABLE IV
RESULTS OF FOUR METHODS FOR ANOMALY DETECTION.

| Scenario | Method | ACC | Delay (s) | FP | TP |
|---|---|---|---|---|---|
| (A) | Hotelling $T^2$ | 10/19 | 0.0310 | **0/6** | 4/13 |
| | CUMSUM | 7/19 | 0.1503 | 2/6 | 3/13 |
| | Leverage | 15/19 | 0.0445 | **0/6** | 9/13 |
| | **CONGO**$^2$ | **19/19** | **0.0025** | **0/6** | **13/13** |
| (B) | Hotelling $T^2$ | 10/19 | 0.0310 | **0/6** | 4/13 |
| | CUMSUM | 11/19 | 0.1226 | **0/6** | 5/13 |
| | Leverage | 16/19 | 0.0569 | **0/6** | 10/13 |
| | **CONGO**$^2$ | **19/19** | **0.0025** | **0/6** | **13/13** |

TABLE V
RESULTS OF THREE METHODS FOR ANOMALY LOCALIZATION.

| Scenario | Method | ACC | FP | TP |
|---|---|---|---|---|
| (A) | Hotelling $T^2$ | 6/19 | **0/6** | 0/13 |
| | CUMSUM | 4/19 | 2/6 | 0/13 |
| | **CONGO**$^2$ | **19/19** | **0/6** | **13/13** |
| (B) | Hotelling $T^2$ | 6/19 | **0/6** | 0/13 |
| | CUMSUM | 6/19 | **0/6** | 0/13 |
| | **CONGO**$^2$ | **18/19** | **0/6** | **12/13** |

which proposes an unsupervised online anomaly detection method based on the leverage score of the feature matrix of all sensors. For convenience, we name this method in [37] as "Leverage". Note that the "Leverage" method can not be directly applied to anomaly localization, thus we only report the detection results of "Leverage" in Table IV.

In order to evaluate the false positive rate in the anomaly detection step, we further simulate six normal cases under six different power level settings. The anomaly detection and localization results are summarized in Tables IV and V, respectively. In Tables IV and V, TP records the true positive rate, Delay (s) records the time difference between the true anomaly time, i.e., 0.2 s, and the detected anomaly time; FP record the false positive rate; ACC records the accuracy. Note that the "Leverage" method focuses only on anomaly detection and is inapplicable to anomaly localization. Therefore, we only compare our method with CUSUM and Hotelling $T^2$ for localization problems.

From Table IV, we have the following observations. First, under both scenarios (A) and (B), the anomaly detection accuracy of our method is 100%, the TP of our method is 100%, and the FP of our method is zero. Second, our method quickly detects the anomaly once it happens. The average delay time of our method is only 0.0025 s, which is much smaller than the average delay time of other methods under both scenarios (A) and (B). In sum, our method achieves the highest TP and accuracy, lowest FP and delay time. From Table V, we observe that in scenario (A), the anomaly localization accuracy and TP of our method are both 100%, while those of Hotelling $T^2$ and CUSUM are less than 50%. In scenario (B), the anomaly localization accuracy and TP of our method decrease to nearly 95%, which is still much higher than the accuracy of other methods.

The Hotelling $T^2$ and CUSUM fail to detect and localize the anomaly for the following reasons. (i) CUSUM fails to capture the correlation between sensors. (ii) Hotelling $T^2$ fails to consider different types of interaction within sensors. (iii) All the other three methods focus on the amplitude information and ignore the phase angle information, which is a more powerful indicator for anomaly detection. Due to the characteristic of branch impedance in the distribution grid, the phase angles of voltage in different nodes are different. Thus, the phase angle can be extracted to monitor the system status. These three reasons make a relatively high false discovery rate. Since the alarming time for each sensor is very close, the CUSUM and Hotelling $T^2$ result in false locations for all 13 anomaly cases.

We then compare the computational cost and communication cost of different methods. It has been shown that the computational cost of CUSUM and Hotelling $T^2$ are both $O(|V|)$ [36]. The computational cost of "Leverage" method has been shown to be $O(|V|^2)$. In comparison, the computational cost of our method is $O(|V|) + O(|E|)$, which is no more than that of "Leverage". In particular, when $G^2(0)$ is a sparse network, i.e., $|E|$ is of the order $O(|V|)$, our method, CUSUM and Hotelling $T^2$ have the same computational cost, i.e., $O(|V|)$. Furthermore, the communication cost (in bits) of our method is $O(|V|) + O(|E|)$, If we apply CUSUM and Hotelling $T^2$ in a distributed way, the communication costs of CUSUM and Hotelling $T^2$ are both $O(|V|)$. When $G^2(0)$ is sparse network, i.e., $|E|$ is of the order $O(|V|)$, our method, CUSUM and Hotelling $T^2$ have the same communication cost, i.e., $O(|V|)$. Since [37] did not design the distributed algorithm of the "Leverage" method, we do not make a comparison with the "Leverage" method regarding the communication cost. Aforementioned observations suggest that our method uniformly outperforms other methods.

