Noise Resilient Learning for Attack Detection in Smart Grid PMU Infrastructure

Prithwiraj Roy*§, Shameek Bhattacharjee†§, Sahar Abedzadeh† and Sajal K. Das*
*Missouri University of Science and Technology, Rolla, MO, USA,
†Western Michigan University, Kalamazoo, MI, USA
E-mail: przhr@mst.edu; shameek.bhattacharjee@wmich.edu; sahar.abedzadeh@wmich.edu; sdas@mst.edu

***Abstract*—Falsified data from compromised Phasor Measurement Units (PMUs) in a smart grid induce Energy Management Systems (EMS) to have an inaccurate estimation of the state of the grid, disrupting various operations of the power grid. Moreover, the PMUs deployed at the distribution layer of a smart grid show dynamic fluctuations in their data streams, which make it extremely challenging to design effective learning frameworks for anomaly based attack detection. In this paper, we propose a noise resilient learning framework for anomaly based attack detection specifically for distribution layer PMU infrastructure, that show real time indicators of data falsifications attacks while offsetting the effect of false alarms caused by the noise. Specifically, we propose a feature extraction framework that uses some Pythagorean Means of the active power from a cluster of PMUs, reducing multi-dimensional nature of the PMU data streams via quick big data summarization. We also propose a robust and noise resilient methodology for learning thresholds based on generalized robust estimation theory of our invariant feature. We experimentally validate our approach and demonstrate improved reliability performance using two completely different datasets collected from real distribution level PMU infrastructures.**

***Index Terms*—Anomaly Detection, Big Data Management, Learning based PMU Security, Smart Grid Security.**

I. Introduction

Modern smart electrical grid is an example of a large scale Internet of Things (IoT) domain, in which advanced IoT telemetry devices like Phasor Measurement Units (PMUs) are deployed for collecting high resolution time series measurements. Such measurements offer highly improved real time situational awareness of the state of the grid compared to the older Supervisory Control And Data Acquisition (SCADA) infrastructure measurements. Another critical difference between the PMU and SCADA technologies is how they gauge the angle measurements. While in PMU, the angle measurements are made directly, the SCADA system measure the voltage angles by using voltage measurements, active and reactive power, network parameters, and a reference angle. Thus the quality of the results depend heavily on network parameters for SCADA system which are not always precise [2]. Thus the PMUs have become the critical cornerstones in the smart grid architecture. Additionally, several internet based communication technologies for sending such data and receiving the corresponding actions are being developed for a vision of an adaptive, reliable, and efficient modern electrical grid.

§Equal contribution

The PMUs are IoT endpoint devices positioned at critical locations at both the transmission and distribution layers of a smart grid infrastructure. The PMUs record time-synchronized measurements (voltage, current, phase angles - collectively called synchrophasor data) and send them to an aggregator called Phasor Data Concentrator (PDC), which in turn relay such data to a Local Control Center (LCC). Multiple LCCs allow smart grid operators to localize and infer the type, time, and location of a fault or disturbance as well as support critical operations such as state estimation.

Data Integrity Threat in PMU: Data integrity in PMUs is extremely critical since the control centers base their decisions (control or actuation) directly on these measurements; the output of various applications also uses these measurements indirectly, such as economic dispatch and contingency analysis. Now with the process of digitalization, these systems have become increasingly more complex and have been exposed to potential security threats which could lead to a fatal effect on the power grid network. These threats are posed by numerous parties such as hackers, ex-employees, competitors, and even maintenance personnel [3]. Various reports on cyber-attacks in power distribution systems are threatening the security and reliability of these operations. For example, a report of the US National Research Council highlights potential multi-state blackouts as a result of coordinated False Data Injection (FDI) attacks on the power systems [6]. An attack on the Ukrainian power grid resulted in the loss of power for 225,000 customers in three different territories which lasted for several hours [7]. The Stuxnet worm that attacks particular programmable logic controller (PLC) on the vulnerable windows computers affected more than 100,000 industrial components [8].

The widely used IEEE C37.118-2 protocol for synchrophasor data communication between the PMUs and PDC is reported to be vulnerable to cyber-attacks [11], [13]. In fact, most synchrophasor data transmission happens on insecure IP networks. Strong encryption is not feasible due to the latency criticality of PMU applications and high data resolution that typically generate 50-120 samples of 12 physical quantities per second. All of the above reasons increase the chances of data falsification attacks once the adversaries laterally intrude into the PMUs. Furthermore, the possibility of transduction attacks causing data falsification on such telemetry data makes cryptography and network traffic based attack detection ineffective [25]. Therefore, more data-driven anomaly based data falsification attack detection is a viable cybersecurity approach

that depends on the unraveling of suspicious patterns.

A. Motivation and Challenges

The primary motivation of this paper is fueled by the need for an anomaly based FDI attack detection scheme that is resistant to outliers caused by either measurement and/or process noise that may bias the accurate learning of the underlying structure of distribution level PMU data under benign operating conditions. Additionally, the resultant framework needs to be light-weight and automated, and it should be able to detect attacks quickly that would work under settings of distribution level power grid systems.

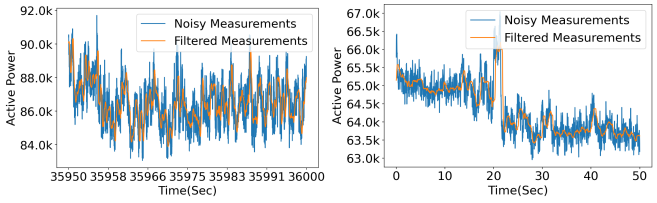
Challenges of Distribution Level PMUs: The specific challenge of distribution level PMUs is that the data under benign conditions change readily under legitimate benign conditions, as opposed to transmission level PMUs, where the dynamics of PMU data streams show inherently fewer variations. The distribution level PMU data streams are particularly prone to unpredictable variations since they are directly connected to the customer layer. The customer layer contains an eclectic mix of loads from campuses, residences, and businesses that create higher variations in current data streams, which are affected by not only physics but also human behavior. Additionally, distributed energy sources (e.g., solar panels) increase uncertainty in the measurement patterns collected from PMUs. These issues make the approaches used in transmission level PMUs for anomaly detection unreliable in the distribution level PMUs setting.

Challenges of Outliers and Noise: After careful synthesis of the existing literature we found that most works in FDI attacks on PMU, have small duration of study that ranged from a few minutes to a few days [15], [21], [22], [24]). In the real world, however, this attack detection needs to run over much longer time horizons and be effective.

Keeping that in mind, we tried to apply our previous approach from our preliminary conference version [1] on the EPFL dataset that has duration in the order of months instead of a few minutes to few days. In our preliminary work, we had found success with both false alarms and attack detection on an LBNL dataset that was 11 days long. However, we found that it failed to achieve any acceptable performance on the much longer duration PMU dataset from the Ecole Polytechnique Federale De Lausanne (EPFL), Switzerland.

We believe that the root causes of such failure can be attributed to outliers and process noise. This motivated us to extend our previous method with robust learning. Specifically, [26] conducted studies on the EPFL dataset and found and quantified noise due to various environmental drivers and events that create changes in the measured data. The paper also proposed a method to quantify noise in the PMU data. Following that method, we tried to quantify noise for both datasets: LBNL and EPFL. We found that Signal-to-Noise (SNR) is higher for EPFL data compared to LBNL data. For example, the SNR for EPFL data is 7.6 while the same for LBNL data is 5.4. Moreover, we have also used a Median Absolute Filter [26] to demonstrate the presence of significant noise in both datasets as shown in Fig. 1. It is evident that the filtered measurements (orange line) closely follow the actual signal for the LBNL data. However, for the EPFL data, the filtered signal is quite far from the actual signal envelope confirming that the later dataset contains more variations that deviate from a central tendency, and it also confirms that filtration causes the filtered data to be far away from the actual data and this caused an increase in false alarms. We tried to understand the underlying cause of this and found the following:

There are many routine benign events like series capacitor switching, load transferring, etc. [44] in the power grids that cause sudden short lived perturbations in the PMU measurements. To properly model the benign system behavior, we ought to take into account the measurement changes caused by routine events. If we use a noise filter such as the popularly used *Median Absolute Filter* [26], [42], [43], many such events get eliminated from the filtered signal along with occasional outliers. This produces an overcompensated model of benign behavior that does not reflect the actual routine behavior during benign operation. This increases the occurrences of false alarms when our preliminary method is applied in the testing/generalization phase with filtered data. This makes time series anomaly based attack detection frameworks unusable in practice. Also, the noise filtering technique in [26] requires the *Filter Order*, M, which is manually extracted from the data. This implies the filter order is dependent on the data under consideration and it adds computational overhead and manual intervention to the process. Therefore, adding existing filtration approaches is not viable in our setting.



