Dynamic Graph-Based Anomaly Detection in the Electrical Grid

Shimiao Li, *Graduate Student Member, IEEE*, Amritanshu Pandey, *Member, IEEE*, Bryan Hooi, Christos Faloutsos, *Member, IEEE*, and Larry Pileggi, *Fellow, IEEE*

***Abstract*—Given sensor readings over time from a power grid, how can we accurately detect when an anomaly occurs? A key part of achieving this goal is to use the network of power grid sensors to quickly detect, in real-time, when any unusual events, whether natural faults or malicious, occur on the power grid. Existing bad-data detectors in the industry lack the sophistication to robustly detect broad types of anomalies, especially those due to emerging cyber-attacks, since they operate on a single measurement snapshot of the grid at a time. New ML methods are more widely applicable, but generally do not consider the impact of topology change on sensor measurements and thus cannot accommodate regular topology adjustments in historical data. Hence, we propose DYNWATCH, a domain knowledge based and topology-aware algorithm for anomaly detection using sensors placed on a dynamic grid. Our approach is accurate, outperforming existing approaches by 20% or more (F-measure) in experiments; and fast, averaging less than 1.7 ms per time tick per sensor on a 60K+ branch case using a laptop computer, and scaling linearly with the size of the graph.**

***Index Terms*—Anomaly detection, dynamic grid, graph distance, LODF, power system modeling.**

I. INTRODUCTION

MAINTAINING and improving the reliability of the electric power grid is a critically important goal. Estimates [1] suggest that reducing outages in the U.S. grid could save $49 billion per year, reduce emissions by 12 to 18%, while improving efficiency could save an additional $20.4 billion per year. Although grid operators and engineers work tirelessly to maintain reliability of the electric grids, many challenges persist. Climate change is increasing the frequency of natural disasters, resulting in higher rate of equipment failure. Adding to the climate risk is a new adversary in the form of cyber-intrusions that is capable of disrupting grid control and communication. This is evident from the recent reports of foreign hackers successfully penetrating control rooms of the U.S. power plants [2] and of cyber-attacks on the Ukrainian grid in 2015-2016 [3], [4] that brought down sections of the network causing damages worth billions of dollars.

A key tool that the grid operators use to safeguard against these failures, whether naturally occurring or malicious, involves the anomaly detection capabilities that are implemented in the grid control rooms. The primary purpose of these techniques is to help grid operators isolate faulty data from the healthy ones to result in accurate situational awareness, which further allows grid operators to take rapid corrective actions. In almost real-time, these methods can analyze measurement values, dynamics and other informative features to detect abnormal events including erroneous topology or measurements, while accommodating normal grid behaviors, including regular topology changes and power configuration adjustments.

In existing power grids, anomaly detection is performed within the Energy Management Systems (EMS) [5] that are installed in the control rooms. The EMS through Supervisory Control and Data Acquisition (SCADA) system collects two primary sources of data: i) online analog measurement data from various sensors such as remote terminal units (RTUs) and phasor measurement units (PMUs) and ii) status data of switching devices and circuit breakers on various devices such as lines and transformers. Both types of data are collected every few seconds: for instance analog measurements are updated every 10 s in PJM [6]; and status data are updated every 4 s in ISO-NE [7]. Upon processing, separate analysis units within the EMS are run to identify anomalies in measurements and topology.

AC state-estimation (ACSE) [8] along with bad-data detection (BDD) algorithms [9] is used today for anomaly detection on measurement data from RTUs and PMUs, via hypothesis test on output residuals. These are run every 1 to 10 minutes (every 5 minutes in ERCOT and US Midwest ISO (MISO), every 3 minutes in the U.K. grid and ISO-NE, and every 1–2 minutes in PJM [10]). The most widely used problem formulation for ACSE is the weighted-least-square (WLS) form [9], minimizing mean-squared measurement error. Some other formulations are designed to achieve intrinsic robustness against bad data, including least absolute value (LAV) based [11], least median of squares based [12], as well as iteratively reweighted least-squares based approaches [13]. Unfortunately when RTUs are included, these

Manuscript received 7 January 2021; revised 28 May 2021 and 29 September 2021; accepted 20 November 2021. Date of publication 6 December 2021; date of current version 19 August 2022. This work was supported in part by C3.ai Inc. and in part by the Microsoft Corporation. Paper no. TPWRS-00022-2021. *(Corresponding author: Shimiao Li.)*

Shimiao Li, Amritanshu Pandey, and Larry Pileggi are with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: shimiaol@andrew.cmu.edu; pandey.amritanshu@gmail.com; pileggi@andrew.cmu.edu).

Bryan Hooi is with the School of Computing and the Institute of Data Science, National University of Singapore, Singapore 119077 (e-mail: bhooi@comp.nus.edu.sg).

Christos Faloutsos is with the Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: christos@cs.cmu.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPWRS.2021.3132852.

Digital Object Identifier 10.1109/TPWRS.2021.3132852

methods generally suffer from difficult convergence due to non-linear measurement models. There have been recent attempts at convexification of the state-estimation problem [14]–[17]; however, several limitations persist. These include inability to detect coordinated attacks like false-data injection attack and high sensitivity to network topology errors.

On the other hand, events of topology change are detected by the network topology processor (NTP) [18][19], which transforms the input circuit breaker/switching status data into the bus-branch model in which network connectivity and meter locations are identified. However, existing operational NTP does not account for topology errors due to erroneous status data caused by communication, operator entry errors, cyber-attacks, etc. As a countermeasure, there exist research works that overcome some challenges of existing NTP. For instance, generalized topology processing [20] creates pseudo-measurements and applies hypothesis tests to detect topology anomalies. Another approach [21] applies hypothesis tests on WLS residuals from SE to detect topological errors, as an extended application of ACSE BDD. More recently, other advanced methods such as [15], [22], and [23] have been developed where TE and ACSE are merged together to perform estimation on a node-breaker model. This enables measurement error and topology error to be effectively identified and separated. However, challenges persist here as well, mostly due to the lack of efficient and scalable methods to handle the non-linearity with conventional measurements and the inability of these methods to detect topology anomalies during coordinated cyber-attacks.

These challenges with existing TE and ACSE can be addressed by using anomaly detection based on statistical behavior. Instead of analysing the well-defined measurement models in a snapshot, one can leverage statistical behaviors to extract some patterns from historical data. This can be helpful since most anomalies, either unexpected faults or cyber-attacks, usually disrupt the statistical consistency of the data stream, despite being invisible from one single snapshot. Existing behavioral anomaly detection methods can be broadly categorized into model-based (where an expectation of observation is obtained by fitting a mathematical model) [24]–[26], representation based [27], [28], graphical methods [29], and others (see Section II for related works). Generally, these families of methods are all directly applicable to the problem described in this paper. However, these methods either do not consider the impact of graph change on the observations [24], [26], and [27], or assume a static topology across the data stream [30]. This is problematic for grid problems since topological changes can happen frequently on a real grid (see Fig. 10(a) in Section V), and different configurations naturally cause differences in grid measurements. Hence, previous observations from disparate topologies and loadings may provide little value in assessing the anomalousness at a given time t. Without a selection of relevant data, methods may produce more false positives by falsely creating alarms when regular topology changes occur.

In light of these challenges, a more applicable way is to make the anomaly detection method *context-aware*, so that it indicates the expected patterns of behavior from data samples collected with relevant context of topology. With that viewpoint,

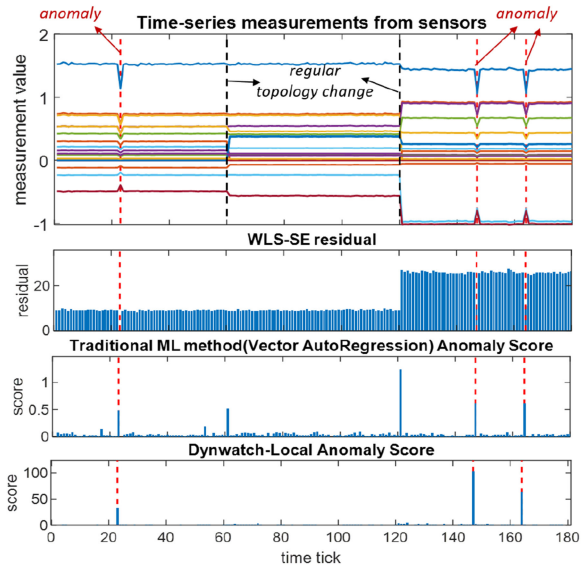


Fig. 1. Example of FDIA: SE BDD fails to give high residuals; traditional ML method VAR misclassifies regular topology changes as anomalies; only Dynwatch is able to detect all anomalies without False Positives (FP).

we propose a method that processes the time series of measurements and the time series of their topology to be *context-aware*. Intuitively, the method works by defining graph distances based on domain knowledge and estimating a reliable distribution of measurements at time t from the most relevant previous data. Then, an alarm is created if the measured value deviates greatly from the center of its distribution. The goal of the method is to directly consider the impact of topology on observations, so that regular topological changes are accommodated, while detecting any false measurement and topological errors. Furthermore, to account for large-scale grids, we develop a locally-sensitive variant, DYNWATCH-LOCAL.

Fig. 1 demonstrates the novelty of the proposed method. We apply a False Data Injection Attack [31] to a 14-bus network, a special type of cyber attack that modifies the measurement output of the selected sensors in the grid. We show that our proposed method is able to detect the anomalies without false positive (FP) outcomes, while the baseline methods are not. Details of this experiment are provided in Section IV-C.

The rest of the paper is organized as follows: Section II provides background of existing methods. Section III shows our proposed method. Section IV validates the method with experiment results. Finally, Section VI concludes the paper.

II. BACKGROUND AND RELATED WORK

A. Existing ML-Based Anomaly Detection Approaches

1) Time Series Anomaly Detection: Numerous algorithms exist for anomaly detection in univariate time series [32]. For multivariate time series, LOF [27] uses a local density approach. Isolation Forests [29] partition the data using a set of trees for anomaly detection. Other approaches use neural networks [26], distance-based [33], and exemplars [34]. However, none of these consider the graph structure.

TABLE I
COMPARISON OF RELATED APPROACHES: ONLY DYNWATCH SATISFIES ALL THE LISTED PROPERTIES

	Time Series	Graph-based	GridWatch	DYNWATCH
Graph Data		✓	✓	✔
Anomalies in Sensor Data	✓		✓	✔
Changing Graph		✓		✔
Locally Sensitive				✔

2) Anomaly Detection in Temporal Graphs: [35] finds anomalous changes in graphs using an egonet (i.e. neighborhood) based approach, while [36] uses a community-based approach. [37] finds connected regions with high anomalousness. [38] detects large and/or transient communities using Minimum Description Length. [39] finds change points in dynamic graphs, while other partition-based [40] and sketch-based [41] also exist for anomaly detection. However, these methods focus on detecting unusual communities or connections, while our approach has a very different goal of detecting disturbances which cause changes in sensor values.