### *E. Sensitivity analysis of CONGO*$^2$

In our proposed method CONGO$^2$, we have a tuning parameter $\tau$ for anomaly detection and a tuning parameter $\rho$ for anomaly localization. In this subsection, we evaluate the performance of CONGO$^2$ under varying $\tau$ and $\rho$. First, to assess the robustness of CONGO$^2$ to different $\tau$, we investigate the anomaly detection accuracy while $\tau$ varies from 1.0 to 1.5. As shown in Fig. 9 (a) and (b), the anomaly detection accuracy keeps one when $\tau$ varies from 1.2 to 1.3, under both scenarios (A) and (B). Second, we investigate the anomaly localization accuracy while $\rho$ varies from 1.05 to 1.5. As shown in Fig. 9 (c), in scenario (A), the anomaly localization accuracy keeps one under various $\rho$. As shown in Fig. 9 (d), in scenario (B), the anomaly localization accuracy stays constant, which is very close to one under various $\rho$. This indicates that CONGO$^2$ is robust to $\rho$. In summary, our method CONGO$^2$ is robust to $\rho$, and we suggest setting $\tau$ between 1.2 and 1.3 in the application.

### *F. Distributed CONGO*$^2$

As discussed in Section II-E, our method is naturally suitable for decentralized distributed implementation with less communication cost than a centralized setting. For the centralized implementation, at time $t$, all information will be sent to a chosen central node from other nodes. Then the

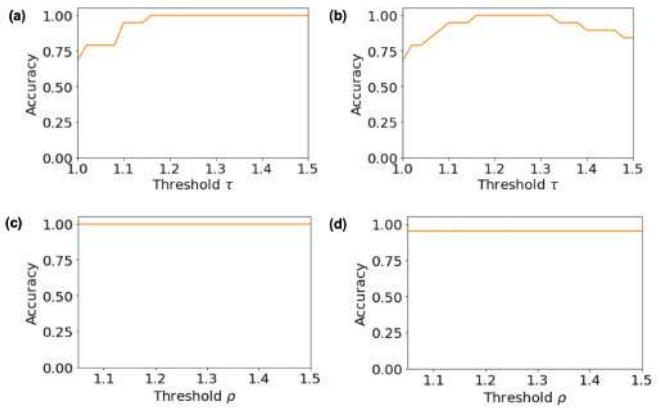


Fig. 9. (a) and (b) show the detection accuracy with $\tau$ ranging from 1.0 and 1.5, under scenario (A) and scenario (B), respectively. (c) and (d) show the localization accuracy with $\rho$ ranging from 1.05 to 1.50, under scenario (A) and scenario (B), respectively.

$PC(t)$ and corresponding $cPC_i(t)$ will be calculated. For the distributed implementation, at time $t$, neighbors first exchange information through broadcast based on the topology of $G^2(t)$. There is no information exchange between them if there is no edge between two nodes in $G^2(t)$. After every node gets its neighbors' information, a tree-based network will be constructed as described in Fig. 6, and the node will send its local result from leaf to root, back and forth for two rounds. In the first round, $dis_i(t)$ and $deg_i(t)$ are sent to the root node to calculate the $PC(t)$ and back along to the leaf nodes to calculate respective $cPC_i(t)$. In the second round, the $cPC_i(t)$ from different nodes are sent back to the root node for further event localization. Since most computation is in the local node, only results would be transferred, so communication overhead is low.

To evaluate the communication cost, we consider the $G^2(t)$ with 13 nodes. Unlike Fig 8(b), the $G^2(t)$ is designed according to the topology of the 37-node power network, and we randomly draw $k$ edges in the corresponding graph. The communication cost ratio between centralized settings and distributed settings with different $k$ is reported in Fig. 10. One can see that the decentralized framework significantly accelerates the algorithm, especially when $G^2(t)$ is sparse.

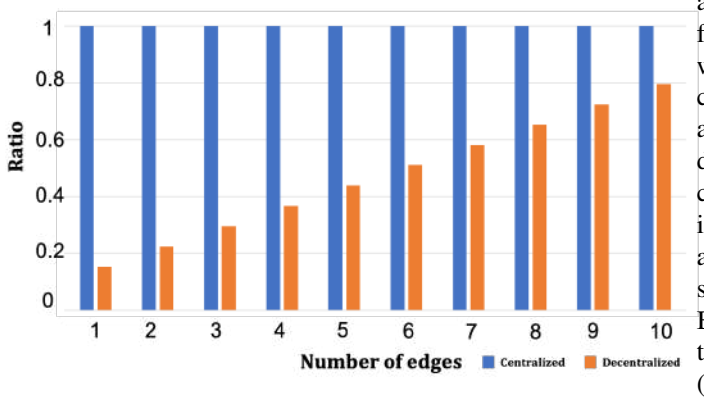


Fig. 10. Communication cost in each time steps $t$ when the number of the edges of $G^2(t)$ are different from the centralized setting.

### G. More details of simulation results

We provide more details and discussions of the simulation results by taking the normal case under setting 1, cases 1, 8 and 13 for representative examples. We opt to present cases 1, 8 and 13 because they have different targeted faults, i.e., DER A, D and three-phase short circuit.

We first present one normal case (in which no anomaly occurs) under setting 1 in Fig. 11. Fig. 11 (a) and (c) show the phase change score of different time for scenarios (A) and (B), respectively. It is observed that there is no abrupt change in phase change score, suggesting no anomaly happens. Fig. 11 (b) and (d) present the $G^2(t)$ at 0.2025 s for scenario (A) and (B), respectively. Since all the $G^2(t)$ plot for all time $t$ are the same, we only demonstrate the plot at time 0.2025 s when the anomaly was detected in Table III. For visualization purposes, at time $t$, we only show an edge between nodes $i$ and $j$ of $G^2(t)$ if $d_{ij}(t)$ is greater than 0.85 throughout this section.