(a) (b)

Fig. 1: Noise in Active Power: (a) LBNL (b) EPFL.

Then we tried the EPFL dataset with small duration, but since this small duration is not representative of all benign conditions, it produced high rate of false alarms when tested over a duration of months. In contrast, when we increased the size of the training set in the EPFL, the detection rate dropped because the thresholds learning was getting biased by more outliers. To explain the link between longer duration and this development, we believe that if a longer duration of training data is taken into consideration, there is a higher chance events/environmental drivers that create outliers manifest themselves. It is a simple result of unbiased sampling. If one considers a small training and testing dataset, one may get lucky because the small portions being considered does not have any event/drivers that lead to outliers in the training and testing. This favors good performance but is not robust. Since outliers in the benign data prevent accurate learning the general representation of the underlying structure of benign operating conditions, it affects the performance of anomaly detection techniques which depend on characterizing the latent

structure of benign data. The root cause of this paradox seems to be sampling bias although there may be other unknown things like any topology change that might have happened in the test bed which we may be unaware of.

Therefore, we concluded that we need to develop *noise/outlier resilient learning* of the anomaly detection thresholds that can represent the most general bounds in the underlying structure of benign operations in the anomaly detection frameworks for distribution level PMUs; which enables more accurate anomaly based attack detection.

B. Contributions of this Work

In this paper, we first discuss multiple attack strategies for data falsification in smart grid PMU infrastructure. Then we propose a semi-supervised learning framework for anomaly based data falsification attack detection in the distribution level PMU network that contains a set of PMUs under the control of a PDC. The proposed framework is divided into two parts: (1) the feature engineering generates a latent space of features in a low dimension, which models the multivariate streams of raw PMU data into one learnable feature across time; and (2) robust learning of anomaly detection threshold and criterion from the generated feature space that contains outliers.

Specifically, for feature engineering, we propose the use of active power as the process variable and use the ratio of harmonic mean to arithmetic mean of the active power on a strategically calculated spatial and temporal granularity as an invariant. Such spatial and temporal granularities lead to time series invariance under no attack, but show changes under various kinds of data falsification attacks. Next, we convert the invariant into stateful residuals which is a feature that balances false alarm versus detection trade-off.

For robust learning of attack detection criterion from the feature representation, we propose to use robust estimation theory to derive the optimal thresholds that characterize the benign operating condition in the presence of noise and heavy tailed nature of the feature distribution. Finally, we validate our work by using two real PMU datasets. The first dataset was collected by the Power Standards Lab (PSL) at Lawrence Barkley National Lab (LBNL) campus in Berkeley, CA, which developed high-precision μ-PMUs deployed at multiple utilities and LBNL campus locations on a 12 kV distribution grid [16]. The second dataset was collected by the smart grid infrastructure monitoring project on a 20 kV distribution grid in the EPFL campus, Switzerland across several months [20].

These two large datasets corroborate the generalizability of our novel approach. The main benefits of our approach are to provide a practical framework to identify the presence of FDI attacks in the distribution level PMU architecture. The framework (i) is noise resilient, real time, light-weight, and semi-supervised; (ii) enables quick identification; (iii) simultaneously works for a variety of data falsification attack types; and (iv) works for multiple data sources with different distribution and frequency ranges.

Paper Organization: The rest of this paper is organized as follows. Section II introduces the system model and describes the datasets, while Section III discusses the threat model. Section IV and Section V respectively present the proposed framework and experimental results. Section VI reviews the related works and Section VII presents further discussion on the scope and applicability of the proposed appraoch. Finally, Section VIII concludes the paper.

II. System Model

In this section, we discuss the PMU system architecture and different datasets used for modeling and validation.

A. PMU System Architecture

Now we layout the architecture of a PMU network on a 3-phase AC system. For any phase j, the PMU devices which are essentially IoT devices that measure time-stamped (t) voltage and current magnitudes denoted by $V_t(j)$, $I_t(j)$ and the phase angles $\theta_t^V(j)$, $\theta_t^I(j)$ respectively. In our paper, we use the term 'load' as synonymous to the current magnitude per phase.

The PMUs are deployed at strategic locations of the distribution layers of a smart electrical grid which send their data to PDC which in turn send sychrophasor data to LCC. The estimated state from LCC informs power generation, transmission, and distribution strategies and applications such as early fault detection and load balancing. The architecture of a typical PMU-PDC infrastructure is shown in Fig. 2.

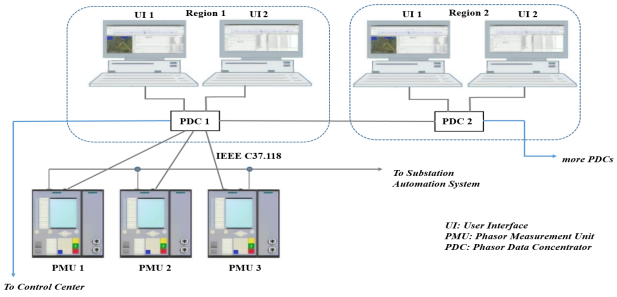


Fig. 2: Architecture of a PMU Infrastructure.

B. Dataset Description

To validate our contribution, we used two datasets viz. EPFL and LBNL with contrasting characteristics, to ensure that our method generalizes across datasets. In each of the 3 phases 4 physical quantities are being measured by a PMU device: voltage magnitude, current magnitude, phase angle of voltage, and phase angle of current. Therefore, in both datasets each PMU is producing 12 streams of data.

EPFL Dataset: This data was collected for a smart grid project by the Ecole Polytechnique (EPFL), Switzerland [20]. It contains data from five PMUs deployed on a 20 kV distribution layer. The PMU devices are named as: PMU2, PMU3, PMU4, PMU5, PMU6 at a data resolution of 50 samples per second. We have taken 6 months of data (Sep-14 to Feb-15) for our model for training and testing, which the longest in the literature of PMU data integrity attack detection research. The work [26], showed that this data-set contains a lot of noise.

LBNL Data: We used another dataset collected by the Power Standards Lab (PSL) located at Lawrence Berkeley National Lab (LBNL), USA, which was used in our preliminary work [1]. We have included the LBNL dataset for performance comparison. This dataset is collected from three high precision μ-PMUs deployed at multiple LBNL campus locations on

12 kV distribution grid [16]. Each μ-PMU device is given a name for unique identification: A6, Grizzly and Bank514. The measurements are taken a rate of 120 Hz with time stamp accuracy of 100 ns, for 11 days (from 1-Oct-15 to 11-Oct-15).

In this paper, we are specifically interested in a decentralized anomaly detection that runs on a PDC or a LCC and facilitates early attack detection from a bunch of PMUs that are geographically proximate in terms of the PMU network.

III. Threat Models

Telemetry attack data on PMU are not available since these are protected systems. Threat models for data integrity attacks in PMU have been most simulated and emulated in the literature in two competing ways. Some works emulate specific strategies but the perturbation amount is random, while others use a strategic perturbation amount to not violate physical bounds of operation (typically applies on voltage and phase data). There is some criticism of this approach in that the attacks are random in nature without quantifiable ways to emulate a threat landscape that would allow security performance evaluation. The other types of PMU data integrity attacks consider optimal attacks or a specific instance of attack hiding [30]. While papers on hiding attacks from bad data detectors exist, the operational impact is not emphasized.

We believe that threat models should ideally parameterize if possible all instances of data falsification attacks and show the performance of a defense framework as a function of those parameters to understand the performance limits. This is because a defender can never know what kind of attack will happen from what adversary. Therefore, assuming a specific attack instance or strategy and then proposing solutions validated against those attacks are realistic. Many adversaries act in a bounded rational and sometimes what defenders would perceive as irrational, which creates the paradox between expectations and reality.