3) Anomaly Detection with Domain Expert Knowledge: Domain-specific anomaly detectors based on optimal power flow [42], SE residual-based test (traditional SE BDI [9], and gross error detection [43], [44]) and TE [18], [20] already exist and are purely based on power system theories, yet they are typically limited to specific disturbances and attacks against the grid components. On the contrary, ML methods, as illustrated above, are more generally applicable; however, without a basic understanding of how the real-world grid operates, they are likely to perform poorly. Motivated by the pros and cons, many efforts [30], [45] have combined the benefits of the two, embedding the domain-knowledge in general ML methods. Such methods with domain knowledge have been shown to have higher performance (see [30], [45]) but still do not fully consider the dynamic nature of the electric grid.

To summarize, the major contribution of DYNWATCH when compared with existing methods are summarized in Table I.

B. Handling Redundant Data

Transmission grids, in order to be observable, have a large number of RTUs and PMUs installed. For the anomaly detection algorithm developed in this paper, processing the large volume of redundant data for anomaly detection is unnecessary and computationally prohibitive. This is because each sensor predominantly captures the relative information of its neighboring sensors as well. Therefore, to create a proper input for anomaly detection, pre-processing techniques can be deployed: Principal Component Analysis (PCA) [46] creates a low-dimensional representation by extracting uncorrelated directions; projection pursuit [47] reduces the input to a low-dimensional projected time series that optimizes the kurtosis coefficient; Independent Component Analysis (ICA) [48] identifies a subset of independent variables. Alternative techniques like cross-correlation analysis [49] also help create a low-dimensional input. A detailed survey regarding dimensionality reduction can be found in [50]. Rather than transforming the redundant input, other algorithms for sensor placement [51] are also applicable, by suggesting the best several locations of sensors to be installed and providing observability. [30] has shown the selection of a small number of sensor locations with a provably near-optimal probability of detecting an anomaly.

C. Standard Graph Distance/Similarity Measures

Many graph distance/similarity measures have been proposed in the past that relate to anomaly detection in dynamic graphs. A survey of such measures can be found in [52], [53]. Most of these measures fit in one of the following categories:

1) **Graph isomorphism and its generalizations:** examples include Maximum Common Subgraph (MCS) distance [54], Graph Edit Distance (GED) [54], and variants of GED [55], etc.
2) **Aggregate statistical measure:** preferred for measure for larger graphs, examples include diameter distance [56], clustering based measures [57], and degree distribution [58], etc.
3) **Iterative methods based on the structural similarity of local neighborhoods:** This type of method exchanges node/edge similarities until convergence and computes the similarity between two graphs by coupling the similarity scores of nodes and edges [52].
4) **Other complex feature-based measures:** examples include graph kernel-based similarities [59], modality distance [60], median graph distance [60], etc.

While all of the above graph distance measures have unique advantages, none of them are designed for grid-specific challenges, nor do they capture the implicit physics of the power grid graph. Section III-C will provide a more detailed discussion about grid-specific challenges and develop a novel graph distance measure to meet the needs of the grid anomaly detection.

D. Background: Line Outage Distribution Factor (LODF)

Line Outage Distribution Factor (LODF) is a sensitivity measure of how much an outage on a line affects real power flow on other lines in the system [61]. This factor can be easily and efficiently calculated by assuming a DC power flow model with lossless lines or a linearized AC power flow model around the operating point and is commonly used to estimate the linear impact of line outage. For an outage on line k, LODF d_l^k gives the ratio between power change Δf_l on an observed line l and the pre-outage real power f_k on the outage line k.

$$d_l^k = \frac{\Delta f_l}{f_k}$$

III. PROPOSED DYNWATCH ALGORITHM

A. Preliminaries

Table II shows the symbols used in this paper.

TABLE II
SYMBOLS AND DEFINITIONS

Symbol	Interpretation
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	Input graph
$\mathcal{S}$	Subset of nodes to place sensors on
n	Number of nodes
s	Number of scenarios
$\mathcal{N}_i$	Set of edges adjacent to node i
$V_i(t)$	Voltage at node i at time t
$I_e(t)$	Current at edge e at time t
$s_{ie}(t)$	Power w.r.t. node i and edge e at time t
$\Delta s_{ie}(t)$	Power change: $\Delta s_{ie}(t) = s_{ie}(t) - s_{ie}(t-1)$
$X_i(t)$	Sensor vector for scenario i at time t

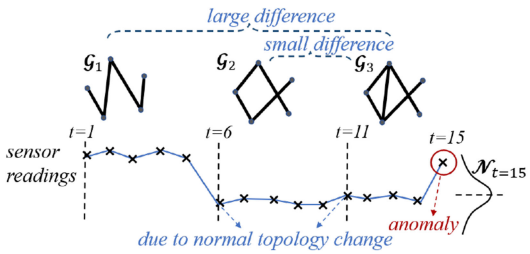


Fig. 2. Simple motivating example: anomaly detection under changing graphs using DYNWATCH.

We are given a dynamic graph (grid) $\mathcal{G}_t = (\mathcal{V}(t), \mathcal{E}(t))$ at each time tick t, where $\mathcal{V}(t)$ denotes the set of nodes (active grid buses), and $\mathcal{E}(t))$ denotes the set of edges (active grid branches). Also we have a fixed set of sensors $\mathcal{S} \subseteq \mathcal{V}$. Each sensor installed on node i can obtain PMU or RTU measurements at each time t.

For each sensor on node i, we obtain the power flows on all lines adjacent to node i, as observed in [30] that using power (rather than current) provides better anomaly detection in practice. For any PMU bus i and edge $e \in \mathcal{N}_i$, define the power w.r.t. i along edge e as $s_{ie}(t) = V_i(t) \cdot I_e(t)^*$, where * is the complex conjugate.

Topology determination: At any time t, the given dynamic graph $\mathcal{G}(t)$ can be considered a 'reference topology'. The observed measurements should be consistent with the reference topology if there are no anomalous measurements and topology. In this work, we use the output from topology estimation (NTP) in the grid energy management system (EMS) as the reference topology. It is a valid reference because the operator believes that this is the best estimate of the grid topology at any time. Notably, this topology is not assumed to be perfect and accurate. Topological errors, along with false measurements, are to be detected by the proposed method.

B. Motivation and Method Overview

Consider the simple power grid shown in Fig. 2, which evolves over time from $\mathcal{G}_1$ to $\mathcal{G}_2$ to $\mathcal{G}_3$. For simplicity, assume that we have a single sensor, from which we want to detect anomalous events. How do we evaluate whether the current time point ($t = 15$) is an anomaly? If the graph had not been changing, we could simply combine all past sensor values to learn a distribution of normal behavior (e.g. fitting a Gaussian distribution as in $\mathcal{N}_{t=15}$), then evaluate the current time point using this Gaussian distribution (e.g. in terms of the number of standard deviations away from the mean).

In the changing graph setting, we still want to learn a model of normal behavior ($\mathcal{N}_{t=15}$), but while taking the graph changes into account. Note that $\mathcal{G}_2$ and $\mathcal{G}_3$ are only slightly different, while $\mathcal{G}_1$ and $\mathcal{G}_3$ are very different. Hence, the sensor values coming from $\mathcal{G}_2$ (i.e. time 6 to 10) should be taken into account more highly when constructing $\mathcal{N}_{t=15}$, as compared to those from $\mathcal{G}_1$. Intuitively, the sensor values from $\mathcal{G}_1$ are drawn from a very different distribution from the current graph, and thus should not influence our learned model $\mathcal{N}_{t=15}$. In general, the more similar a graph is to the current graph, the more we should take its sensor values into account when learning our current model. This motivates the 3-step process we use:

1) **Graph Distances:** Measure the distance between each past graph and the current graph.
2) **Temporal Weighting:** Weight the past sensor data, where data from graphs that are similar to the current one are given higher weight.
3) **Anomaly Detection:** Learn a distribution of normal behavior ($\mathcal{N}_t$) from the weighted sensor values, and measure the anomalousness at the current time based on its deviation from this distribution.

To further clarify the motivation and methodology, we provide an informal definition for the anomaly detection problem and a statistical definition of the anomaly pattern.

Definition III.1 (Dynamic electric grid anomaly detection problem): Given time series data of sensor observations on a set of sensors $\{s\}$, and a time series of topologies, find (1) the timestamps that correspond to an anomaly pattern (2) the top-k sensor locations that contribute most to the anomaly pattern.

Definition III.2 (Anomaly pattern): At any time t, given a time series of observations $X(t), X(t-1), \ldots X(t-W)$, a time series of topologies, a graph distance measure $D(Gi, Gj)$, and a detection threshold τ, we assign different (trust) weights to observations at $t-1, \ldots, t-W$, based on $D(G_t, G_{(t-1)}), \ldots, D(G_t, G_{(t-W)})$: the larger the distance, the lower the weight. This provides a statistical distribution of $X(t)$, parameterized by weighted median $\mu(t)$ and weighted IQR(t). The instance t is predicted as anomalous if $X(t)$ is an outlier of the distribution, i.e., $|X(t) - \mu(t)| > \tau \cdot IQR(t)$.

In electric grids, the target anomalies correspond to topological errors (unexpected topology changes unknown to the operator) and measurement errors. These are the types of anomalies that our method is proposed for and is likely to detect based on empirical results provided at the end of the manuscript.

In the following sections, we first introduce our domain-aware graph distance measure based on Line Outage Distribution Factors (LODF) [61]. Then, we describe our temporal weighting and anomaly detection framework, which flexibly allows for any given graph distance measure. Finally, we present an alternate

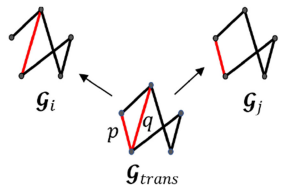


Fig. 3. Transition state of two graphs: the union of two graphs.

distance measure that is locally sensitive, i.e., it accounts for the local neighborhood around a given sensor.

C. Proposed Graph Distance Measure

In this section, we describe our proposed graph distance measure to calculate the distance $D(\mathcal{G}_i, \mathcal{G}_j)$ between any pair of graphs. For ease of understanding, the rest of Section III uses the example of anomaly detection at $t = 15$ in Fig. 2 as an extended case study, but our approach can be easily extended to the general case.