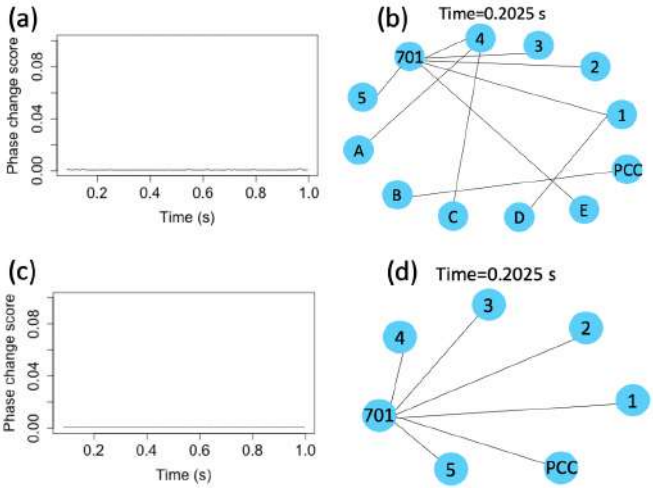


Fig. 11. (a) Phase change score for scenario (A); (b) Visualization of the $G^2(t)$ at 0.2025 s for scenario (A); (c) Phase change score for scenario (B); (d) Visualization of the $G^2(t)$ at 0.2025 s for scenario (B).

We then present the results of case 1 in Fig. 12. In case 1, a cyber attack occurs from 0.2 s to 0.4 s. The attacker falsifies DER A's controller sensor and the measurement $i_{lf}, u_{cf}, i_{lg}$ are changed into fake ones. The related parameters can be found in Tables I and II. Fig. 12 (a) visualizes the location where the true anomaly occurs for case 1. The red and pink circle highlights the anomaly localization for scenarios (A) and (B). Fig. 12 (b) and (d) show the phase change score of different times for scenarios (A) and (B), respectively. One can observe that the sudden jump first occurs around 0.2 s, implying that the anomaly happens. The two sudden jumps around 0.4 s and 0.7s stand for the attack ended, and the system is back to normal after the system self-adjustment. Because it takes some time for the PI controller to track the reference and adjust the system status, Fig. 12 (c) and (e) present the $G^2(t)$ at 0.2025 s for scenarios (A) and (B), respectively. It is observed that in scenario (A), the candidate anomaly sensors detected by our method are sensor A and sensor 4; in scenario (B), the candidate anomaly sensors

detected by our method are sensor 4. The $cPC$s for the sensors 01–05, 701, PCC, and DERs A–E are around 0.01, 0.03, 0.03, 0.25, 0.03, 0.07, 0.07, 0.4, 0.03, 0.05, 0.00, and 0.03, respectively at time 0.2025 s. This implies the anomaly happens in branch 4, detected by our algorithm.

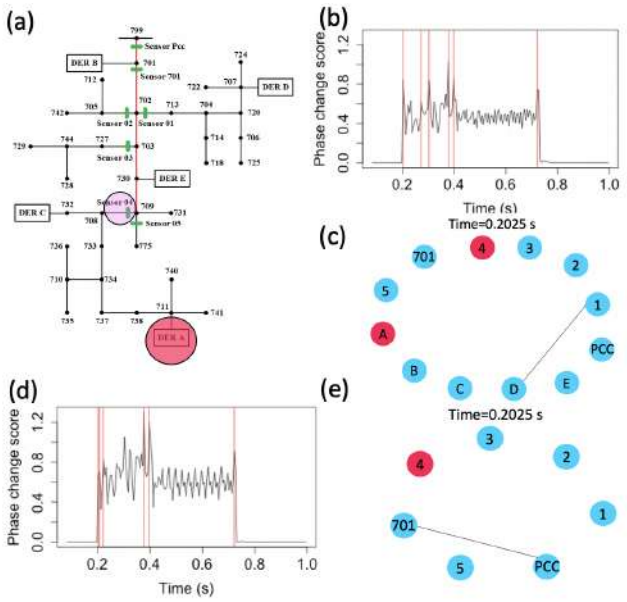


Fig. 12. (a) The 37-node distributed grid, with the circles in red and pink representing the true anomaly targeted location for scenarios (A) and (B), respectively; (b) Phase change score on the time interval for scenario (A); (c) Visualization of the $G^2(t)$ on time 0.2025 s under case 1 and scenario (A). (d) Phase change score on the time interval for scenario (B); (e) Visualization of the $G^2(t)$ on time 0.2025 s under case 1 and scenario (B). In (b) and (d), the red lines highlight when phase changes are detected. In (c) and (e), the nodes colored in red are the candidate anomaly sensors by the CONGO$^2$ Algorithm.

In case 8, a cyber attack is designed for compromising the controller sensor of DER D. Fig. 13 (a) visualizes the location where the true anomaly occurs for case 8. Fig. 13 (b) shows the phase change scores for scenario (A). One can observe a sudden jump at time 0.2025 s in phase change score, which implies the anomaly happens, and another jump around 0.4s, corresponding to the case that the attack ended. This is because DER D is modeled as CSI. The sensor attack changes the measurement feedback of PI control, which only impacts the performance of the current control loop. Compared to case 1, an attack on DER A influences the voltage of the whole system, which means it takes a longer time for DER A to restore the voltage status of the whole system. The $cPC$s for the sensors 01–05, 701, PCC, and DERs A–E are around 0.57, 0.02, 0.02, 0.01, 0.02, 0.06, 0.04, 0.01, 0.02, 0.01, 0.20, and 0.02, respectively at time 0.2025 s. This implies the anomaly occurs near or at sensor 01, the accurate attacked branch in the grid. The results for scenario (B) that only the data in sensors 01–05, 701, and PCC are used are shown in Fig. 13 (d) and 13 (e) we can also conclude the same result.