Therefore, to avoid this pitfall we have used a generalized attack emulation technique proposed in several earlier papers [4], [5] that parameterizes the data integrity threat landscape into the following features: (i) Attack types (ii) Attack Strength (iii) Attack Scale (iv) Attack Strategy. All possible instances/values of these features are implemented in the attack simulation to create a super set of attack combinations that do not have optimistic or pessimistic assumptions about adversaries' actions or capabilities. Our method is tested against all the whole super set and performance is reported accordingly. Below we provide details of how we created the data integrity attack landscape for this work.

Attack Types: Attack types indicate how data is changed from each PMU and the impact objective behind data falsification. Organized adversaries may inject false data from one or multiple compromised PMUs concurrently by changing either data at rest, or in transit from the PMUs to the LCC. Any of the four streams (*Voltage Magnitude, Voltage Angle, Current Magnitude, Current Angle*) of any phase may be falsified.

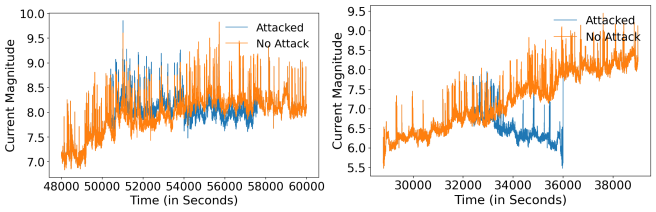
However, in this paper's validation, we show adversary falsifies the 'current magnitude' because the current data stream at the distribution level changes constantly over a wide range even under benign conditions. This is not the case with the voltage and phase angle values. For example, in power grids, the electrical devices and equipment that are connected to the network are often very sensitive to voltage. Hence, sudden increase or drops in voltage damage such expensive devices. To counter that, voltage stabilizers are used in electric grid network to keep the voltage stable. Thus, if the current is compromised, that does not necessarily impact the voltage. Instead power measurements get impacted because of the current modifications. Small margin attacks hide behind these randomnesses. While our method may work for any falsification (as reasoned later with active power), all our results and validation assume current data falsification.

Let $I_t^i(act)$ be the actual current magnitude measured by i-th PMU at time slot t and let I_t^i be its advertised value that reached the PDC. Under no attacks, $I_t^i = I_t^i(act)$, while in the presence of attacks, the reported value I_t^i can be perturbed in the following ways:

- Deductive Attack: The current magnitude from the i-th compromised PMU at time t is reduced from its actual value such that $I_t^i(act) - I_{\delta_t}$, where $I_{\delta_{min}} \leq I_{\delta_t} \leq I_{\delta_{max}}$, where the I_{δ_t} is a perturbation value that is sampled according to a strategic statistical distribution (which is explained under attack strategies), bounded within the interval $[I_{\delta_{min}}, I_{\delta_{max}}]$ with $I_{\delta_{min}} > 0$.
  If the current magnitude drops in a certain phase, it disrupts the efficiency of the grid by reducing the power utilization factor (or efficiency, η) which indicates the ratio of mechanical input power to the generated electrical output power. This creates a motivation to falsify current measurements with a deductive attack type.
- Additive Attack: Similarly, an additive data falsification may be injected by modifying the current magnitude $I_t^i = I_t^i(act) + I_{\delta_t}$ from a compromised PMU, where I_{δ_t} is chosen with a similar strategic principle as mentioned in the deductive attack type. An increase in the current on a given phase will trigger a drop in the phase voltage. If the current magnitude increases beyond a safe limit, the concerned phase is shed, or the load is switched/distributed to other phases. To keep the voltage from dropping more, power needs to be injected in a given phase. The above reasons create a motivation for additive perturbation of current measurements collected from PMUs. The attack type can be launched by a rival utility to make the control center believe in a sudden increase in load which might lead to load shedding in that particular phase.
- Alternating Switching Attack: The adversary alternates between additive and deductive falsification for equal amounts of time, ensuring the same average of perturbation amounts I_{δ_t} over the time domain. In such a case, the total average additive and the total average deductive perturbation amounts over a particular attack time period will balance each other. This statistical balancing makes it hard for many cumulative change point based statistical anomaly detectors to detect such attacks. Fig. 3a demonstrates the impact on the current magnitude if a PMU is compromised with such an attack. The grid is affected in

the same way as additive and deductive types would for the respective durations, but will escape some statistical detectors.

- Mirroring Attack:Here the attacker monitors the I_t^i for a given time period and then he replaces the then actual current measurements with a mirror image of the previously recorded I_t^i over a previous period [9]. The mirroring attack makes sense when the adversary wants to hide the detection of a legitimate event or change point by the PDC/LCC. By mirroring the old perfectly believable values, the PDC/LCC never receives the updated real measurements that would have enabled them to detect a significant change in the state of the network. Fig. 3b shows the impact of mirroring attack on the current magnitude, by hiding a grid state with a sudden increase in current magnitude, by replaying a mirror image of the just previously recorded readings.



(a) (b)

Fig. 3: Attack on Current Magnitude of PMU2: (a) Alternating Attack; (b) Mirroring Attack.

Attack Strength: We denote $I_{\delta_{avg}}$ as the *average margin of false data* per compromised PMU device. The $I_{\delta_{avg}}$ is a strategic parameter that is selected by an adversary depending on how much damage it wants to inflict on the electrical grid. We keep this as an uncontrolled variable to test detection sensitivity since there could be various applications of PMU data. We consider that the attack falsification sample $I_{\delta_t} \in [I_{\delta_{min}}, I_{\delta_{max}}]$ being sampled from a distribution with a certain strategy (see later) whose long term average is $I_{\delta_{avg}}$ The $I_{\delta_{min}}, I_{\delta_{max}}, I_{\delta_{avg}}$ extremes are informed by known limits of each quantity physically possible and bad data detector [30].

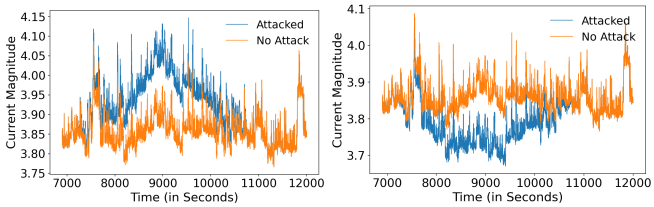
Attack Strategies: We consider three types of attack strategies that are employed by the attacker for an attack period Δ_a:

(a) Step Strategy: In this case the adversary modifies all samples to higher (additive) or lower (deductive) values by I_{δ_c} in the attack period, Δ_a [34].

$$I_t^i = \begin{cases} I_t^i(act), & \text{if } t \notin \Delta_a \\ I_t^i(act) + I_{\delta_c}, & \text{if } t \in \Delta_a. \end{cases}$$

(b) Ramp Strategy: A ramp attack involves gradual falsification of the actual measurements. Here adversary gradually increases the I_{δ_t} in each time slots to reach $I_{\delta_{max}}$ and then again gradually decreases I_{δ_t} [12]. Here λ_r is the gradient of the introduced ramp attack. Based on the adversary's intent, this attack can also be both additive and deductive in nature. Fig. 4a and 4b shows the impact on current magnitude for additive and deductive ramp attacks respectively.

$$I_t^i = \begin{cases} I_t^i(act), & \text{if } t \notin \Delta_a \\ I_t^i(act) + \lambda_r.t, & \text{if } t \in \Delta_a/2 \\ I_t^i(act) - \lambda_r.(\Delta_a - t), & \text{if } \Delta_a/2 \in t \in \Delta_a. \end{cases}$$



(a) (b)

Fig. 4: Attack on Current Magnitude of PMU2 (a) Additive Ramp; (b) Deductive Ramp.

(c) Random Strategy: This attack involves the addition/subtraction of positive values generated by a uniform random function to the actual measurements [5]. The lower ($I_{\delta_{min}}$) and upper ($I_{\delta_{max}}$) bounds for selection are provided to the function as an input. While this is simple, that does not change the resultant shape of the load distribution drastically, making it a less obvious attack.

$$I_t^i = \begin{cases} I_t^i(act), & \text{if } t \notin \Delta_a \\ I_t^i(act) \pm rand(I_{\delta_{min}}, I_{\delta_{max}}), & \text{if } t \in \Delta_a. \end{cases}$$

Attack Scale: This includes how many PMUs in a decentralized system are involved in data falsification at the same time which can affect the accuracy of outcomes.