Section II-C has provided a short review of existing graph distance measures, but these measures do not apply directly to the grid-specific challenge in this paper. For an anomaly detection algorithm to work well on the power grid applications, the choice of graph distance needs to consider **problem-specific challenges** along with desirable properties for an anomaly detection algorithm (scalability, sensitivity to change, and **'importance-of-change awareness'**). One grid-specific challenge is that the ideal graph distance should capture 'grid physics' rather than only graph structural changes. Specifically, the distance should be sensitive to the redistribution of power flow, not only the addition/deletion of nodes/edges, since the anomaly information is extracted from power flow measurements. Meanwhile, as the 'importance-of-change awareness' indicates, grid changes that cause big shifts in power flow (measurements) should result in larger graph distances, than changes that cause minor power impact. Unfortunately, none of the classical graph distances can capture the physics of the power flow and quantify the impact of graph change in terms of power. To handle this, this work proposes a novel design of graph distance by making use of the power sensitivity factor.

Intuitively, the goal is for our graph distance to represent **redistribution of line power flow**. Critical changes in topology result in large redistributions of power. Thus, the graph distance arising from a topology change should be large if the changed edges can potentially cause large amounts of power redistribution.

Hence, given two graphs $\mathcal{G}_i(\mathcal{V}(i), \mathcal{E}(i))$ and $\mathcal{G}_j(\mathcal{V}(j), \mathcal{E}(j)$ with different topology, we first define a transition state (see Fig. 3) which takes the union of the two graphs:

$$\mathcal{G}_{trans} = (\mathcal{V}(i) \cup \mathcal{V}(j), \mathcal{E}(i) \cup \mathcal{E}(j))$$

Then the topology changes from $\mathcal{G}_i$ to $\mathcal{G}_j$ can be considered as different line deletions from their base graph $\mathcal{G}_{trans}$. For each single line deletion, e.g. line p, we define its contribution x_p to graph distance by taking the average of its power impacts on all other lines as measured by LODF:

$$x_p = \frac{1}{|\mathcal{E}(i) \cup \mathcal{E}(j)|} \sum_{l \in \mathcal{E}(i) \cup \mathcal{E}(j) \setminus \{p\}} (|d_l^p|)$$

where $|\mathcal{E}(i) \cup \mathcal{E}(j)|$ denotes the cardinality of set $\mathcal{E}(i) \cup \mathcal{E}(j)$, d_l^p denotes the LODF coefficient with p as outage line and l as observed line.

Then graph distance $D(\mathcal{G}_i, \mathcal{G}_j)$ is given by summing up the contributions of different line deletions from the base graph:

$$D(\mathcal{G}_i, \mathcal{G}_j) = \sum_{p \in (\mathcal{E}(i) - \mathcal{E}(j)) \cup (\mathcal{E}(j) - \mathcal{E}(i))} x_p$$

where $\mathcal{E}(i) - \mathcal{E}(j) = \{p | p \in \mathcal{E}(i), p \notin \mathcal{E}(j)\}$ and accordingly, $(\mathcal{E}(i) - \mathcal{E}(j)) \cup (\mathcal{E}(j) - \mathcal{E}(i))$ denotes all the edge changes between the two graphs.

This definition uses LODF as a measure of the impact on power flow of the removal of line p. Hence, edges with high LODF to many other edges can potentially cause greater changes in power flow, and thus our graph distance measure places greater importance on these edges. Appendix A demonstrates the effectiveness of the LODF-based graph distance by comparing it against traditional distance measures for anomaly detection.

Algorithm 1: Temporal Weighting Framework at Time $t = 15$ (see Example in Fig. 2).

Input: Graph distance $D(\mathcal{G}_1, \mathcal{G}_3)$, $D(\mathcal{G}_2, \mathcal{G}_3)$, $D(\mathcal{G}_3, \mathcal{G}_3)$; sensor data $s_i(t)$ with $t = 1, 2, \ldots 15$, $i = 1, 2, \ldots, N_{sensor}$.

Output: Anomaly score $A(15)$.

1: **Extend graph distance to tick-wise distance.** Each previous time tick is given a distance d_t according to the graph it comes from:

$$d_t = \begin{cases} D(\mathcal{G}_1, \mathcal{G}_3) & \text{for } t = 1, 2, \ldots, 5 \\ D(\mathcal{G}_2, \mathcal{G}_3) & \text{for } t = 6, 7, \ldots, 10 \\ D(\mathcal{G}_3, \mathcal{G}_3) & \text{for } t = 11, \ldots, 14 \end{cases}$$

2: **Temporal Weighting:** Use $d_1, \ldots d_{14}$ to assign weights w_1,..., w_{14} to the past sensor data using Algorithm 2.

D. Proposed Temporal Weighting Framework

In this section, we assume that we are given any distance measurements $D(\mathcal{G}_i, \mathcal{G}_j)$ between any pair of graphs $\mathcal{G}_i$ and $\mathcal{G}_j$, and explain how to use them to assign weights to each previous sensor data. This procedure can take the LODF-based distance defined in the previous subsection as input, but also allows us to flexibly use any given graph distance measure. The proposed Temporal Weighting is given in Algorithm 1.

For the purpose of utilizing previous data from a series of dynamic graphs, **Temporal Weighting** plays an important role. The resulting weights directly determine how much information to extract from each previous record, thus requiring special care. Intuitively, the weights should satisfy the following principles:

- The larger the distance d_t, the lower the weight w_t. This is because high d_t indicates that time tick t is drawn from a very different graph from the current one, and thus should not be given high weight when estimating the expected distribution at the current time
- Positivity and Normalization: $\sum_t w_t = 1, w_t \geq 0$

To satisfy these conditions, we use a principled optimization approach based on **bias-variance trade-off**. Intuitively, the problem with using data with high d_t is **bias**: it is drawn from a distribution that is very different from the current one, and that can be considered a biased sample. We treat d_t as a measure of the amount of bias. Hence, given weights $w_1, \ldots, w_{14}$ on previous data (in Fig. 2 example), the total bias we incur can be defined as $\sum_{t\in\{1,\ldots,14\}} w_t d_t$.

We could make the bias low simply by assigning positive weights to only time points from the most recent graph. However, this is still unsatisfactory as it results in a huge amount of **variance**: since very little data is used to learn $\mathcal{N}_{t=15}$, the resulting estimate has high variance. Multiplying a fixed random variable by a weight w_t scales its variance proportionally to w_t^2. Hence, given weights $w = [w_1, w_2, \cdots]$, the total amount of variance is proportional to $\frac{1}{2}w^T w$, which we define as our variance term.

We thus formulate the following optimization problem as minimizing the sum of bias and variance, thereby balancing the goals of low bias (i.e. using data from similar graphs) and low variance (using sufficient data to form our estimates). We formulate the problem as:

$$\min_w \sum_t w_t d_t + \frac{1}{2}w^T w$$

subject to

$$\sum_t w_t = 1$$

$$w_t \geq 0, \forall t$$

By writing out its Lagrangian function:

$$L(w, \lambda, u) = d^T w + \frac{1}{2}w^T w + \lambda(1 - \sum_t w_t) - u^T w$$

and applying KKT conditions, we can see the optimal primal-dual solution (w, λ^*, u^*) must satisfy:

$$d_t + w_t - \lambda^* - u_t^* = 0$$

Since we have $d_t \geq 0$, by further manipulation we have:

$$w_t = \max\{\lambda^* - d_t, 0\}$$

Moreover, there is a unique choice of λ^* such that the resulting weights w_t sum up to 1. This w_t against d_t relationship is shown in Fig. 4. This result is intuitive: as d_t increases, the resulting weight we assign w_t decreases, and if d_t passes a certain threshold, it becomes large enough so that any reduction in variance it could provide is more than offset by its large bias, in which case we assign it a weight of 0.

Our Temporal Weighting algorithm is in Algorithm 2. During implementation, we adjust the relative importance of bias and variance by normalizing and scaling the graph distances (scaling factor 0.005 works well based on our empirical observation).

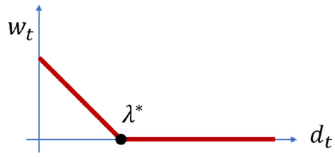


Fig. 4. $w_t - d_t$ relationship.

Algorithm 2: Computing Temporal Weights w_t.

Input: distance d_t, with $t = 1, 2, \ldots, N$
Output:weights w_t, with $t = 1, 2, \ldots, N$

1: Compute the unique λ^* that satisfies:

$$\sum_{t\in\{1,2,\ldots N\}} \max\{\lambda^* - d_t, 0\} = 1$$

2: Get weights w_t:

$$w_t = \max\{\lambda^* - d_t, 0\}$$

E. Proposed Anomaly Detection Algorithm

Having obtained our weights w_t, the remaining step is to compute our anomaly score, as shown in Algorithm 3.

We focus on 3 metrics from sensor data as indications of power system anomalies. These metrics were studied in [30] and found to be effective for detecting anomalies in power grid sensor data. In our setting, recall that for each sensor, we can obtain Δs_i that contains changes of real and reactive power on the adjacent lines, over time. The 3 metrics are:

- *Edge anomaly metric:* $X_{edge,i}(t) = \max_{l\in E_{adj}} \Delta s_{i,l}$ which measures the maximum line flow change among lines connected to the sensor. Let E_{adj} denote the set of lines connected to sensor i:
- *Average anomaly metric:* $X_{ave,i}(t) = \text{mean}\{\Delta s_{i,l} \mid l \in E_{adj}\}$, which measures the average line flow change on the lines connected to the sensor:
- *Diversion anomaly metric:* $X_{div,i}(t) = \text{std}\{\Delta s_{i,l} \mid l \in E_{adj}\}$, which measures the standard deviation of line flow change over all lines connected to the sensor:

Intuitively, for each metric, we want to estimate a model of its normal behavior. To do this, we compute the weighted median and interquartile range (IQR)[1] of the detection metric, weighting the time points using our temporal weights $w_1, \ldots, w_t$. (Weighted) median and IQR are preferred choice of distribution parameters over the mean and variance for anomaly detection since they are robust measures of central tendency and statistical dispersion (i.e. they are less likely to be impacted by outliers) [62]. We can then estimate the anomalousness of the current time tick by computing the current value of a metric, then subtracting its weighted median and dividing by its IQR. The exact steps are given in Algorithm 3.

[1]IQR is the difference between 1st and 3rd quartiles of the distribution, and is commonly used as a robust measure of spread.

Algorithm 3: Anomaly Detection (see Figure 2).