In case 13, a three-phase short circuit is simulated. This fault location is shown in Fig. 14 (a): between current inverter E and load 730. The detailed settings can be obtained from TABLE II. Fig. 14 (b) displays the phase change scores for

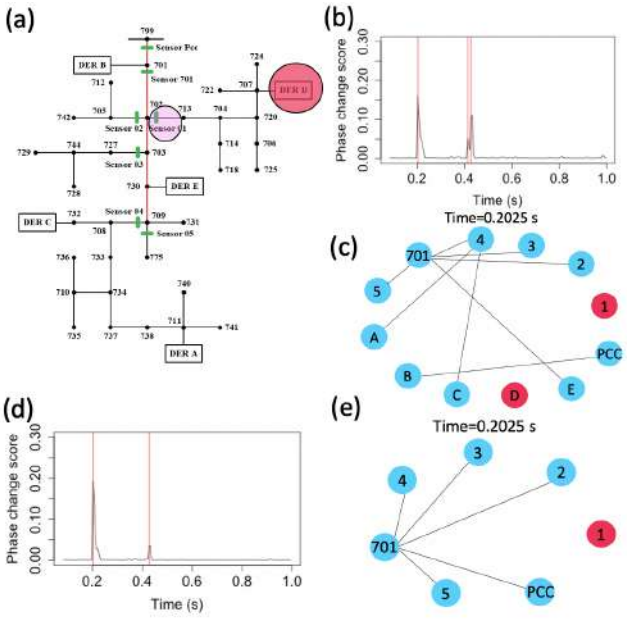


Fig. 13. (a) The 37-node distributed grid, with the circles in red and pink representing the true anomaly targeted location for scenarios (A) and (B), respectively; (b) Phase change score on the time interval for scenario (A); (c) Visualization of the $G^2(t)$ on time 0.2025 s under case 8 and scenario (A); (d) Phase change score on the time interval for scenario (B); (e) Visualization of the $G^2(t)$ on time 0.2025 s under case 8 and scenario (B). In (b) and (d), the red lines highlight when phase changes are detected. In (c) and (e), the nodes colored in red are the candidate anomaly sensors by the CONGO$^2$ Algorithm.

scenario (A). One can observe a sudden jump at time 0.2025 s in phase change score, which implies the anomaly happens, and another jump around 0.4s corresponds to the case that the attack ended. The $cPC$s for the sensors 01–05, 701, PCC, and DERs A–E are around 0.15, 0.03, 0.15, 0.15, 0.03, 0.17, 0.17, 0.03, 0.03, 0.03, 0.03, and 0.03, respectively at time 0.2025 s. In this case, sensors 01, 03, 04, 701, and PCC are anomaly sensors according to the CONGO$^2$ Algorithm. More than half branches are detected as abnormal, which implies the anomaly may happen at the trunk. The results for scenario (B) that only the data in sensors 01–709 and PCC are collected, are shown in Fig. 14 (d) and 14 (e) we can also locate the anomaly at the trunk.

## IV. Conclusion

In this paper, we developed a graph-based methodology, i.e., CONGO$^2$, to detect and localize anomalies on power grids. Some typical DIAs and faults, such as the three-phase short circuits, were simulated in the IEEE 37-node distributed power grid. In particular, we combined the physical knowledge and observed waveform data to construct a two-layer graph to describe the power network condition at each time point. The CONGO$^2$ enjoys the following advantages. First, our method can both detect and localize anomalies with high accuracy. Second, our method is based on unsupervised online learning, which does not require a training stage, making it efficient and implementable in a real-time problem. Third, our method only requires the local topological structure of the smart grid. Thus, our method can be easily extended to a large-scale

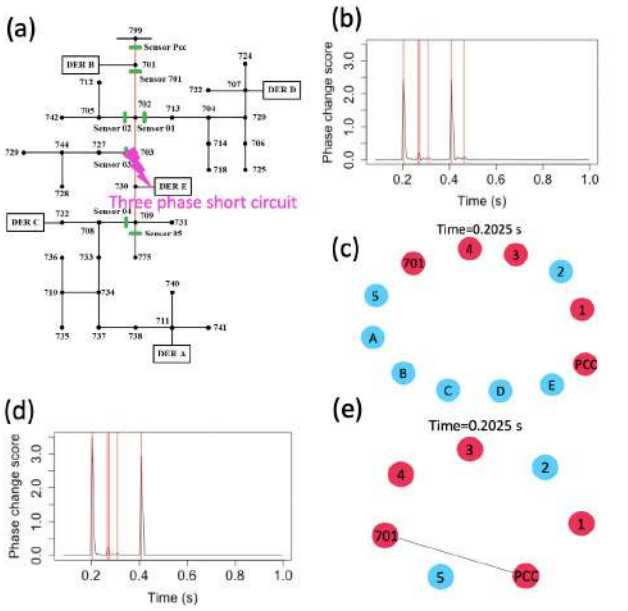


Fig. 14. (a) The 37-node distributed grid, with the circles in red and pink representing the true anomaly targeted location for scenarios (A) and (B), respectively; (b) Phase change score for scenario (A); (c) Visualization of the $G^2(t)$ at time 0.2025 s under case 13 and scenario (A); (d) Phase change score for scenario (B); (e) Visualization of the $G^2(t)$ at 0.2025 s under case 13 and scenario (B). In (b) and (d), the red lines highlight when phase changes are detected. In (c) and (e), the nodes colored in red are the candidate anomaly sensors by the CONGO$^2$ Algorithm.

grid system. Fourth, the graph representation of smart grids allows us to leverage many cutting-edge graph algorithms for downstream analysis. As a sequel to this work, a graph neural network based method for identifying the root cause of anomalies, e.g., three-phase short circuit (case 13 in this paper) and cyber attack (cases 1-12 in this paper), is under active development. Despite the success of our method in detecting the cyber threats in the device, as demonstrated in Section III-B, some interesting situation, such as detection of DIAs in the system estimation and control, still needs to be studied in future work.