IV. Proposed Framework

We divide the proposed framework into the following stages: (1) PMU-specific process variable (rather than stream-specific); (2) Invariant based anomaly detection metric building by optimizing spatial and temporal granularities of the process variables; (3) Building stateless and stateful residuals; (4) Learning thresholds of the stateful residual in a noise resilient manner to establish the stateful residual's metric under benign conditions; 5) Determine the detection criteria parameters, based on learning from the training and cross validation steps, and apply it on the testing set. An overall flow diagram of the proposed framework is shown in the following Fig. 5.

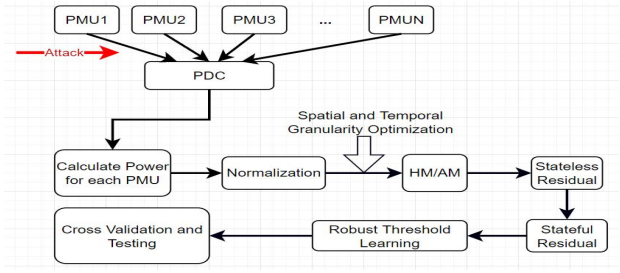


Fig. 5: Flow Diagram of the Proposed Framework.

A. Active Power as Process Variable

As indicated earlier, there are 12 streams of data per PMU, and each stream is generated at rates as high as 50 to 120 sample data points per second. This shows that the variety of data and velocity of the data are extremely large. If a framework requires monitoring each of these multiple data streams per PMU individually, the anomaly monitoring of all these streams separately increases the computational cost and

latency in anomaly detection analytics. Given the high velocity of the data, quick and lightweight big data summarization is required.

To address the above challenge, we propose the phase wise active power calculated from synchrophasor data streams per PMU, as the process variable over which the data driven invariant is calculated. The active power ($p(j)$) per phase from PMU measurements is calculated using the following standard power equations:

$$p(j) = V(j)I(j)\cos\theta(j) \tag{1}$$

where $j \in \{1, 2, 3\}$ denote the phases and $V(j), I(j), \theta(j)$ are voltage magnitude, current magnitude, and $\theta(j) = \theta_t^V(j) - \theta_t^I(j)$ is the angle difference between voltage and current phases respectively, for the j-th phase. This reduces the complexity of monitoring each stream separately unlike existing works. Another advantage is that any deliberate falsification of the voltage or current (both in terms of magnitude and phase) will impact the active power, and hence we can potentially detect an attack on any of the data streams from PMUs. Therefore, for our anomaly detection, we propose to use the phase wise monitoring of the active power $p(j)$ as a starting point. Given that our analysis is phase specific we have dropped the phase j from subsequent equations for simplicity.

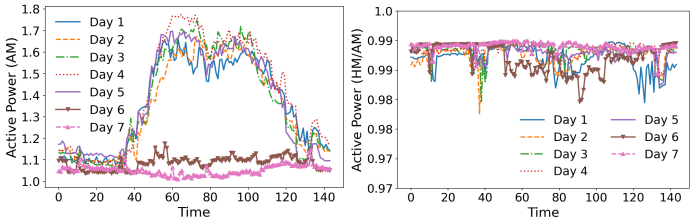
Winsorization and Normalization: We apply winsorization of $\alpha\%$ on the derived active power dataset during the training to remove instantaneous disturbances typically caused due to electromagnetic transients which ensures that we learn the most general benign patterns. Please note, this is done only for the training set.

As mentioned earlier, PMUs are deployed at different locations of the grid network that attach themselves to multiple kinds of customer loads. Due to this reason, the active power from each PMU is not in the same range and varies across PMUs.Variables that are measured on different scales do not contribute equally to the feature set. Therefore, we do normalization via *MinMax* Scaling. We also add a constant value ($= 1$) as per Eqn. 2 to keep all the values $>= 1$ which is required for Harmonic Means to work in the intended manner. For notation simplicity, we used P instead of P_{scaled} to indicate active power for the rest of the paper.

$$P_{scaled} = \frac{p - min(p)}{max(p) - min(p)} + 1. \tag{2}$$

B. Invariant for Anomaly Detection Metric

For real time anomaly detection in CPS, it has been established that a metric which is invariant under normal operating conditions (without any attack) is ideal for attack detection [4], [31]. However, unlike tightly controlled industrial CPS applications, the distribution level synchrophasor data is affected by randomness in renewable power outputs and heterogeneous consumer types. This causes traditional statistical invariants to have high randomness. As shown in Fig. 6a the arithmetic mean of active powers of the time series is not stationary. Prior works such as [17] propose the use of derived smoothing statistics of the arithmetic mean (such as ARMA, EWMA, CUSUM control charts) for time series anomaly detection. However, as shown in the Fig. 6a, the time series of mean active power of PMUs vary greatly over time windows, making it difficult to distinguish legitimate changes from a malicious one. Any moving average or smoothing technique either loses sensitivity for a small margin of attacks (since the moving average does not reflect the changes beyond already existing deviations) or has a high number of false alarms. To avoid this we need a new approach for these smart living CPS.



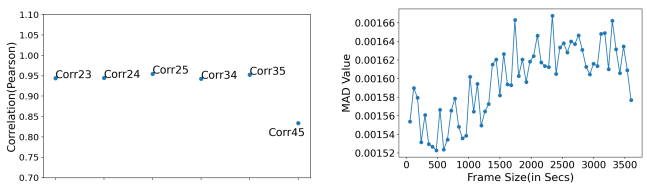
(a) (b)

Fig. 6: Illustrations of AM and HM/AM: Weekdays: Day 1,4,5,6,7; weekends: 2,3 (a) AM; (b) HM/AM.

Recently, in [4], for a different problem relating to smart metering, we had shown that the ratio of harmonic mean and arithmetic mean of a positively correlated set of random variables exhibit invariance in their time series even as the individual means show non-stationarity. Furthermore, [4] proved that any data perturbation in any variable of this correlated set will cause the ratio to lose its invariance and show deviations.

However, this stability is guaranteed for appropriately correlated variables only. Hence, our domain specific innovation is to understand how to make it work on active power derived from the PMUs. To achieve this, we aim to find a clustering with an appropriate spatial and temporal granularity that maximizes the correlation between active powers on a given phase across different PMUs in a distribution grid, which ensures invariance in the following metric.

The following framework applies for every phase individually because specific utilities are connected to individual phases which are predefined and they do not switch between phases. Since our anomaly detection depends on unraveling an inconsistency in the system's behavior, the inconsistency check applies only within a given phase. Since the two phases are connected to different consumer loads, it is not possible to do anomaly detection since there is no way to establish the causal link to distinguish a legitimate change from a malicious one. However, within the same phase, since they are attached to loads belonging to similar consumer loads, the method is feasible per phase separately.



(a) (b)

Fig. 7: (a) PMU clustering using correlation among the PMUs; (b) MAD over time windows.

*1) **Optimizing Spatial Granularity**:* Intuitively, a group of PMUs connected to the same feeder or serving proximate geographical areas should exhibit some interdependence in the PMU data streams. We opt to utilize the pairwise Pearson correlation coefficient to identify clusters that show some level of positive correlation. Please note, higher the desired level of invariance, the higher is the required level of positive correlation. We calculate Pearson's correlation among all possible pairs of PMU devices in the data to find groups having a maximum correlation. In the EPFL dataset, the mean of hourly correlations between PMU 2, 3, 4, 5 are shown in Fig. 7a. It is evident from the mean correlations that PMU2, PMU3, and PMU5 show a strong correlation and can be considered in a single cluster. The correlation identifies PMUs to be clustered under one instance of the anomaly detection technique. A similar clustering for the LBNL data is available in our previous work [1]. However if the PMU infrastructure is larger than the kind of real deployments considered in our work, then to address the scalability challenge in grouping clusters of PMUs with maximum positive correlation, detailed plan on it, is provided in Sec. VII-B.

*2) **Optimizing Temporal Granularity**:* Now we investigate on selecting the appropriate time granularity over which the ratio metric is to be calculated. The time granularity should be selected in such a way that the invariance in the ratio metric is maximized. In other words, we want to minimize the measure of dispersion in the ratio statistic. Therefore, our approach is to solve the following search problem:

$$T = \underset{T^*}{\operatorname{argmin}} MAD(Q^r(T^*)) \tag{3}$$

where $MAD(Q^r(T^*))$ is the median absolute deviation of the resulting ratio time series with different time granularity and We choose T^* that minimizes the MAD of the ratio time series (shown in Fig. 7b).