Input: Temporal weights w_t; sensor data $s_i(t)$ with $t = 1, 2, ...15$, $i = 1, 2, ...N_{sensor}$

Output: anomaly score $A(15)$

for $i \leftarrow 1$ *to* N_{sensor} **do**

  Compute weighted median and IQR:

  $\mu_{edge} = \text{Weighted Median}\{X_{edge,i}(t) \mid t = 1, ..., 14\}$

  $IQR_{edge} = \text{Weighted IQR}\{X_{edge,i}(t) \mid t = 1, ..., 14\}$

  weighted by $w_1, ...w_{14}$ (similarly for X_{ave}, X_{div}).

  Calculate sensor-wise anomaly score at t=15:

  $$a_i(15) = \max_{metric \in \{edge, ave, div\}} \frac{X_{metric,i}(15) - \mu_{metric}}{IQR_{metric}}$$

Calculate anomaly score for target time tick, as the max score over sensors:

$$A(15) = \max_{i \in \{1,...,N_{sensor}\}} a_i(15)$$

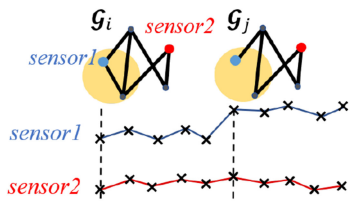


Fig. 5. Simple motivating example: DYNWATCH-LOCAL. The two graphs are very different within the yellow localized region. However, sensor 2 is far away from the yellow region and thus experiences no changes.

F. Proposed Locally Sensitive Distance Measure

Motivation: In the previous section, we computed a single distance value $D(\mathcal{G}_i, \mathcal{G}_j)$ between any pair of graphs. However, consider two graphs $\mathcal{G}_i$ and $\mathcal{G}_j$ in Fig. 5 that are very different due to a small yellow localized region (e.g. in a single building that underwent heavy renovation). Hence, $D(\mathcal{G}_i, \mathcal{G}_j)$ is large, indicating not to use data from $\mathcal{G}_i$ when we analyse a time tick under $\mathcal{G}_j$. However, from the perspective of a single sensor s (sensor 2) far away from the localized region, this sensor may experience little or no changes in the power system's behavior, so that data from graph $\mathcal{G}_i$ may have a similar distribution as data from graph $\mathcal{G}_j$, and so for this sensor (sensor 2) we can still use data from $\mathcal{G}_i$ to improve anomaly detection performance. Hence, rather than computing a single distance $D(\mathcal{G}_i, \mathcal{G}_j)$, we compute a separate **locally-sensitive** distance $D_s(\mathcal{G}_i, \mathcal{G}_j)$ specific to each sensor, which measures the amount of change between graphs $\mathcal{G}_i$ and $\mathcal{G}_j$ in the 'local' region to sensor s. Clearly, the notion of 'local regions' must be carefully defined: we will define them based on LODF, recalling that LODF measures how much changes on one edge affect each other edge.

Intuitively, the local distance between two graphs with respect to sensor s is large if the changed edges can potentially cause large power change nearby the sensor. Hence, given two graphs $\mathcal{G}_i(\mathcal{V}(i), \mathcal{E}(i))$ and $\mathcal{G}_j(\mathcal{V}(j), \mathcal{E}(j))$ with their transition state $\mathcal{G}_{trans}$ and a sensor s of interest, the local graph distance contribution y_p of line p with respect to sensor s can be calculated by multiplying the whole-grid-wide contribution x_p with a weighing factor c_p^s. This c_p^s coefficient filters the power impact for sensor s using the maximum power impact of line deletion on lines around this sensor:

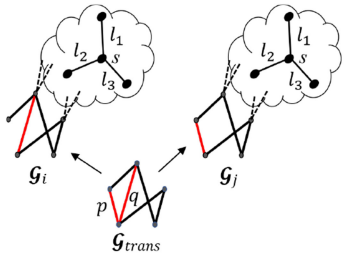


Fig. 6. Local graph distance: the adjacent lines connected to each sensor s are considered.

$$c_p^s = \max_{l \in \mathbb{E}_{sensor}(s)} |d_l^p|$$

$$y_p = x_p c_p^s$$

where $\mathbb{E}_{sensor}(s)$ denotes the set of edges around sensor s (e.g. in Fig. 6, $\mathbb{E}_{sensor}(s) = \{l_1, l_2, l_3\}$), and d_l^p denotes the LODF with p as outage line and l as observed line.)

Then, as before, the local graph distance with respect to sensor s is defined by summing up the local graph distance contributions of different line deletions from the graph:

$$D_s(\mathcal{G}_i, \mathcal{G}_j) = \sum_{p \in (\mathcal{E}(i) - \mathcal{E}(j)) \cup (\mathcal{E}(j) - \mathcal{E}(i))} y_p$$

G. Statistical Error Analysis

This section focuses on a quantitative analysis of the performance of our method, through statistical error analysis.

Let T denote the width of time window for analysis, then for $\forall$ sensor s, the anomalousness of its observation x_{T+1} is evaluated based on its previous data $x_1, x_2, \ldots, x_T$. The anomaly detection method works by assigning weights $w_1, w_2, \ldots, w_T$ ($w_t \geq 0, \forall t, \sum_{t=1}^{T} w_t = 1$) to all the previous observations and an alarm is created if x_{T+1} deviates $\sum_{t=1}^{T} w_t x_t$ by a certain threshold.

Here we investigate the properties of statistical error based on the following definitions and assumptions:

Assumption III.1 (Temporal independence): For any sensor s and time t, its measured data x_t is drawn from a Gaussian distribution $P(x_t) = N(\mu_t, \sigma^2)$ independently from other time ticks, where σ^2 accounts for all uncertainties caused by measurement noise, load/generation variation, weather uncertainty, etc.

Assumption III.2 (identical distribution conditioned on topology): Given a certain topology G and $\forall$ sensor s, all data of s under the same topology G are drawn independently from the same distribution $P(x_t|G) = N(\mu_G, \sigma^2)$ (i.e., for any two time ticks t_1, t_2 with the same topology G, we have $\mu_{t_1} = \mu_{t_2} = \mu_G$.)

Definition III.3 (Optimal graph distance): For any time-series data $x_1, x_2, \ldots, x_T$ of a sensor s and its latest observation

x_{T+1}, d_t denotes the graph distance between the graph at time t and the graph at $T+1$, i.e., $d_t = D(G_t, G_{T+1}), d_t \geq 0, \forall t$. Then the optimal graph distance d_t^* for $\forall t$ satisfies $|\mu_t - \mu_{T+1}| = |\mu_{G_t} - \mu_{G_{T+1}}| \propto d_t^*$, or equivalently, $\exists$ constant c such that $|\mu_t - \mu_{T+1}| = c \cdot d_t^*$.

We first demonstrate that the statistical error can be bounded:

Theorem III.1 (Error bound): Based on Assumption III.1, III.2 and Definition III.3, the statistical error $\mathbb{E}_{x_1,x_2,\ldots,x_T,x_{T+1}}[(\sum_{t=1}^{T} w_t x_t - x_{T+1})^2]$ with $w_t \geq 0, \forall t$ and $\sum_{t=1}^{T} w_t = 1$, satisfies:

$$\sigma^2 \leq \mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] \leq (1+\max_t w_t)\sigma^2 + c \max_t d_t^*$$

Upon obtaining the error bound, here are some intuitive explanations of the upper bound being dependent on $\max_t w_t$ and $\max_t d_t^*$:

- $\max_t w_t$: large value for this term indicates that the estimation method depends heavily on a particular prior data point with weight w_t. This can lead to overfitting and as a result higher error (upper bound) due to high variance.
- $\max_t d_t^*$: large value for this term indicates that a prior data point from a very different distribution has been used for estimation, which can lead to higher error (upper bound) due to high bias.

Another question of interest to us is the properties in the limit of infinite data:

Theorem III.2 (Unbiased estimation under infinite data): In the limit of infinite data, the statistical error limits at the lower bound:

$$\mathbb{E}_{x_1,x_2,\ldots,x_T,x_{T+1}}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] = \sigma^2$$

and an unbiased estimate of the true distribution $x_{T+1} \sim N(\mu_{T+1}, \sigma^2)$ is obtainable using the previous samples, i.e.,

$$\mathbb{E}\left[\sum_{t=1}^{T} w_t x_t\right] = \mu_{T+1}$$

$$\mathbb{E}\left[\frac{1}{T-1}\sum_{t}\left(x_t - \frac{\sum_t x_t}{T}\right)^2\right] = \sigma^2$$

Detailed proofs of the two theorems are included in Appendix A.

IV. Experiments

We design experiments to answer the following questions:

- **Q1. Anomaly Detection Performance:** how accurate is the anomaly detection from our method compared to other ML baselines?
- **Q2. Scalability:** how do our algorithms scale with the graph size?
- **Q3. Practical Benefits:** how can our algorithm enhance the standard practices (SE BDD) in today's grid operator?

Our code and data are publicly available at https://github.com/bhooi/dynamic.git. Experiments were done on a 1.9 GHz Intel Core i7 laptop, 16 GB RAM running Microsoft Windows 10 Pro.

Case Data: We use 2 test cases: CASE2383is an accurate reconstruction of part of the European high voltage network, and ACTIVSG25 K is a synthetic network that mimics the Texas high-voltage grid in the U.S. The ACTIVSG25 K represents a similarly sized system as the PJM (the largest independent service operator (ISO) in the U.S.) grid, which contains around 25 to 30 k buses [63].

Selection of sensors and network observability: Due to the spatial impact of grid anomalies and the efficacy of anomaly metrics, full observability [64][65] and optimal sensor placement for observability [66] are not necessary for Dynwatch to perform. However, access to more sensors as inputs alongside optimal sensor selection [30] can improve the detection performance and help with localization of anomaly. To be conservative, for these experiments, we select random subsets of sensors (of varying sizes) as input. The good performance even with randomly selected sensor measurements validates the effectiveness of our method in selecting relevant time frames from historical data.

Threshold tuning: For a fair comparison of different methods, our experiment section, without using any threshold, compares the top K anomalies scored by each algorithm, where K is the number of anomalies simulated. However, in practical use, a detection threshold is required for the algorithm to identify an anomaly. A proper threshold can be either a fixed threshold from an empirical value or domain-knowledge or a learned threshold to facilitate optimal decision making. In particular, optimal threshold tuning needs to take **class imbalance and asymmetric error** into account. Since only a minority of instances are expected to be abnormal, there is an unbalanced nature of data. Moreover, as grid applications are safety-critical, mislabeling an anomaly as normal, i.e., false negatives (FN), could cause fatal consequences, while false positives (FP) which cause loss of 'fidelity' are less serious. Proper techniques for tuning a threshold **offline** include:

1) Calculating the evaluation metric (e.g. F-measure which quantifies the balance of precision and recall) for each threshold to select the one that maximizes the metric.
2) Plotting the ROC curve or precision-recall curve to select the threshold that gives the optimal balance.
3) A cost approach that, when the cost of FP, FN, TP, TN are available, minimizes the average overall cost of a diagnostic test, yet domain-specific knowledge is needed for reasonable quantification of the costs.