## APPENDIX A
## MATHEMATICAL ANALYSIS OF CONGO$^2$

In this section, we provide a detailed mathematical analysis of CONGO$^2$, showing that CONGO$^2$ can detect anomalies with a theoretical guarantee. We assume that the observed waveform $x_{ij}(t), j = 1, \ldots, 6$, is a linear combination of the true signal and the random noise, i.e., we have

$$x_{ij}(t) = z_{ij}(t) + \varepsilon_{ij}(t), \tag{9}$$

where $z_{ij}(t)$ represents the true signal, and $\varepsilon_{ij}(t)$ represents the random noise. In the following analysis, we consider two scenarios.

- **Scenario 1**: Noise-free scenario, i.e., $\varepsilon_{ij}(t) = 0$.
- **Scenario 2**: Random noise term exists, $\varepsilon_{ij}(t) \neq 0$.

To establish the connection between anomaly and our proposed $PC$ score we impose the following Model Assumption 1 and Condition 1.

**Model Assumption 1.** *If no anomaly occurs, the six waveform signals of all sensors show the stable pattern: (1) For each signal $z_{ij}(t)$, $i = 1, \ldots, m; j = 1, \ldots, 6$, it has a stable linear relationship with its previous values, i.e., $z_{ij}(t) = a_{ij}^{\langle 1 \rangle} z_{ij}(t-1) + \ldots + a_{ij}^{\langle q_j \rangle} z_{ij}(t-q_j)$, where $a_{ij}^{\langle 1 \rangle}, \ldots, a_{ij}^{\langle q_j \rangle}$ are constant coefficients, and $q_j$ varies for different $j$. (2) For any two signals $z_{ij}(t)$ and $z_{il}(t)$, $j = 1, \ldots, 6, l = 1, \ldots, 6$, the interaction between them is stable, i.e., $z_{ij}(t) = \beta_{ij}^{\langle 1 \rangle} z_{il}(t-1) + \ldots + \beta_{ij}^{\langle q_{jl} \rangle} z_{il}(t-q_{jl})$, where $\beta_{ij}^{\langle 1 \rangle}, \ldots, \beta_{ij}^{\langle q_{jl} \rangle}$ are constant coefficients, and $q_{jl}$ may vary for different $j$ and $l$. Note that in this paper, we only consider the interaction between voltage signals and the current signals, indicating that we only consider $\{q_{14}, q_{15}, q_{16} \ldots, q_{36}\}$.*

**Condition 1.** *Let $k, k'$ denote the number of rows and columns of the trajectory matrix in Eq. (1) in the main manuscript, $k, k'$ and $q_1, \ldots, q_6, q_{14}, \ldots, q_{36}$ satisfy that $\min(k, k') > \max(q_1, \ldots, q_6, q_{14}, \ldots, q_{36})$.*

**Analysis under the scenario 1.** Under the noise-free scenario, the following Lemma 1 indicates that the proposed $PC$ score is a good indicator for anomaly detection.

**Lemma 1.** *Under the noise-free scenario, assume Model Assumption 1 and Condition 1 hold, if no anomaly occurs at time $t$, we have $PC(t) = 0$. Thus, $PC(t) > 0$ indicates that an anomaly occurs at time $t$.*

*Proof.* To prove $PC(t) = 0$, it is sufficient to prove $dis_i(t) = 0$ and $\Delta deg_i(t) = 0$, when there is no anomaly happens, for each sensor $i$.

First, we prove that if no anomaly occurs at time $t$, we have $dis_i(t) = 0$, $i = 1, \ldots, m$. Recall that $dis_i(t) = 0.5||B_i(t) - B_i(t-1)||_2^2 + 0.5||A_i(t) - A_i(t-1)||_F^2$. The $dis_i(t) = 0$ implies $||B_i(t) - B_i(t-1)||_2^2 = 0$ and $||A_i(t) - A_i(t-1)||_F^2 = 0$.