After the above process, we get a clusters of PMUs of size N. Since the same principle applies on each cluser the rest of the paper's methodology discusses from a cluster specific perspective. Let $\boldsymbol{P_t} = [P_t^1, ..., P_t^N]$ denote the active power for a phase from N PMUs selected to be in the same cluster at time slot t. We have taken second wise average of active power for our analysis, thus $t = 1$ second.

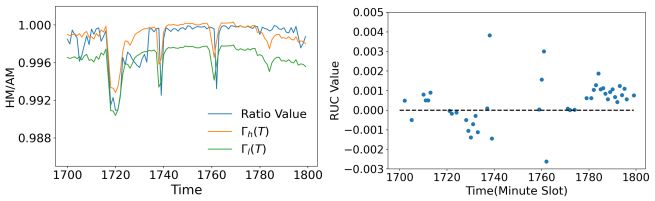
*3) **Harmonic to Arithmetic Mean Ratio**:* Let the harmonic mean (HM_t) and arithmetic mean (AM_t) of P_t at time slot t be defined as:

$$HM_t = N(\sum_{i=1}^{N} P_t^i)^{-1} \quad \text{and} \quad AM_t = \frac{1}{N}\sum_{i=1}^{N} P_t^i. \tag{4}$$

We calculate HM_t and AM_t for slot t over a time window T of length n slots. Then we calculate the average HM_t to AM_t ratio, $Q^r(T)$, at the end of each window as follows:

$$Q^r(T) = \frac{\sum_{t=1}^{n} HM_t}{\sum_{t=1}^{n} AM_t} \tag{5}$$

where $0 \leq Q^r(T) \leq 1$, as $HM_t \leq AM_t$. Compared to normal arithmetic mean proposed $Q^r(T)$ shows high stability as depicted in Fig. 6b.



(a) (b)

Fig. 8: Illustration: (a) Safe Margins (b) RUC(T) samples.

C. Stateless and Stateful Residual based Threshold Design

Now, the anomaly detection needs to identify a proximate spatial region around the ratio time series that specifies the behavior of the invariant under no attacks. Usually, a threshold based approach is considered by tracking the difference between the actual time series value and its smoothed value over time. However, a simple threshold based approach, cannot decrease both false alarms and misdetections simultaneously [17]. Hence, we propose a two-tier approach with stateless and stateful residuals.

*1) **Stateless Residuals:*** The stateless residual denotes the instantaneous difference between the ratio metric and a time varying parameter per time window T. We compute the mean μ_r and Median Absolute Deviation (MAD), m_Q, from the probability distribution of ratio values $Q^r(T)$ for each PMU cluster (shown in Fig. 8a).

We use MAD as a scale parameter for designing the stateless residual instead of the standard deviation (SD), because MAD is more robust to outliers in the training ratio samples. The MAD is more robust than SD since it is based on a squared error from the mean, so a finite number of outliers can increase SD easily compared to MAD, thus reducing sensitivity to small attack strengths.

Stateless residual is parameterized as $\kappa = \epsilon m_Q$ where $\epsilon \in (0, 5]$, such that $\kappa \in (0, 5m_Q]$ and m_Q is the MAD. Intuitively, larger κ values produce wider safe margins, thus reducing false alarms but increasing misdetection and vice-versa. Hence, a trade-off is necessary for choosing a threshold that will automatically generalize into lowering false alarms while not sacrificing the detection sensitivity.

Our framework calculates a parameterized 'stateless residual' $\nabla(T)$ using two values; $\Gamma_l(T)$, and $\Gamma_h(T)$ around the observed instantaneous ratio values $Q^r(T)$ (called safe margin), on each time window on the training dataset, such that:

$$\Gamma_h(T) = Q^r(T) + \epsilon m_Q. \tag{6}$$

$$\Gamma_l(T) = Q^r(T) - \epsilon m_Q. \tag{7}$$

Now the an instantaneous stateless residual $\nabla(T)$ which is the 'signed residual distance' between the observed ratio and the stateless residuals is calculated by:

$$\nabla(T) = \begin{cases} Q^r(T) - \Gamma_h(T), & \text{if } Q^r(T) > \Gamma_h(T); \\ Q^r(T) - \Gamma_l(T), & \text{if } Q^r(T) < \Gamma_l(T); \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

The value of $\nabla(T)$ could be positive (or negative) depending on whether the ratio sample observed is above (or below) the

upper (or lower) safe margin $\Gamma_h(T)$ (or $\Gamma_l(T)$). Thus, $\nabla(T)$ is zero when the ratio observed is within $[\Gamma_h(T)\ ,\ \Gamma_l(T)]$.

*2) **Stateful Residuals:*** Now the framework maintains the sum of residuals between the ratio value and the $\Gamma_h(T)$ and $\Gamma_l(T)$ over a sliding frame of past K time windows. We denote this sum as $RUC(T)$. As depicted in Fig. 8b, this $RUC(T)$ shows more stability compared to stateless residuals. So, the framework calculates the *sum of residuals*, $RUC(T)$ over a sliding frame of past K time windows as:

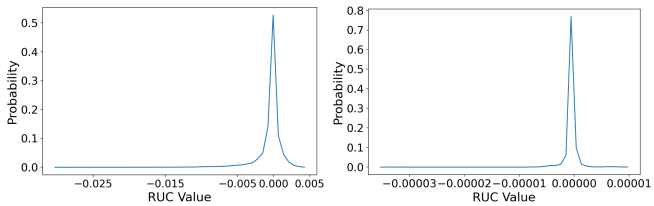
$$RUC(T) = \sum_{f=T-K}^{T} \nabla(f). \tag{9}$$

$RUC(T)$ samples form the feature set which serves an the input to the learning of anomaly detection threshold and thereby attack detection criterion. Note that the $RUC(T)$ may have both negative and positive values. The aim of the learning is to find an upper and a lower thresholds that identify the safe operating region. Any RUC(T) in the test set beyond the safe operating region is an indication of an attack. However, the process of identifying the safe operating region is not easily since the $RUC(T)$ features characterize the underlying benign structure of a noisy system.

D. Theory of Resilient Learning of Detection Criterion

This subsection is not part of the solution framework but contains theoretical underpinnings of the resilient learning of anomaly detection threshold from the latent space of stateless residuals that are derived from the raw data. Although, not part of framework implementation, it is important to discuss here to completely appreciate the learning of the anomaly detection thresholds. For framework implementation, this subsection is not relevant and the reader should refer to the next subsection.

The probability distribution of RUC values from the two datasets is shown in Fig. 9. We observed that RUC values follow a heavy tailed distribution. This is particularly pronounced for the EPFL dataset that has a much longer duration and reveals the true nature of real world PMU data with a higher number of outliers compared to the measure of central tendency (Check Fig. 1a and Fig. 1b). This was an indication that our previous approach [1] is unlikely to offer reliable and robust attack detection performance. This is because large outliers on the tails of the RUC learning feature bias the best fit threshold line towards itself thus widening the limits. We tested this experimentally to verify and it showed severe true positive detection performance degradation under the EPFL dataset when compared to the LBNL dataset. What happened in our preliminary conference work for the LBNL dataset, we just got lucky with a smaller dataset, like other works using the same or similar duration datasets. This is because external drivers of process noise may not manifest them during that small duration being considered. However, active power which is our derived process variable that is influenced by the current data/load drawn, is directly dependent on occasional unpredictable seasonal/weather events that are not representative of the most general pattern but also correspond to the presence of attacks. These occasional unpredictable events create a process noise in our active power variable. This indirectly gets captured into the $RUC(T)$ stateless residual metric (which acts as a latent space), that contains outliers that do not represent the most general underlying structure of benign data, but also are not related to attacks. The robust learning needs to balance these two seemingly opposing perspectives. Hence, the conclusion is that neither we can exclude outliers completely, nor include outliers completely. From the above, we need an approach of robust learning of the detection thresholds that is a compromise between these extremes.



(a) (b)

Fig. 9: Dataset Distribution: (a) EPFL Data; (b) LBNL Data.

Theoretical Intuition of Robust Estimator Let us assume that there are m training input RUC(T) values each denoted by r_k, such that $k \in 1, \cdots, m$. Our job is to learn best fit threshold say τ_{opt} based on the known r_k. The search space of τ_{opt} is the parameter estimation search space denoted by τ. For simple linear regression the typical parametric assumption is a Gaussian distribution under maximum likelihood estimation. The estimate of the parameter is the one that maximizes the likelihood function. More generally, the likelihood function for parameter estimation is specified under an parametric assumption on a distribution of choice.