A. Q1. Anomaly Detection Accuracy

In this section, we compare DYNWATCH against baseline anomaly detection approaches, while varying the number of sensors in the grid.

1) Experimental Settings: Starting with a particular test case as a base graph G, we first create 20 different topology scenarios where each of them deactivates a randomly chosen branch in the base graph. These subsequent 20 network topologies represent

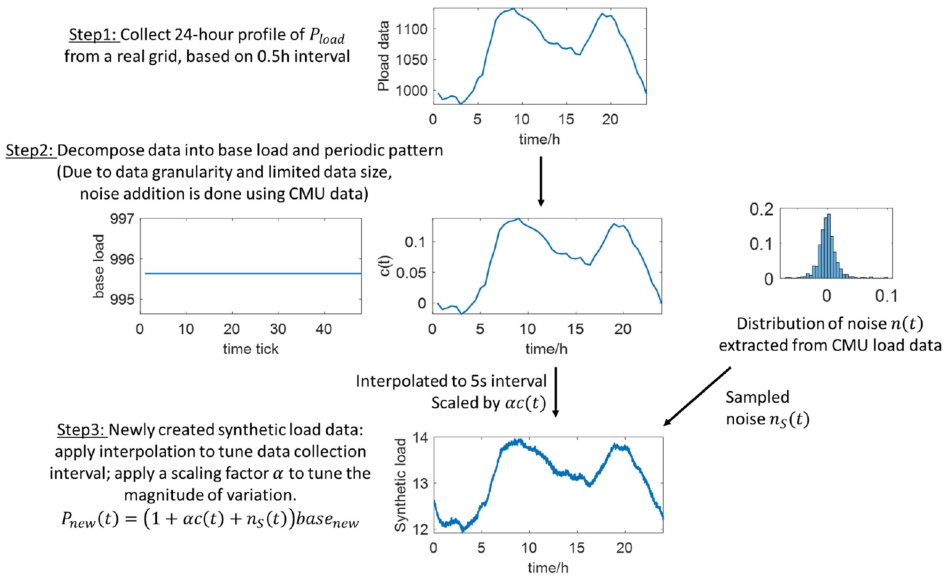


Fig. 7. Methodology for construction of synthetic data from a real utility-provided load data (See Section V-A). The generated data is used for evaluating performance in Section IV where the 1200 time-ticks are used to model the time-series data of grid operation over 1 h 40 min.

the dynamic grid with topology changes due to operation and control. Then for each topology scenario, we use MatPower [63], a standard power grid simulator, to generate 60 sets of synthetic measurements based on the load characteristics described in the following paragraph. As a result, the multivariate time series with $20 \times 60 = 1200$ time ticks mimics the real-world data setting where sensors receive measurements at each time tick t, and the grid topology changes every 60 time ticks. Finally, we sample 50 random ticks out of 1200 as times when anomalies occur. Each of these anomalies is added by randomly deleting an edge on the corresponding topology.

Following [30], to generate an input time series of loads (i.e. real and reactive power at each node), we use the patterns estimated from two real datasets:

- Carnegie Mellon University (CMU) campus load data recorded for 20 days from July 29 to August 17, 2016;
- Utility-provided 24 h dataset of a real U.S. grid. See Section V-A for dataset description and statistical findings related to it.

to create synthetic time-series load based on a 5 s interval, with the magnitude of daily load variation scaled to a predefined level, and with added Gaussian noise sampled from the extracted noise distribution [67]. The detailed data generation process is shown in Fig. 7.

Given this input, each algorithm then returns a ranking of the anomalies. We evaluate this using standard metrics, AUC[2] (area under the ROC curve) and F-measure[3] ($\frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$), the latter computed on the top 50 anomalies output by each algorithm.

2) Baselines: Dynamic graph anomaly detection approaches [35], [36], [38], [68] cannot be used as they consider graph structure only, but not sensor data. [37] allows sensor data but requires graphs with fully observed edge weights, which is inapplicable as detecting failed power lines with all sensors present reduces to checking if any edge has current equal to 0. Hence, instead, we compare DynWatch to GridWatch [30], an anomaly detection approach for sensors on a static graph, and the following multidimensional time-series based anomaly detection methods: Isolation Forests [29], Vector Autoregression (VAR) [24], Local Outlier Factor (LOF) [27], and Parzen Window [69]. Each uses the currents and voltages at the given sensors as features. For VAR, the norms of the residuals are used as anomaly scores; the remaining methods return anomaly scores directly. For Isolation Forests, we use 100 trees (following the defaults in scikit-learn [70]). For VAR, following standard practice, we select the order by maximizing AIC. For LOF we use 20 neighbors (following the default in scikit-learn), and we use 20 neighbors for Parzen Window.

As shown in Fig. 8, DynWatch clearly outperforms the baselines on both metrics, having an F-measure of $> 20\%$ higher than the best baseline.

B. Q2. Scalability

In this subsection, we seek to analyze the scalability of our DynWatch and DynWatch-Local. In reality, PJM, the largest ISO in the U.S., runs ACSE on a 28 k bus model, performed every 1 min [10], thus any anomaly detection algorithm that takes significantly less than 1 min may provide valuable information to prevent wrong control decisions in real-time. The following results demonstrate the proposed method's capability to achieve this.

Here, we generate test cases of different sizes by starting with the case2383 case and duplicating it $3, 4, 5, \ldots, 12$ times. After each duplication, edges are added to connect each node with its counterpart in the last duplication, so that the whole grid is connected. Then for each testcase, we generate 20 dynamic grids

[2]AUC is the probability of correct ranking of a random "positive"-"negative" pair.

[3]F-measure is a trade-off between precision and recall.

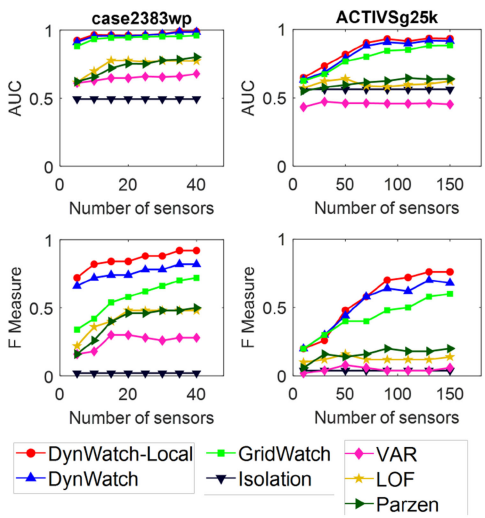


Fig. 8. Experiment results by AUC and F-measure.

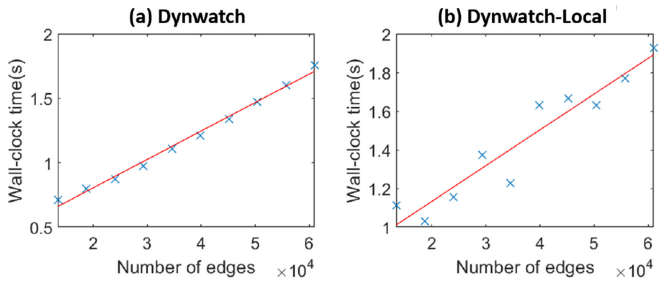


Fig. 9. Our algorithms scale linearly: wall-clock time of (a) DYNWATCH; and (b) DYNWATCH-LOCAL against number of edges, when detecting on all 1200 time ticks. The red lines are best-fit regression lines.

and sensor data with 1 randomly chosen sensor and 1200 time ticks, following the same settings as the previous sub-section. Finally, we measure

Fig. 9 shows that our method is fast: even on a large case with 60k+ branches, both methods took less than 2 s to apply anomaly detection on all 1200 time ticks of the sensor, corresponding to an average of less than 1.7 ms per time tick per sensor. The ACTIVSG25 K (similar to the largest real-time network in the US) has 32k+ branches, and thus run-time of anomaly detection at each time t will be significantly less than 1 min. The figure also shows that our methods scale close to linearly with the grid size.

C. Q3. Practical Benefits

In this section, we explore how the proposed Dynwatch algorithm can improve the performance of the standard residual-based ACSE bad-data detection (BDD) method, by testing a type of grid-specific anomaly that SE BDD is known to fail against False Data Injection Attack (FDIA).

False Data Injection Attack (FDIA) [31] is a cyber-attack in which attackers manipulate the value of measurements according to the grid physical model such that the SE outputs incorrect grid estimates while ensuring that its residual does not change by much (ideally remains unchanged).

In this experiment, we construct an attack on a 14 bus network to mislead the operator into thinking that the load reduces by 20%. For any anomalous time tick t, this is implemented by simulating power flow with the reduced load and generating measurements based on that.

The time-series measurement data and a comparison of anomaly scores are shown in Fig. 1. Results show that the ACSE residual, which is metric for BDD reduced in value when anomalies occur (see the residual decrease in Fig. 1 during anomalous operation shown by the dotted red line), implying that the standard SE BDD, along with any other residual-based method, will not be able to detect this coordinated attack.

In addition to standard ACSE BDD, we also implemented the auto-regression (VAR) method to detect grid anomalies. As the VAR algorithm does not consider dynamic graph properties of the power grid, it tends to create alarms on all abrupt measurement changes. This will easily lead to false positives since regular topological changes also result in sudden temporal change. This can be seen in the Fig. 1 wherein during regular topology changes (shown in black dotted line) sudden spikes in VAR anomaly score are observed.

In comparison, our proposed method is able to detect all anomalies without False positives (FP). This indicates that the proposed algorithm is more likely to detect anomalies due to complex attack scenarios while being able to reduce the occurrences of false positives.

V. FURTHER CONSIDERATIONS

Some special conditions of interest are related to the application of our proposed method. They are discussed below.

A. Realistic Grid Patterns

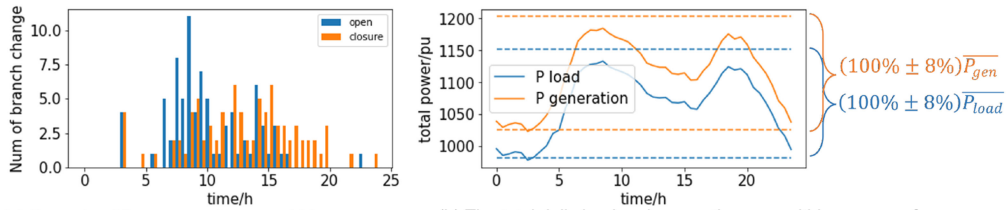
Here we explore the realistic grid pattern, using a **utility-provided real-world dataset** (for reasons of confidentiality, we cannot make this dataset public).