We now present the calculation of $b_{ij}(t)$ which is the $j$th elements in $B_i(t)$. Let $\boldsymbol{Z}_{ij}(t)$ denote the trajectory matrix calculated using the data $z_{ij}(t)$, where the trajectory matrix is defined in Eq. (1) in the main manuscript. Let $\mathcal{Z}_{ij}(t-1)$ and $\mathcal{Z}_{ij}(t)$ be the column spaces spanned by $\boldsymbol{Z}_{ij}(t-1)$ and $\boldsymbol{Z}_{ij}(t)$. Under the Model Assumption 1, if no anomaly occurs at time $t$, we have $z_{ij}(t) = a_{ij}^{\langle 1 \rangle} z_{ij}(t-1) + \ldots + a_{ij}^{\langle q_j \rangle} z_{ij}(t-q_j)$. Therefore, the last column in $\boldsymbol{Z}_{ij}(t)$ can be represented by a linear combination of the last $q_j$ columns in $\boldsymbol{Z}_{ij}(t-1)$. Note that when the Condition 1 hold, all the columns in $\boldsymbol{Z}_{ij}(t)$ can be represented by a linear combination of the columns in $\boldsymbol{Z}_{il}(t)$. Thus, we have $\mathcal{Z}_{ij}(t) \subseteq \mathcal{Z}_{ij}(t-1)$. By Theorem 2 in [26], when $\mathcal{Z}_{ij}(t) \subseteq \mathcal{Z}_{ij}(t-1)$, the Krylov subspace distance between $\mathcal{Z}_{ij}(t-1)$ and $\mathcal{Z}_{ij}(t)$ is zero, i.e., $b_{ij}(t) = 0$ for all $i = 1, \ldots, m$ and $j = 1, \ldots, 6$. Thus, if no anomaly occurs, we have $||B_i(t) - B_i(t-1)||_2^2 = 0$. for all $i = 1, \ldots, m$.

Analogously, we can prove that if no anomaly occurs at time $t$, $\mathcal{Z}_{il}(t) \subseteq \mathcal{Z}_{il}(t)$. Again we apply the Theorem 2 in [26], we prove that $Kry(z_{ij}(t), z_{il}(t)) = 0$, implying $exp\{-Kry(z_{ij}(t), z_{il}(t))\} = 1$. Thus, when anomaly occurs, all edges have a constant weight, i.e., one. Thus, if no anomaly occurs, we have $||A_i(t) - A_i(t-1)||_F^2 = 0$, for all $i = 1, \ldots, m$ and $j = 1, \ldots, 6$. Therefore, combining the results of $||B_i(t) - B_i(t-1)||_2^2$ and $||A_i(t) - A_i(t-1)||_F^2$,

we come to the conclusion that if no anomaly occurs at time $t$, we have $dis_i(t) = 0$.

Second, we show that if no anomaly occurs at time $t$, we have $\Delta deg_i(t) = 0, i = 1, \ldots, m$. From previous analysis, we know that if no anomaly occurs, $(b_{i1}, \ldots, b_{i6}) = (b_{j1}, \ldots, b_{j6}) = (0, \ldots, 0)$, thus we have $||B_i(t) - B_j(t)||_2^2 = 0$. Furthermore, all edges in $A_i(t)$ and $A_j(t)$ have the same weight, i.e., one, when no anomaly occurs. Thus, we have $||A_i(t) - A_j(t)||_F^2 = 0$. Therefore, combining the results of $||B_i(t) - B_j(t)||_2^2$ and $||A_i(t) - A_j(t)||_F^2$, we come to the conclusion that if no anomaly occurs at time $t$, we have $\Delta deg_i(t) = 0$.

Combining the results of $dis_i(t)$ and $\Delta deg_i(t)$, we conclude that under the noise-free scenario, assume Model Assumption 1 and Condition 1 hold, $PC(t) > 0$ indicates that an anomaly occurs at time $t$. □

**Analysis under the scenario 2.** Under the scenario where the random noise term exists, the observed signal $x_{ij}(t)$ is not necessarily equal to the true signal $z_{ij}(t)$. From Corollary 6.1 in [27], we know that assuming $\varepsilon_{ij}(t)$ is a Gaussian white noise, as $k \to \infty$ and $k' \to \infty$ with $k/k' \to c_0 > 0$, where $c_0$ is a constant, the observed signal $x_{1j}(t)$ is stochastically separable from the random noise $\varepsilon_{ij}(t)$. This indicates that the space spanned by the trajectory matrix $\boldsymbol{X}_{ij}(t)$, which is calculated using $x_{ij}(t)$, can be decomposed into two orthogonal subspaces, thus we have $\boldsymbol{X}_{ij}(t) = \boldsymbol{Z}_{ij}(t) + \boldsymbol{E}_{ij}(t)$, where $\boldsymbol{Z}_{ij}(t)$ denotes the trajectory matrix calculated using $z_{ij}(t)$, and $\boldsymbol{E}_{ij}(t)$ denotes the trajectory matrix calculated using $\varepsilon_{ij}(t)$.

We can write the singular value decomposition (SVD) of trajectory matrices $\boldsymbol{Z}_{ij}(t)$, $\boldsymbol{E}_{ij}(t)$ and $\boldsymbol{X}_{ij}(t)$ as $\boldsymbol{Z}_{ij}(t) = \sum_l \lambda_{ijl} U_{ijl}(t) V_{ijl}^T(t)$, $\boldsymbol{X}_{ij}(t) = \sum_l \tilde{\lambda}_{ijl} \tilde{U}_{ijl}(t) \tilde{V}_{ijl}^T(t)$, $\boldsymbol{E}_{ij}(t) = \sum_l \dot{\lambda}_{ijl} \dot{U}_{ijl}(t) \dot{V}_{ijl}^T(t)$, where $\lambda_{ijl}$ is the $l$th eigenvalue of $\boldsymbol{Z}_{ij}(t)$, $U_{ijl}(t)$ is the left singular vector that spans the column space of $\boldsymbol{Z}_{ij}(t)$; $V_{ijl}(t)$ is the right singular vector; $\tilde{\lambda}_{ijl}$, $\tilde{U}_{ijl}(t)$, and $\tilde{V}_{ijl}^T(t)$ denote the analogous representation for $\boldsymbol{X}_{ij}(t)$; $\dot{\lambda}_{ijl}$, $\dot{U}_{ijl}(t)$, and $\dot{V}_{ijl}^T(t)$ denote the analogous representation for $\boldsymbol{E}_{ij}(t)$.