To understand the best parametric distribution of choice for our purpose, we tried fitting the RUC sample distribution against known distribution models in parametric statistics (such as normal, beta, gamma, t-distribution) using a Kolmogorov–Smirnov test [29] to identify the best fitted distribution model. We calculated the sum of squared error for each distribution model for model selection and found that the student t-distribution, gives the lowest sum of squared errors compared to all other distributions. Hence, the parametric distribution of choice for us was the student t-distribution.

Following the theory of t-distributions, we know that the pdf of a t-distribution is not expressed in terms of the raw input (r_i in our case) but a standardized input. The standardizing involves the error between the raw input and the parameter that is being estimated, divided by the standard deviation of the raw input. In our problem, this takes the recognizable form $\frac{r_k-\tau}{\sigma}$. However, [28] observes that this does not take into account varying scale ranges of the input. Therefore, a more general form of the t distribution is more accurate. Therefore, letting $r_k - \tau = s_k$, the pdf of the student's t-distribution with auxiliary scale correction is given by:

$$f_\tau(s_k) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi(c\sigma)^2}\Gamma(\frac{\nu}{2})}\left(1 + \frac{{s_k}^2}{\nu(c\sigma)^2}\right)^{-\frac{\nu+1}{2}} \tag{10}$$

where Γ is the gamma function, ν is degree of freedom, s_k is error difference between a candidate parameter choice τ and the training point r_k, σ is the standard deviation of the r_k values, and c is a tuning constant for auxiliary scale correction.

In parametric estimation, it is well known that optimizing the likelihood function is the same as optimizing the log-likelihood function. Therefore, we can write the log-likelihood function for estimation as $\rho = -log(f(s_k))$. In general, the parameter estimate in MLE is found by empirical risk minimization problem which can be expressed by the following optimization problem.

$$\underset{\tau}{\operatorname{argmin}} \sum_{k=1}^{m} \rho(s_k) \quad \text{where} \quad \rho(s_k) = -logf(s_k) \tag{11}$$

With some algebra and derivation it can shown that negative natural logarithm of pdf of t- distribution is given:

$$\rho(s_k) = \frac{\nu+1}{2} ln\left(1 + \frac{s_k^2}{\nu(c\sigma)^2}\right) + constant \tag{12}$$

Given a degrees of freedom ν, we can derive a generalized MLE which is an M-estimator of estimated parameter τ by finding the solution to the above empirical risk minimizer which considers ρ for all training data points $i \in \{1, N\}$ such that the M-estimator is defined as a solution of following m summation equation for τ under a t-distribution

Theory of M-estimators in robust estimation states that finding the parameter estimate via minimizing log-likelihood function is the same as finding the solution to $\sum_{k=1}^{m} \psi(s_k) = \sum_{k=1}^{m} \psi(r_k, \tau) = 0$

$$\text{where} \quad \psi(s_k) = \frac{\partial \rho(s_k)}{\partial s_k} \tag{13}$$

With some algerbra and derivation, it can be shown that the derivative $\frac{\partial \rho}{\partial s_k}$ (of Eqn. 12) is Influence Function(IF) of an M-estimator under a t-distribution [28] as a parametric choice for learning the estimate.

$$\frac{\partial \rho(s_k)}{\partial s_k} = \frac{\nu+3}{\nu + (\frac{s_k}{c\sigma})^2}(s_k) \tag{14}$$

In theoretical robust statistical estimation, derivative of a cost function with respect to the error residual (that is directly dependent on the input) is called the influence function. So we can say the following is an influence function.

$$\implies IF = \frac{\partial \rho}{\partial r_k} = \frac{\nu+3}{\nu + (\frac{r_k-\tau}{c\sigma})^2}(r_k - \tau) \tag{15}$$

Now in our problem, we have just parameter that is free to change statistically and we only sample one RUC(T) value in a window. Therefore, according degrees of freedom definition in t-distribution small values of degree of freedom give heavier tail to the t-distribution and for bigger values it resembles a Gaussian distribution. Therefore, from both perspectives, our pick of the degrees of freedom $\nu = 1$ is sound. Different large values of RUC will affect the value of IF of the location estimator τ which will be influenced highly. However, these large values of ruc will have less affect the learnt best fit of τ under this the t-distribution as the parametric choice for the MLE and this IF. Therefore, putting $\nu = 1$, in Eqn. 10, we get the influence function appropriate for our purpose as:

$$IF = \frac{4(r_k - \tau)}{1 + (\frac{r_k-\tau}{c\sigma})^2} \tag{16}$$

From the above equation, let $c\sigma = \beta$, be a scaling hyperparameter. This modeling relaxation is useful since in a t-distribution the standard deviation is biased due to small number of large outliers and its true value is often unknown. Given, we know already that $r_k - \tau = s_k$, the relaxed influence function can be written as:

$$IF = \frac{4(s_k)}{(\frac{s_k}{\beta})^2 + 1} \tag{17}$$

To figure out the most appropriate estimator for our problem, we need to take the integral of the influence function with respect to s_k. This follows from theoretical robust estimation theory, since the derivative of the estimator with respect to the input is the influence function, we can do the integral of the influence function to obtain the appropriate estimator. By taking integral over Eqn. 17 and with considering parameter correction for auxiliary scale we find

$$\int \frac{4s_k}{1 + (\frac{s_k}{\beta})^2} ds_k \tag{18}$$

Let $u = 1 + (\frac{s_k}{\beta})^2$, then it can be shown $ds_k = \frac{\beta^2}{2s_k} du$. Plugging it back into the integral and substituting the value of u, it can be shown that we get

$$\beta^2 \log(1 + (\frac{s_k}{\beta})^2) = L \tag{19}$$

We found that the above functional form has been reported in computer vision and robust statistics literature as a Cauchy-Lorentzian function. Given, the above analytical reasoning, we were convinced that among the several possible approaches to resilient learning, that this is the most appropriate choice of the estimator that is robust to the heavy tails in our derived feature space $RUC(T)$. Therefore, we reason the use of Eqn 19 as the most appropriate choice of loss function to learn and predict the standard limit thresholds.

Analyzing Performance Overhead with Robust Estimator

We need to analyze whether introducing the Cauchy Lorentz loss in place of the previously used L1 norm in our previous work, comes at a sacrifice of increased overhead?. Interestingly, the Cauchy Lorentz loss turns out to be differentiable at all points, unlike the L1 norm (in the previous method), which not differentiable at all points due to gradient shattering. Therefore, to get exact optimal result, the previous method required a brute force approach to learn the two thresholds. In contrast, our new choice of Cauchy Lorentz is convex and differentiable at all points, and therefore gradient descent directly applies. Therefore, our new modification does not require a brute force approach to find the minima. Now, it is well known that gradient descent has equal to (in the best case) or less overhead to converge to a minima compared to a brute force approach. While we theoretically optimized why the L1 should be changed to Cauchy Lorentz to improve robustness against outliers and noise, the side benefit of this design change is that it does not come at the cost of increased overhead to learn the thresholds that dictate the attack detection criterion.

E. Robust Threshold Learning Approach

We need to learn an upper threshold τ_{max} and a lower threshold τ_{min} from the RUC values, which act as unknown model parameters to be estimated. The RUC values which go beyond these thresholds indicate attacks as they do not confirm the underlying structure learned during benign operations, In our previous work [1], we used a simple regression to calculate the best fit line for the upper and lower thresholds. However, given the larger time horizon dataset of EPFL with more PMUs and more outliers due to process noise, that approach cannot compensate for outlying RUC values, which form the training data. Given the heavy tailed nature of the RUC latent feature, we use the derived loss function under an MLE with a t-distributed fitting.

Therefore, in our training, we replace the regular regression ordinary least squares with 19. Additionally, the negative and positive RUC values have a different range. Therefore, we introduce the scaling hyper-parameter β for the 19 should be cross-validated separately as β_+ in RUC^+ and β_- in RUC^- data training. Therefore, we calculate both upper standard limit (τ_{max}) and lower standard limit (τ_{min}), for different values of the optimization parameter β_+ for τ_{max} (and β_- for τ_{min}).