Dataset description- 24 hours worth of ACSE output data from a real utility in the Eastern Interconnect of the U.S. The dataset contains all operational data of the grid based on a 0.5-hour interval.

Here we document the statistical findings with the following notable observations:

- Fig. 10(a): Topology changes frequently in reality, thus our proposed method, which considers the impact of topology change, provides realistic values in handling time-series sensor data.
- Fig. 10(b): Real part of daily loads and generations change smoothly within a range of $(100 \pm 8)\%$ times their average values, with the generation following the load demand throughout the day.
- Fig. 11: 90% of the time, an individual real part of load changes less than 10% of its maximum daily value within a 30-min interval; less than 17% within a 2-hour interval; and less than 50% within 24 hours.

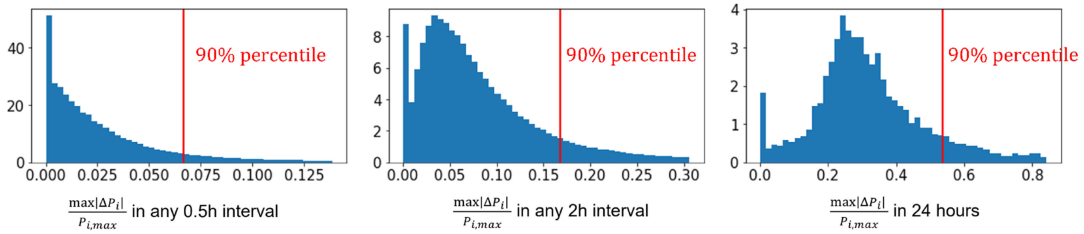
We observed similar behavior for reactive power in the system as well.



(a) Records of line status changes: 164 status changes are observed within 24 hours, on a real grid with >17,000 lines.

(b) The total daily load and generation vary within a range of $(100\% \pm 8\%)\bar{P}$ at most of the time; and the generation follows the load demand throughout the day

Fig. 10. Change of line status, total load and generation on for a real-world load dataset.



(a) During any 0.5h interval, 90% of the loads vary less than 7% of their maximum daily load.

(b) During any 2h interval, 90% of the loads vary less than 17% of their maximum daily load.

(c) 90% of the loads vary less than 50% of their maximum daily load in 24 hours.

Fig. 11. Distribution of individual load variations for real-world load dataset.

B. Non-Synchronization of Measurements

In the worst possible case of measurement non-synchronization, the measurements and status data can be collected at such a different time that the measurements and reference topology are inconsistent with each other. If this happens, our method should detect it as an anomaly, since they are equivalent to topological or measurement errors.

However, the impact of this event is low. In case "uncorrected" unsynchronized measurements or topology exists, then that is an error that would disrupt the grid operator's state estimation. In practice, the grid has some mechanisms to prevent this from happening:

1) Both status data and analog measurements are sampled very frequently: status data are updated upon change of status in PJM and every 4 s in ISO-NE; analog measurements are updated every 10 s in PJM. Considering that the frequency of topology change is low, the high acquisition frequency is likely to guarantee all information is up to date.
2) Topology estimation (TE) can check the consistency between switch status and analog measurements: e.g., if the switch is observed as 'open' but the current measurement on it is non-zero, then this status data is wrong and should be corrected. This processing is within the TE algorithms in the control room.

Therefore, it is likely that we can still obtain a good reference topology not affected by the non-synchronization.

C. Abrupt Changes in Aggregated Load or Generation

Considering that: 1) the load changes slowly following a daily cycle; 2) conventional generators are re-dispatched every 15 min (less frequent than data acquisition we have sufficient time ticks before each re-dispatch) and each re-dispatch is limited by ramp rate; 3) renewable forecasts are improving and the auxiliary services like batteries are making them more stable and "firm" sources of energy, it's still reasonable to assume that for any time t, its recent previous measurements can provide meaningful information.

What if sudden load change happens? At transmission grid level the load is aggregated at high-voltage nodes because of which abrupt change in load is uncommon. However, there exist rare events like load shedding or an unusual re-dispatch where there might be a sudden redistribution of power in the grid in a very short interval. In case of these rare events, the proposed method may detect the abrupt change as an anomaly. In reality, these events may not be anomalous and will be misclassified as anomalies. During such instances, we will rely on the operator to clear these false positives in their decision-making process where additional trustworthy information is available. This is reasonable since these events are rare and the operators are well aware of the situation of active ongoing load-shedding in a region. In case that the abrupt load change is unexpected to the operator, an indicator for anomalous behavior might be a rather useful one.

VI. Conclusion

In this paper, we proposed DynWatch, an online algorithm that accurately detects anomalies using sensor data on a changing graph (grid). DynWatch applies a similarity-based approach to measure how much the graph changes over time, with which we assign greater weight to previous graphs which are similar to the current graph. We use a domain-aware graph similarity

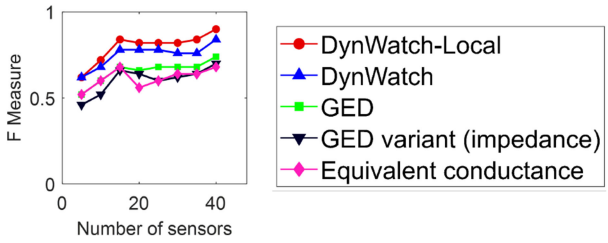


Fig. A1. Result of F-measure on case2383wp, with 40 sensors installed: the proposed LODF-based graph distance outperforms other distance measures.

measure based on Line Outage Distribution Factors (LODF), which exploit physics-based modeling of how changes in one line affect other lines in the graph.

By plugging in different graph similarity measures, our approach could be applied to other domains. Hence, future work could study how sensitive various detectors are for detecting anomalies in graph-based sensor data from different domains.

1) **Problem Formulation and Algorithm:** we propose a novel and practical problem formulation, of anomaly detection using sensors on a changing graph. For this problem, we propose a graph-similarity based approach, and a domain-aware similarity measure based on Line Outage Distribution Factors (LODF).
2) **Effectiveness:** Our DYNWATCH algorithm outperforms existing approaches in accuracy by 20% or more (F-measure) in experiments.
3) **Scalability:** DYNWATCH is fast, taking 1.7 ms on average per time tick per sensor on a 60 k edge graph, and scaling linearly in the size of the graph.

Reproducibility: our code and data are publicly available at https://github.com/bhooi/dynamic.git.

APPENDIX A
COMPARISON OF GRAPH DISTANCE MEASURES

To quantitatively justify the effectiveness of our proposed graph distance, we compared the proposed distance with other traditional measures applicable to power grids:

- Simple GED [54]: the distance between two graphs is equal to the number of changed edges.
- Variant of GED with line admittance used as weights assigned to the changed edges. Admittance is used here because the larger the admittance, the more likely the edge has large power flows on it, meaning it is important to the grid.
- Equivalent conductance-based measure: the distance between two graphs is equal to the sum of the equivalent conductance of all changed edges. Equivalent conductance is able to take more consideration of the system-wise impact of each edge.

Result in Fig. A1 shows our proposed measure outperforms the baselines above. Here the time series data is generated using the pattern from the utility-provided data set, following the process described in Fig. 7.

APPENDIX B
PROOFS OF STATISTICAL ERROR ANALYSIS

Here we will prove the theorems demonstrated in Section III-G.

Theorem III.1 (Error bound): Based on Assumption III.1, III.2 and Definition III.3, the statistical error $\mathbb{E}_{x_1,x_2,\ldots,x_T,x_{T+1}}[(\sum_{t=1}^{T} w_t x_t - x_{T+1})^2]$ with $w_t \geq 0, \forall t$ and $\sum_{t=1}^{T} w_t = 1$, satisfies:

$$\sigma^2 \leq \mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] \leq (1+\max_t w_t)\sigma^2 + c\max_t d_t^*$$

Proof:

$$\mathbb{E}_{x_1,x_2,\ldots x_T}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] \tag{1}$$

$$= \mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \mu_{T+1} + \mu_{T+1} - x_{T+1}\right)^2\right] \tag{2}$$

$$= \mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \mu_{T+1}\right)^2\right] + \mathbb{E}[(\mu_{T+1} - x_{T+1})^2] \tag{3}$$

$$+ 2\mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \mu_{T+1}\right)(\mu_{T+1} - x_{T+1})\right] \tag{4}$$

Based on Assumption III.1, we have $\mathbb{E}[\mu_{T+1} - x_{T+1}] = 0$, and thus

$$\mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \mu_{T+1}\right)(\mu_{T+1} - x_{T+1})\right]$$

$$= \mathbb{E}\left[\sum_{t=1}^{T} w_t x_t(\mu_{T+1} - x_{T+1})\right] - \mu_{T+1}^2 + \mu_{T+1}\mathbb{E}[x_{T+1}]$$

$$= \mathbb{E}\left[\sum_{t=1}^{T} w_t x_t\right]\mathbb{E}[\mu_{T+1} - x_{T+1}] - \mu_{T+1}^2 + \mu_{T+1}^2$$

$$= 0$$

Therefore we have

$$\mathbb{E}_{x_1,x_2,\ldots x_T,x_{T+1}}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] \tag{5}$$

$$= \mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \mu_{T+1}\right)^2\right] + \mathbb{E}[(\mu_{T+1} - x_{T+1})^2] \tag{6}$$

$$= \mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \sum_{t=1}^{T} w_t\mu_t + \sum_{t=1}^{T} w_t\mu_t - \mu_{T+1}\right)^2\right] \tag{7}$$

$$+ \mathbb{E}[(\mu_{T+1} - x_{T+1})^2] \tag{8}$$

$$= \mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \sum_{t=1}^{T} w_t \mu_t\right)^2\right] + \mathbb{E}\left[\left(\sum_{t=1}^{T} w_t \mu_t - \mu_{T+1}\right)^2\right] \tag{9}$$

$$+ 2\mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \sum_{t=1}^{T} w_t \mu_t\right)\left(\sum_{t=1}^{T} w_t \mu_t - \mu_{T+1}\right)\right] \tag{10}$$

$$+ \mathbb{E}[(\mu_{T+1} - x_{T+1})^2] \tag{11}$$

Similarly based on Assumption III.1, it is easy to show that

$$\mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \sum_{t=1}^{T} w_t \mu_t\right)\left(\sum_{t=1}^{T} w_t \mu_t - \mu_{T+1}\right)\right] = 0$$