Under the Model Assumption 1, the dimension of column space of $\boldsymbol{Z}_{ij}(t)$ is $q_j$. Recall that $\mathcal{Z}_{ij}(t)$ denote the column space of $\boldsymbol{Z}_{ij}(t)$. Let $\mathcal{U}_{ij}(t)$ denote the space spanned by $\tilde{U}_{ij1}(t), \ldots, \tilde{U}_{ijq_j}(t)$. If $\min_l \{\lambda_{ijl}\} > \max_l \{\dot{\lambda}_{ijl}\}$, for all $i = 1, \ldots, m; j = 1, \ldots, 6$, $\mathcal{U}_{ij}(t)$ is a consistent estimate of $\mathcal{Z}_{ij}(t)$. Thus, $Kry(\mathcal{U}_{ij}(t-1), \mathcal{U}_{ij}(t))$ is a consistent estimate of $Kry(\mathcal{Z}_{ij}(t-1), \mathcal{Z}_{ij}(t))$. Thus $Kry(\mathcal{U}_{ij}(t-1), \mathcal{U}_{ij}(t)) > 0$ implies $Kry(\mathcal{Z}_{ij}(t-1), \mathcal{Z}_{ij}(t)) > 0$, which further indicates an anomaly occurs by Lemma 1. Recall that in this paper, $PC$ score depends on $Kry(\mathcal{U}_{ij}(t-1), \mathcal{U}_{ij}(t))$ to determine whether an anomaly occurs. Therefore, $PC$ is a good indicator for anomaly detection under the scenario 2.

## APPENDIX B
## PSEUDO CODE OF CONGO$^2$ ALGORITHM

**Algorithm 1:** CONGO$^2$ Algorithm

**Result:** The anomaly detection time $t^*$ and localized anomaly sensors at time $t^*$.
**Input**: Waveforms of $m$ sensors at each time point, and the length of the time interval $h$, predefined parameters $\tau$ and $\rho$ ($\rho > 1$) ;
**for** *time interval* $[t-h, t]$ **do**
- 1) Build an undirected colored-node graph $G_i(t)$ for sensor $i$, where $i = 1, \ldots, m$ (details of constructing $G_i(t)$ can be found in section II-B).;
- 2) Build an undirected colored-node graph $G^2(t)$ (details of constructing $G^2(t)$ can be found in section II-C).;
- 3) Calculate the phase change $PC(t)$ according to Eq. (5).;
- 4) **if** $PC(t)/\text{EMAPC}(t) > \tau$ **then**
  - 4.1 Anomaly detection. Output $t^* = t$ ;
  - 4.2 Calculate the contribution of the phase change score, i.e., $cPC_i(t)$, according to Eq.(6) for each sensor. Get candidate anomaly sensors by selecting those sensors satisfying $cPC_i(t) > \rho/m$.
  - 4.3 Anomaly localization. Localize the anomaly according to the four criteria in Section II-D.
- **end**

**end**

## REFERENCES


[1] J. C. Balda, A. Mantooth, R. Blum, and P. Tenti, "Cybersecurity and power electronics: Addressing the security vulnerabilities of the internet of things," *IEEE Power Electronics Magazine*, vol. 4, pp. 37–43, 2017.

[2] F. Li, A. Shinde, Y. Shi, J. Ye, X. Li, and W. Song, "System statistics learning-based iot security: Feasibility and suitability," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6396–6403, 2019.

[3] F. Li, R. Xie, Z. Wang, L. Guo, J. Ye, P. Ma, and W. Song, "Online distributed IoT security monitoring with multidimensional streaming big data," *IEEE Internet of Things Journal*, vol. 7, pp. 4387–4394, 2020.

[4] S. Sridhar and M. Govindarasu, "Model-based attack detection and mitigation for automatic generation control," *IEEE Transactions on Smart Grid*, vol. 5, pp. 580–591, 2014.

[5] Y. Isozaki, S. Yoshizawa, Y. Fujimoto, H. Ishii, I. Ono, T. Onoda, and Y. Hayashi, "Detection of cyber attacks against voltage control in distribution power grids with PVs," *IEEE Transactions on Smart Grid*, vol. 7, pp. 1824–1835, 2015.

[6] 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, pp. 2693–2703, 2017.

[7] K. Pan, P. Palensky, and P. M. Esfahani, "From static to dynamic anomaly detection with application to power system cyber security," *IEEE Transactions on Power Systems*, vol. 35, pp. 1584–1596, 2020.

[8] X. Shi, R. Qiu, Z. Ling, F. Yang, H. Yang, and X. He, "Spatio-temporal correlation analysis of online monitoring data for anomaly detection and location in distribution networks," *IEEE Transactions on Smart Grid*, vol. 11, pp. 995–1006, 2020.

[9] M. Cui, J. Wang, and M. Yue, "Machine learning-based anomaly detection for load forecasting under cyberattacks," *IEEE Transactions on Smart Grid*, vol. 10, pp. 5724–5734, 2019.

[10] M. Adiban, A. Safari, and G. Salvi, "Step-gan: A one-class anomaly detection model with applications to power system security," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 2605–2609.

[11] Z. Qu, H. Liu, Z. Wang, J. Xu, P. Zhang, and H. Zeng, "A combined genetic optimization with adaboost ensemble model for anomaly detection in buildings electricity consumption," *Energy and Buildings*, vol. 248, p. 111193, 2021.