Another key observation we had is that the number of points was not symmetrically distributed around the candidate parameter τ. Some outlying points are still representing some benign and legitimate behavior of the PMU system. Ignoring them completely in the learning process, would yield an overly restrictive system. Therefore, we need to treat the error between the data point and the candidate fit τ.

Algorithm 1 Robust Learning of τ_{max}

Input: $RUC(T)$=Training Input, $[\tau]$= All Candidate Parameter Choices, β_+ = scaling hyper-parameter , w_1, w_2= Quantile Regression Weights

Result: τ_{max}

```
for τ ∈ [τ] do
    cost_+ = 0
    for r ∈ RUC(T) do
        if r > 0 then
            s = r − τ
            if s >= 0 then
                cost_+ : log(1 + (s*w_2/β_+)^2)
            end
            if s < 0 then
                cost_+ : log(1 + (s*w_1/β_+)^2)
            end
        end
    end
    TotalCost : β_+^2 * cost_+
end
τ_max = argmin_τ(TotalCost)
```

Hence, for learning τ_{max}, since we do not want the higher residual r values to be completely ignored by the robust loss function, we add a higher weight w_2 to the error corresponding to $s > 0$, which happens when $r > \tau$. As we have calculated τ_{max} and τ_{min} separately, in our algorithm we simply used r and s instead of r_k and s_k respectively. In contrast, we assign the errors s a weight $w_1 < w_2$, if $s < 0$, which corresponds to points $r < \tau$ or the points with smaller r.

Description of Algorithm 1: First, for each potential τ out of $[\tau]$, we calculate the total loss over positive $RUC(T)$ (which is essentially $RUC^+(T)$) by assigning different weights (w_1 and w_2) depending on whether the $RUC(T)$ is higher or lower than the τ. The regression errors are represented by ($r-\tau = s$), which is the difference between the candidate parameter fit. If the data point r is outward and the candidate fit τ is inward, it corresponds to the situation where $s > 0$ while learning the upper threshold τ_{max}. However, the current choice of τ will be equivalent to a false alarm, since we are learning over benign data. Due to the base rate fallacy, we know that false alarm reduction is more critical for anomaly based intrusion detection systems. Therefore, the scenario $s > 0$ needs to have a different weight w_2, compared to the opposite scenario (where $s <= 0$ that corresponds to a missed detection). Therefore, in our learning framework, firstly $w_2 \neq w_1$ and $w_2 > w_1$ and balances the false alarm trade-off in a way that handles base rate fallacy. Since loss function quantifies goodness of fit per training data point, the position of the data point w.r.t to a candidate parameter fit plays a role. Across all training data points, it acts as a weighted sum of Cauchy loss where the weights w_1 and w_2 are controlled by the position of the point. The Cauchy loss helps reduce the impact of outliers while the quantile weights help reduce the chances of false alarms while doing so.

Cross Validation: Finally we use a cross validation set with minimum δ_{avg} that we target to detect and calculate *false alarms (FA)* and *Mis-Detection (MD)*. Specifically, we use the following optimization to select optimal τ_{max} and the corresponding β_+. Algorithm 1 provides the optimization of the upper standard limit. A similar approach is used for lower standard limit (τ_{min}) as well (when $r < 0$), with minor changes to keep the same logic.

For learning the hyperparameters β_+, w_1, w_2, we use a cross validation set which uses the following criterion:

$$\underset{\beta_+, w_1, w_2}{\operatorname{argmin}} \left(d_1 FA^{cv} + d_2 MD^{cv}\right) \tag{20}$$

where d_1, d_2 are trade-off weights. The d_1 represents the importance of reducing false alarms and d_2 represents the importance of reducing missed detection. We have $d_1 + d_2 = 1$ and $d_1 > d_2$ such that we give more importance to the false alarm rate compared to misdetection rate. This is because the probability of an actual attack is very low, and seemingly low false alarm rates, do not necessarily indicate a good usable attack detector. In the end, we choose a threshold τ_{max} (and τ_{min}) which minimizes the total loss for the corresponding RUC inputs.

F. Detection Criterion in Test Set

The main idea behind attack detection is that RUC in the test set ($RUC(T^C)$) should not deviate from the standard limit obtained from the training set. We first calculate the stateless residuals for each time window of the testing set T^C such that $\Gamma_h(T^C) = Q^r(T^h) + \kappa_{opt}$ and $\Gamma_l(T^C) = Q^r(T^h) - \kappa_{opt}$, where κ_{opt} is the margin that resulted in the optimal standard

limit. The historical value of the ratio on that time window $Q^r(T^h)$, where T^c is the current time window and T^h is the corresponding time window in the training set, $\Gamma_{high}(T^c)$ and $\Gamma_{low}(T^c)$ are the safe margins at T^c of the test set.

From $\Gamma_h(T^C)$ and $\Gamma_l(T^C)$, we calculate the $RUC(T^C)$ using Eqn. 21. Then we check whether $RUC(T^C)$ violates the standard limit range identified during the training set.

$$RUC(T^c) : \begin{cases} \in [\tau_{min}, \tau_{max}], \text{No Anomaly;} \\ \notin [\tau_{min}, \tau_{max}], \text{Anomaly.} \end{cases} \quad (21)$$

TABLE I: Parameter Descriptions and Values

Parameter	Symbol	EPFL	LBNL
Spatial Granularity	N	PMU 2,3,5	PMU 1,3
Temporal Granularity	T	600	60
Sliding Frame	K	10	5
Scaling	β_+, β_-	.0008,.0008	.006,.002
Weights	w_1, w_2	.5, 2	.5, 2
Trade-off Weights	d_1, d_2	1/3, 2/3	1/3, 2/3

V. Experimental Evaluation

We have used both the LBNL [16] and EPFL [20] dataset for our experimental results. For the LBNL dataset, the total available data is of 11 days from 1-Oct-2015 to 11-Oct-2015. We have used the first 7 days of data as training and next 2 days as cross validation and the final 2 days as our testing set.

For the EPFL dataset, we have considered a total of 6 months of data (Sep-14 to Feb-15) in our experiment. The first 17 weeks are selected as training and next 4 weeks as cross validation and the final 4 weeks of data as testing set.

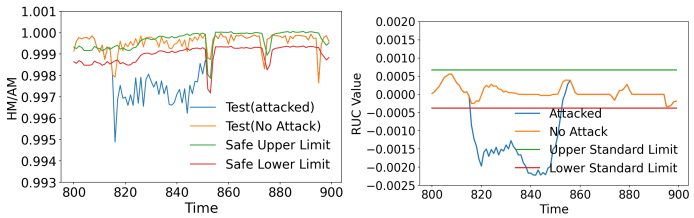
We conducted extensive experiments for different falsification margins and attack strategies. We divide this section into two parts: (1) Snapshot Results These results are illustrative results to aid in the explain-ability of how and why the framework is successful in detecting the attacks under various attack types and strategies and what kind of signatures are observed under attacks; (2) Performance Evaluation that shows the true positive rate versus the false alarm across varying attack margins. The basic parameters selected for the EPFL data are shown in Table I and the same information for the LBNL data is available in our previous work [1].

Details of Testing Set and Attacks: For false alarm performance, we calculate the base rate false alarm, which is the false alarm rate in the presence of no attacks throughout the whole testing set. For attack detection performance, we introduced 100 (20 for LBNL data) short term attacks of length 2 hours distributed throughout the test period. We continue to do these attacks on different PMUs. Then we check the average detection accuracy by measuring the fraction of total attacked samples which violated the standard limits to prevent bias in the time periods selected for attacks. The specifics of different attack parameters are discussed while explaining snapshots and performance evaluation under each attack type, strength, scale, and strategy. This is done because we are exploring the full strategy space of possible FDI attacks, instead of a specific instance. Such a parameterized approach to threat modeling leads to a more unbiased evaluation.

A. Snapshot Results

For all snapshot results, we have the attack strength of $\delta_{avg} = 0.5$ amps on the current magnitude of a single PMU (PMU2), phase 1, from the EPFL dataset, and the attacks continue for a period of 2 hours from the test set. We select a small time period to prove the visual intuition and explainability. Rigorous experimental evaluation with the whole duration of the testing set is reported in the performance evaluation subsection. We do not show the snapshot results on the LBNL dataset since most of the corresponding results are already available in our previous work [1].