Thus we have

$$\mathbb{E}_{x_1, x_2, \ldots x_T, x_{T+1}}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] \tag{12}$$

$$= \underbrace{\mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x - \sum_{t=1}^{T} w_t \mu_t\right)^2\right]}_{\text{Variance}} + \underbrace{E\left[\left(\sum_{t=1}^{T} w_t \mu_t - \mu_{T+1}\right)^2\right]}_{\text{Bias}} \tag{13}$$

$$+ \underbrace{\mathbb{E}\left[(\mu_{T+1} - x_{T+1})^2\right]}_{\text{irreducible error}} \tag{14}$$

Based on Assumption III.1 and $w_t \geq 0$ for $\forall t, \sum_{t=1}^{T} w_t = 1$, the variance term can be upper bounded by:

$$\mathbb{E}\left[\left(\sum_{t=1}^{T} w_t x_t - \sum_{t=1}^{T} w_t \mu_t\right)^2\right] = \sum_{t=1}^{T} w_t^2 \mathbb{E}[(x_t - \mu_t)^2] \leq (\max_t w_t)\sigma^2$$

Further making use of Assumption III.2, it is easy to get an upper bound for the bias^2 term:

$$E\left[\left(\sum_{t=1}^{T} w_t \mu_t - \mu_{T+1}\right)^2\right] = \mathbb{E}[(\sum_{t=1}^{T} w_t |\mu_t - \mu_{T+1}|)^2] = \sum_{t=1}^{T} w_t c d_t^* \leq c \max_t d_t^*$$

Finally, as $\mathbb{E}[(\mu_{T+1} - x_{T+1})^2] = \sigma^2$ based on the assumption that $x_{T+1} \sim N(\mu_{T+1}, \sigma^2)$, we are able to **upper bound** the statistical error as:

$$\mathbb{E}_{x_1, x_2, \ldots x_T, x_{T+1}}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] \tag{15}$$

$$\leq (\max_t w_t)\sigma^2 + \max_t c d_t^* + \sigma^2 \tag{16}$$

$$= (1 + \max_t w_t)\sigma^2 + c \max_t d_t^* \tag{17}$$

Meanwhile the **lower bound** is also obvious:

$$\mathbb{E}_{x_1, x_2, \ldots x_T, x_{T+1}}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] \geq \sigma^2 \tag{18}$$

□

Theorem III.2 (Unbiased estimation under infinite data): In the limit of infinite data, the statistical error limits at the lower bound:

$$\mathbb{E}_{x_1, x_2, \ldots, x_T, x_{T+1}}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] = \sigma^2$$

and an unbiased estimate of the true distribution $x_{T+1} \sim N(\mu_{T+1}, \sigma^2)$ is obtainable using the previous samples, i.e.,

$$\mathbb{E}\left[\sum_{t=1}^{T} w_t x_t\right] = \mu_{T+1}$$

$$\mathbb{E}\left[\frac{1}{T-1}\sum_t (x_t - \frac{\sum_t x_t}{T})^2\right] = \sigma^2$$

Proof: In the limit of infinite data, there exist infinite data with the same topology as x_{T+1}, thus it is possible to find the time series data such that $x_1, x_2, \ldots, x_T$ are drawn independently from the same distribution, s.t, $x_t \sim N(\mu_{T+1}, \sigma^2), \forall t$ and $T \longrightarrow \infty$.

Thus we have $w_t = \frac{1}{T} \longrightarrow 0, d_t^* = 0$ for $\forall t \in 0, 1, \ldots, T$ and it is easy to show from Theorem 1 that:

$$\sigma^2 \leq \mathbb{E}_{x_1, x_2, \ldots, x_T, x_{T+1}}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] \leq \sigma^2$$

Thus,

$$\mathbb{E}_{x_1, x_2, \ldots, x_T, x_{T+1}}\left[\left(\sum_{t=1}^{T} w_t x_t - x_{T+1}\right)^2\right] = \sigma^2$$

Also we have,

$$\mathbb{E}\left[\sum_{t=1}^{T} w_t x_t\right] = \sum_{t=1}^{T} w_t \mathbb{E}[x_t] = \mu_{T+1}$$

And based on Bessel's correction, it is easy to get

$$\mathbb{E}\left[\frac{1}{T-1}\sum_t \left(x_t - \frac{\sum_t x_t}{T}\right)^2\right] = \sigma^2$$

REFERENCES

[1] S. M. Amin, "U.S. grid gets less reliable [The data]," *IEEE Spectr.*, vol. 48, no. 1, pp. 80–80, Jan. 2011.

[2] N. Perlroth, "Hackers are Targeting Nuclear Facilities, Homeland Security Department and FBI Say," New York Times, 2017, vol. 6.

[3] D. U. Case, "Analysis of the cyber attack on the ukrainian power grid," *Electricity Inf. Sharing Anal. Center*, vol. 388, 2016.

[4] R. M. Lee, M. Assante, and T. Conway, *Crashoverride: Analysis of the Threat to Electric Grid Operations*. Dragos Inc., Mar. 2017.

[5] F. Maghsoodlou, R. Masiello, and T. Ray, "Energy management systems," *IEEE Power Energy Mag.*, vol. 2, no. 5, pp. 49–57, Sep./Oct. 2004.

[6] PJMManual, "1-Control center and data exchange requirements," 2011. [Online]. Available: pjm.com/ /media/documents/manuals/m01.ashx

[7] "ISO new England operating procedure 14 Appendix F," 2020. [Online]. Available: https://www.iso-ne.com/static-assets/documents/rules_proceds/operating/isone/op14/op14f_rto_final.pdf

[8] F. C. Schweppe and J. Wildes, "Power system static-state estimation, Part I: Exact model," *IEEE Trans. Power App. Syst.*, vol. PAS-89, no. 1, pp. 120–125, Jan. 1970.

[9] E. Handschin, F. C. Schweppe, J. Kohlas, and A. Fiechter, "Bad data analysis for power system state estimation," *IEEE Trans. Power App. Syst.*, vol. 94, no. 2, pp. 329–337, Mar. 1975.

[10] "PJM manual 12: Balancing operations," 2020. [Online]. Available: https://www.pjm.com/∼/media/documents/manuals/m12.ashx

[11] M. Göl and A. Abur, "LAV based robust state estimation for systems measured by PMUs," *IEEE Trans. Smart Grid*, vol. 5, no. 4, pp. 1808–1814, Jul. 2014.

[12] L. Mili, V. Phaniraj, and P. J. Rousseeuw, "Least median of squares estimation in power systems," *IEEE Trans. Power Syst.*, vol. 6, no. 2, pp. 511–523, May 1991.

[13] R. C. Pires, A. S. Costa, and L. Mili, "Iteratively reweighted least-squares state estimation through Givens Rotations," *IEEE Trans. Power Syst.*, vol. 14, no. 4, pp. 1499–1507, Nov. 1999.

[14] S. Li, A. Pandey, and L. Pileggi, "A WLAV-based robust hybrid state estimation using circuit-theoretic approach," 2020, *arXiv:2011.06021*.

[15] Y. Weng, M. D. Ilić, Q. Li, and R. Negi, "Convexification of bad data and topology error detection and identification problems in AC electric power systems," *IET Gener., Transmiss. Distrib.*, vol. 9, no. 16, pp. 2760–2767, 2015.

[16] A. Jovicic, M. Jereminov, L. Pileggi, and G. Hug, "An equivalent circuit formulation for power system state estimation including PMUs," in *Proc. North Amer. Power Symp.*, 2018, pp. 1–6.

[17] S. Li, A. Pandey, S. Kar, and L. Pileggi, "A circuit-theoretic approach to state estimation," in *Proc. IEEE PES Innov. Smart Grid Technol. Europe*, 2020, pp. 1126–1130.

[18] A. Monticelli, *State Estimation in Electric Power Systems: A Generalized Approach*. Berlin, Germany: Springer, 2012.

[19] M. Prais and A. Bose, "A topology processor that tracks network modifications," *IEEE Trans. Power Syst.*, vol. 3, no. 3, pp. 992–998, Aug. 1988.

[20] O. Alsac, N. Vempati, B. Stott, and A. Monticelli, "Generalized state estimation," *IEEE Trans. Power Syst.*, vol. 13, no. 3, pp. 1069–1075, Aug. 1998.

[21] F. F. Wu and W.-H. Liu, "Detection of topology errors by state estimation (Power systems)," *IEEE Trans. Power Syst.*, vol. 4, no. 1, pp. 176–183, Feb. 1989.

[22] E. M. Lourenço, E. P. Coelho, and B. C. Pal, "Topology error and bad data processing in generalized state estimation," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 3190–3200, Nov. 2015.

[23] B. Donmez, G. Scioletti, and A. Abur, "Robust state estimation using node-breaker substation models and phasor measurements," in *Proc. IEEE Milan PowerTech*, 2019, pp. 1–6.

[24] J. D. Hamilton, *Time Series Analysis*, vol. 2. Princeton, NJ, USA: Princeton Univ. Press, 1994.

[25] Z. Wang, J. Yang, Z. ShiZe, and C. Li, "Robust regression for anomaly detection," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–6.

[26] S. Yi, J. Ju, M.-K. Yoon, and J. Choi, "Grouped convolutional neural networks for multivariate time series," 2017, *arXiv:1703.09938*.

[27] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," *ACM Sigmod Record*, ACM, vol. 29, no. 2, pp. 93–104, 2000.

[28] M. Amer and S. Abdennadher, "Comparison of unsupervised anomaly detection techniques," Bachelor's Thesis, Media Engineering and Technology German University in Cairo and Multimedia Analysis and Data Mining Competence Center German Research Center for Artificial Intelligence (DFKI GmbH) Kaiserslautern, Germany, 2011.

[29] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. Int. Conf. Data Mining*, 2008, pp. 413–422.

[30] B. Hooi *et al.*, "GridWatch: Sensor placement and anomaly detection in the electrical grid," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, Springer, 2018, pp. 71–86.

[31] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Trans. Smart Grid*, vol. 8, no. 4, pp. 1630–1638, Jul. 2017.

[32] E. Keogh, J. Lin, S.-H. Lee, and H. Van Herle, "Finding the most unusual time series subsequence: Algorithms and applications," *Knowl. Inf. Syst.*, vol. 11, no. 1, pp. 1–27, 2007.

[33] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient algorithms for mining outliers from large data sets," *ACM Sigmod Record*, ACM, vol. 29, no. 2, pp. 427–438, 2000.

[34] M. Jones, D. Nikovski, M. Imamura, and T. Hirata, "Anomaly detection in real-valued multidimensional time series," in *Proc. Int. Conf. Bigdata/Socialcom/Cybersecurity*. Stanford University, ASE, Citeseer, 2014.