[12] D. Saraswat, P. Bhattacharya, M. Zuhair, A. Verma, and A. Kumar, "Ansmart: A svm-based anomaly detection scheme via system profiling in smart grids," in *2021 2nd International Conference on Intelligent Engineering and Management*, 2021, pp. 417–422.

[13] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, 2009.

[14] R. Killick and I. A. Eckley, "changepoint: An R package for changepoint analysis," *Journal of Statistical Software, Articles*, vol. 58, pp. 1–19, 2014.

[15] D. C. Montgomery, *Introduction to Statistical Quality Control (8th Edition)*. Introduction to Statistical Quality Control, 2019.

[16] X. Xuan and K. Murphy, "Modeling changing dependency structure in multivariate time series," in *Proceedings of the 24th international conference on Machine learning*, 2007, pp. 1055–1062.

[17] G. D'Antona, C. Muscas, and S. Sulis, "Localization of nonlinear loads in electric systems through harmonic source estimation," *IEEE*

*Transactions on Instrumentation and Measurement*, vol. 60, pp. 3423–3430, 2011.

[18] D. Saxena, S. Bhaumik, and S. Singh, "Identification of multiple harmonic sources in power system using optimally placed voltage measurement devices," *IEEE Transactions on Industrial Electronics*, vol. 61, pp. 2483–2492, 2013.

[19] T. Idé, S. Papadimitriou, and M. Vlachos, "Computing correlation anomaly scores using stochastic nearest neighbors," in *Seventh IEEE international conference on data mining*. IEEE, 2007, pp. 523–528.

[20] T. Idé, A. C. Lozano, N. Abe, and Y. Liu, "Proximity-based anomaly detection using sparse structure learning," in *Proceedings of the 2009 SIAM international conference on data mining*. SIAM, 2009, pp. 97–108.

[21] S. Hirose, K. Yamanishi, T. Nakata, and R. Fujimaki, "Network anomaly detection based on eigen equation compression," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 1185–1194.

[22] R. Jiang, H. Fei, and J. Huan, "Anomaly localization for network data streams with graph joint sparse pca," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011, pp. 886–894.

[23] S. Hara, T. Morimura, T. Takahashi, H. Yanagisawa, and T. Suzuki, "A consistent method for graph based anomaly localization," in *Artificial intelligence and statistics*, 2015, pp. 333–341.

[24] H. Cheng, Y. Ning, Z. Yin, C. Yan, X. Liu, and Z. Zhang, "Community detection in complex networks using link prediction," *Modern Physics Letters B*, vol. 32, p. 1850004, 2018.

[25] X. Liu, H. Cheng, and Z. Zhang, "Evaluation of community detection methods," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1736–1746, 2020.

[26] T. Idé and K. Tsuda, "Change-point detection using krylov subspace learning," in *Proceedings of the 2007 SIAM International Conference on Data Mining*. SIAM, 2007, pp. 515–520.

[27] N. Golyandina, V. Nekrutkin, and A. A. Zhigljavsky, *Analysis of time series structure: SSA and related techniques*. CRC press, 2001.

[28] L. Scrucca, "qcc: an R package for quality control charting and statistical process control," *R News*, vol. 4/1, pp. 11–17, 2004.

[29] M. Valero, F. Li, S. Wang, F. Lin, and W. Song, "Real-time cooperative analytics for ambient noise tomography in sensor networks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 5, pp. 375–389, 2018.

[30] J. Ye, A. Giani, A. Elasser, S. K. Mazumder, C. Farnell, H. A. Mantooth, T. Kim, J. Liu, B. Chen, G.-S. Seo *et al.*, "A review of cyber-physical security for photovoltaic systems," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 2021.

[31] M. Chlela, G. Joos, M. Kassouf, and Y. Brissette, "Real-time testing platform for microgrid controllers against false data injection cybersecurity attacks," in *2016 IEEE Power and Energy Society General Meeting*, 2016, pp. 1–5.

[32] J. Choi, D. Narayanasamy, B. Ahn, S. Ahmad, J. Zeng, and T. Kim, "A real-time hardware-in-the-loop (hil) cybersecurity testbed for power electronics devices and systems in cyber-physical environments," in *2021 IEEE 12th International Symposium on Power Electronics for Distributed Generation Systems*, 2021, pp. 1–5.

[33] D. Jin, Z. Li, C. Hannon, C. Chen, J. Wang, M. Shahidehpour, and C. W. Lee, "Toward a cyber resilient and secure microgrid using software-defined networking," *IEEE Transactions on Smart Grid*, vol. 8, pp. 2494–2504, 2017.

[34] T. Kim, J. Ochoa, T. Faika, A. Mantooth, J. Di, Q. Li, and Y. Lee, "An overview of cyber-physical security of battery management systems and adoption of blockchain technology," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 2020.

[35] A. Barua and M. A. Al Faruque, "Special session: Noninvasive sensor-spoofing attacks on embedded and cyber-physical systems," in *2020 IEEE 38th International Conference on Computer Design (ICCD)*, 2020, pp. 45–48.

[36] R. L. Mason and J. C. Young, *Multivariate statistical process control with industrial applications*. SIAM, 2002, vol. 9.

[37] F. Li, R. Xie, B. Yang, L. Guo, P. Ma, J. Shi, J. Ye, and W. Song, "Detection and identification of cyber and physical attacks on distribution power grids with pvs: An online high-dimensional data-driven approach," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 2019.