[35] L. Akoglu, M. McGlohon, and C. Faloutsos, "Oddball: Spotting anomalies in weighted graphs," in *Proc. Pacific-Asia Conf. Knowl. Discov. Data Mining*, Springer, 2010, pp. 410–421.

[36] Z. Chen, W. Hendrix, and N. F. Samatova, "Community-based anomaly detection in evolutionary networks," *J. Intell. Inf. Syst.*, vol. 39, no. 1, pp. 59–85, 2012.

[37] M. Mongiovi, P. Bogdanov, R. Ranca, E. E. Papalexakis, C. Faloutsos, and A. K. Singh, "NetSpot: Spotting significant anomalous regions on dynamic networks," in *Proc. Int. Conf. Data Mining*, SIAM, 2013, pp. 28–36.

[38] M. Araujo *et al.*, "Com2: fast automatic discovery of temporal ('Comet') communities," in *Proc. Pacific-Asia Conf. Knowl. Discov. Data Mining*, Springer, 2014, pp. 271–283.

[39] L. Akoglu and C. Faloutsos, "Event detection in time series of mobile communication graphs," in *Proc. Army Sci. Conf.*, 2010, pp. 77–79.

[40] C. C. Aggarwal, Y. Zhao, and S. Y. Philip, "Outlier detection in graph streams," in *Proc. IEEE 27th Int. Conf. Data Eng.*, 2011, pp. 399–409.

[41] S. Ranshous, S. Harenberg, K. Sharma, and N. F. Samatova, "A scalable approach for outlier detection in edge streams using sketch-based approximations," in *Proc. SIAM Int. Conf. Data Mining*, SIAM, 2016, pp. 189–197.

[42] J. Valenzuela, J. Wang, and N. Bissinger, "Real-time intrusion detection in power system operations," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 1052–1062, May 2013.

[43] D. Falcao and S. De Assis, "Linear programming state estimation: Error analysis and gross error identification," *IEEE Trans. Power Syst.*, vol. 3, no. 3, pp. 809–815, Aug. 1988.

[44] N. G. Bretas and A. S. Bretas, "A two steps procedure in state estimation gross error detection, identification, and correction," *Int. J. Elect. Power Energy Syst.*, vol. 73, pp. 484–490, 2015.

[45] R. Mitchell and R. Chen, "Behavior-rule based intrusion detection systems for safety critical smart grid applications," *IEEE Trans. Smart Grid*, vol. 4, no. 3, pp. 1254–1263, Sep. 2013.

[46] S. Papadimitriou, J. Sun, and C. Faloutsos, "Streaming pattern discovery in multiple time-series," in *Proc. 31st Int. Conf. Very Large Data Bases*, 2005, pp. 697–708.

[47] P. Galeano, D. Peña, and R. S. Tsay, "Outlier detection in multivariate time series by projection pursuit," *J. Amer. Stat. Assoc.*, vol. 101, no. 474, pp. 654–669, 2006.

[48] R. Baragona and F. Battaglia, "Outliers detection in multivariate time series by independent component analysis," *Neural Comput.*, vol. 19, no. 7, pp. 1962–1984, 2007.

[49] H. Lu, Y. Liu, Z. Fei, and C. Guan, "An outlier detection algorithm based on cross-correlation analysis for time series dataset," *IEEE Access*, vol. 6, pp. 53593–53 610, 2018.

[50] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A review on outlier/anomaly detection in time series data," *ACM Comput. Surveys*, pp. 1–13, 2020.

[51] D. J. Brueni and L. S. Heath, "The PMU placement problem," *SIAM J. Discrete Math.*, vol. 19, no. 3, pp. 744–761, 2005.

[52] L. A. Zager and G. C. Verghese, "Graph similarity scoring and matching," *Appl. Math. Lett.*, vol. 21, no. 1, pp. 86–94, 2008.

[53] L. Akoglu, H. Tong, and D. Koutra, "Graph based anomaly detection and description: A survey," *Data Mining Knowl. Discov.*, vol. 29, no. 3, pp. 626–688, 2015.

[54] P. Shoubridge, M. Kraetzl, W. Wallis, and H. Bunke, "Detection of abnormal change in a time series of graphs," *J. Interconnection Netw.*, vol. 3, no. 01n02, pp. 85–101, 2002.

[55] K. M. Kapsabelis, P. J. Dickinson, and K. Dogancay, "Investigation of graph edit distance cost functions for detection of network anomalies," *Anziam J.*, vol. 48, pp. C 436–C449, 2006.

[56] M. E. Gaston, M. Kraetzl, and W. D. Wallis, "Using graph diameter for change detection in dynamic networks," *Australas. J. Combinatorics*, vol. 35, pp. 299–311, 2006.

[57] J. Jost and M. P. Joy, "Evolving networks with distance preferences," *Phys. Rev. E*, vol. 66, no. 3, 2002, Art. no. 0 36126.

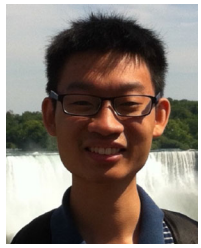
[58] L. Zager, "Graph similarity and matching," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, USA, 2005.

[59] S. V. N. Vishwanathan, N. N. Schraudolph, R. Kondor, and K. M. Borgwardt, "Graph kernels," *J. Mach. Learn. Res.*, vol. 11, pp. 1201–1242, 2010.

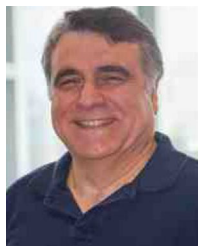
[60] H. Bunke, P. J. Dickinson, M. Kraetzl, and W. D. Wallis, *A Graph-Theoretic Approach to Enterprise Network Dynamics*, vol. 24. Berlin, Germany: Springer, 2007.
[61] A. J. Wood, B. F. Wollenberg, and G. B. Sheblé, *Power Generation, Operation, and Control*. Hoboken, NJ, USA: Wiley, 2013.
[62] F. R. Hampel, "Robust statistics: A brief introduction and overview," in *Research Report/Seminar für Statistik, Eidgenössische Technische Hochschule (ETH)*, vol. 94. *Seminar für Statistik, Eidgenössische Technische Hochschule*, 2001.
[63] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011.
[64] E. Castillo, A. J. Conejo, R. E. Pruneda, and C. Solares, "Observability analysis in state estimation: A unified numerical approach," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 877–886, May 2006.
[65] M. C. de Almeida, E. N. Asada, and A. V. Garcia, "Power system observability analysis based on gram matrix and minimum norm solution," *IEEE Trans. Power Syst.*, vol. 23, no. 4, pp. 1611–1618, Nov. 2008.
[66] S. Chakrabarti and E. Kyriakides, "Optimal placement of phasor measurement units for power system observability," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 1433–1440, Aug. 2008.
[67] H. A. Song, B. Hooi, M. Jereminov, A. Pandey, L. Pileggi, and C. Faloutsos, "PowerCast: Mining and forecasting power grid sequences," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, Springer, 2017, pp. 606–621.
[68] N. Shah, D. Koutra, T. Zou, B. Gallagher, and C. Faloutsos, "Timecrunch: Interpretable dynamic graph summarization," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, ACM, 2015, pp. 1055–1064.
[69] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, 1962.
[70] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

Shimiao Li (Graduate Student Member, IEEE) received the B.E. degree in electrical engineering from Tianjin University, Tianjin, China, in 2018. She is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering (ECE), Carnegie Mellon University, Pittsburgh, PA, USA, where she is advised by Lawrence Pileggi. Her research interests include analytical methods for grid operation and optimization, as well as physics-informed machine learning for real-world grid cybersecurity, reliability and optimization appliations.

Amritanshu Pandey (Member, IEEE) was born in Jabalpur, India. He received the M.Sc. and Ph.D. degrees in electrical engineering in 2012, and 2019, respectively, from Carnegie Mellon University, Pittsburgh, PA, USA. He is currently a Special Faculty with the Electrical and Computer Engineering Department, Carnegie Mellon University. He was with the Pearl Street Technologies, Inc. as a Senior Research Scientist and as an Engineer with MPR Associates, Inc. He develops first principles-based physics driven and ML-based data driven methods and models for energy systems that are expressive (i.e., capture sufficient physical behavior accurately) and efficient in terms of speed and scalability.



Bryan Hooi received the B.S. degree in mathematics and the M.S. degree in computer science from Stanford University, CA, USA. He received the Ph.D. degree in machine learning from Carnegie Mellon University, Pittsburgh, PA, USA, where he was advised by Christos Faloutsos. He is currently an Assistant Professor with the School of Computing and the Institute of Data Science, National University of Singapore, Singapore. His research interests include scalable machine learning, deep learning, graph algorithms, anomaly detection, and biomedical applications of AI.



Christos Faloutsos (Member, IEEE) is a Professor with Carnegie Mellon University, Pittsburgh, PA, USA, and an Amazon Scholar. He is an ACM Fellow, he was a Member of the Executive Committee of SIGKDD, he has authored or coauthored more than 400 refereed articles, 17 book chapters and three monographs. He holds 8 patents (and several more are pending), and he has given more than 50 tutorials and more than 25 invited distinguished lectures. His research interests include large-scale data mining with emphasis on graphs and time sequences, anomaly detection, tensors, and fractals. He was the recipient of the Fredkin Professorship in Artificial Intelligence (2020), the Presidential Young Investigator Award by the National Science Foundation (1989), the Research Contributions Award in ICDM 2006, the SIGKDD Innovations Award (2010), the PAKDD Distinguished Contributions Award (2018), 29 "best paper" awards (including 8 "test of time" awards), and four teaching awards. Eight of his advisees or co-advisees have attracted KDD or SCS dissertation awards.

Lawrence Pileggi (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 1989. He is the Tanoto Professor and the Head of Electrical and Computer Engineering with Carnegie Mellon University, and has previously held positions with Westinghouse Research and Development and the University of Texas at Austin, Austin, TX, USA. His research interests include various aspects of digital and analog integrated circuit design, and simulation, optimization and modeling of electric power systems. He was a Co-Founder of Fabbrix Inc., Extreme DA, and Pearl Street Technologies. He was the recipient of various awards, including Westinghouse Corporation's Highest Engineering Achievement Award, the Semiconductor Research Corporation (SRC) technical excellence awards in 1991 and 1999, the FCRP inaugural Richard A. Newton GSRC Industrial Impact Award, the SRC Aristotle Award in 2008, the 2010 IEEE Circuits and Systems Society Mac Van Valkenburg Award, the ACM/IEEE A. Richard Newton Technical Impact Award in Electronic Design Automation in 2011, the Carnegie Institute of Technology B.R. Teare Teaching Award for 2013, and the 2015 Semiconductor Industry Association (SIA) University Researcher Award.