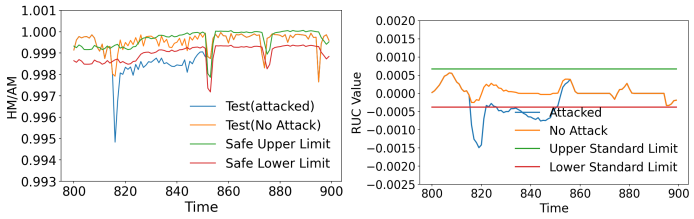
We first show the impact of the deductive type of attacks with a step strategy. Fig. 10a confirms the deviation of the ratio metric beyond the set safe margin, and the Fig. 10b, shows that the $RUC(T)$ feature during the test set, successfully violates the learnt standard limit thresholds τ_{max} and τ_{min} from the training and cross-validation.



(a) (b)

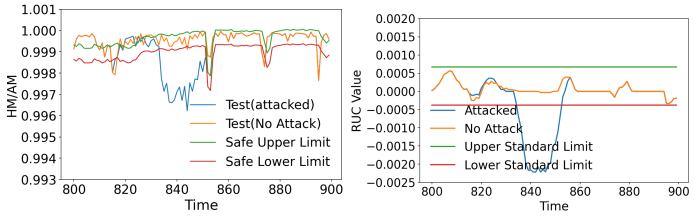
Fig. 10: Deductive Attack with $\delta_{avg} = 0.5$ amps (a) Ratio Snapshot; (b) $RUC(T)$ Snapshot.

Similarly, for additive type of attack with step strategy, the behavior of the ratio and the stateful residual under a deductive attack is shown in Fig. 11a and Fig. 11b respectively. The conclusions on successful deviation and violation of the anomaly detection criterion via the learned thresholds can be visually confirmed.



(a) (b)

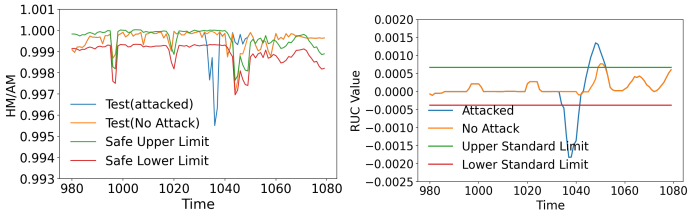
Fig. 11: Additive Attack with $\delta_{avg} = 0.5$ amps (a) Ratio Snapshot; (b) $RUC(T)$ Snapshot.



(a) (b)

Fig. 12: Alternating Switching Attack with $\delta_{avg} = 0.5$ amps (a) Ratio Snapshot; (b) $RUC(T)$ Snapshot.

Fig. 12a and Fig. 12b. validate our success in creating deviations in both the ratio and thereby the $RUC(T)$ metric to violate the learned attack detection criterion under an alternating switching attack type that as discussed in our threat model.



(a) (b)

Fig. 13: Anomaly Detection for Mirroring Attack for $\Delta_a = 1$ hour (a) Ratio Snapshot; (b) $RUC(T)$ Snapshot.

Finally, the detection of the mirroring attacks are shown in Fig. 13a and Fig. 13b. To implement such attacks we have selected a random period of one hour of a PMU and used the mirror image of the captured time series for the next hour.

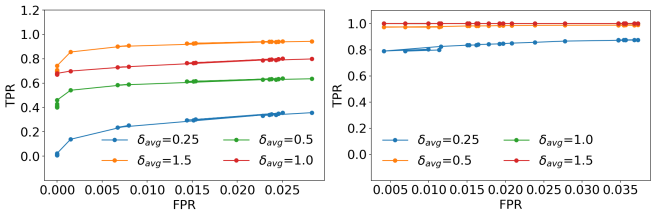
B. Performance Evaluation

For performance evaluation, we first generate the ROC curve that characterizes the trade-off between the probability of attack detection vs. the probability of false alarm.

To remove attack period selection bias we introduced continuous attacks for the whole testing period with different attack margins and measured the % of sample points outside the standard limit boundaries for different scalar factors (ϵ).

We introduced attacks on each PMU and have taken the average detection and false alarm rate to remove the bias of a selected PMU for an attack scale of 1 PMU and 2 PMU compromised out the cluster of 3 PMUs.

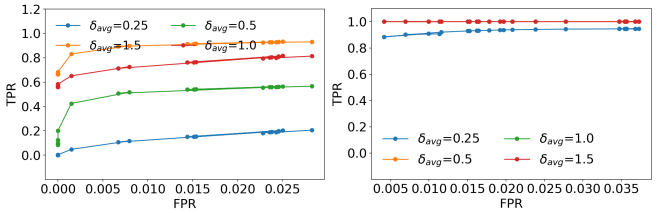
1) **Under Various Attack Types and Strengths:** While we do not limit attack margins within any arbitrary range to keep an unbiased evaluation, we stopped reporting performance for attack strengths where our missed detection rate becomes 50% or worse, since that would mean a random coin toss will be a better detector. Hence, the lowest attack strength shown corresponds to this performance limit.



(a) (b)

Fig. 14: Performance under Deductive Attack for varying Strengths: (a) EPFL Data (b) LBNL data

Fig. 14a and 14b depicts the ROC of deductive type of attacks with step strategy for EPFL and LBNL dataset respectively for different values of δ_{avg}. Similarly, Fig. 15a and 15b depicts the same ROC for additive type of attacks with step strategy for both the datasets. For low margin of attacks in the EPFL setup, as shown in Fig. 14a and Fig. 15a, when

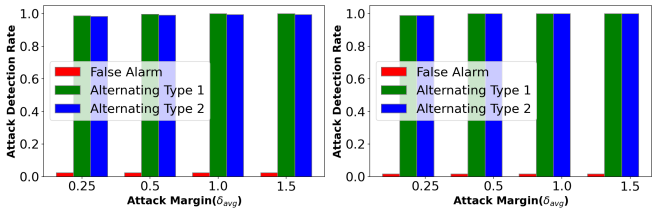


(a) (b)

Fig. 15: Performance under Additive Attack for varying Strengths: (a) EPFL Data (b) LBNL Data

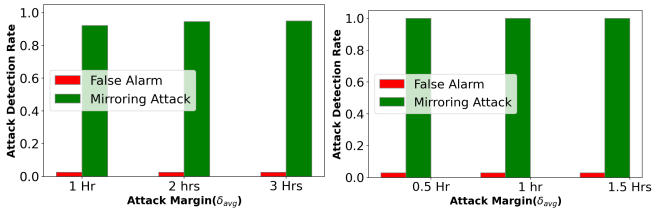
the margin is 0.25 amps the method performs poorly with a misdetection rate of $>= 0.5$, which is the evasion point of our method. However, any attack strength above 1.5 amps reaches a 0.98 detection rate even as the false alarm rate is 0.025. Such low false alarm rates are a significant improvement given that is the main challenge of anomaly based approaches.

2) **Under Various Attack Strategies:** For any anomaly detection mechanism, it is essential to investigate the false alarms raised and ROCs cannot always be the primary evaluation metric for that. So to compare this, we first introduced 100 (20 for LBNL data because of its shorter duration) short term attacks (of length 2 hours) on different PMUs and checked the average detection accuracy by measuring the fraction of total attacked samples which violated the standard limits. Then we calculated the false alarm raised in the presence of no attacks in the selected attack periods.



(a) (b)

Fig. 16: Performance under Alternating Attack for varying Strengths: (a) For EPFL Data; (b) For LBNL Data.

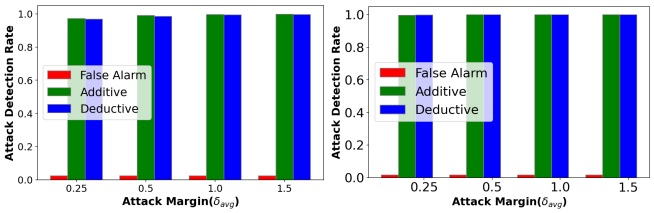


(a) (b)

Fig. 17: Performance under Mirroring Attack for varying Strengths: (a) For EPFL Data; (b) For LBNL Data.

Fig. 16a and Fig. 16b shows the performance of our model under *Alternating* types of attacks for EPFL and LBNL data respectively. For *Alternating Type 1* the adversary uses additive attack for the first half and then uses deductive attack for the second half of the attack period. *Alternating Type 2* indicates the opposite of *Type 1* where the adversary uses deductive first and then uses the additive attack. Now for mirroring type of attacks, we have selected different lengths of attack periods.

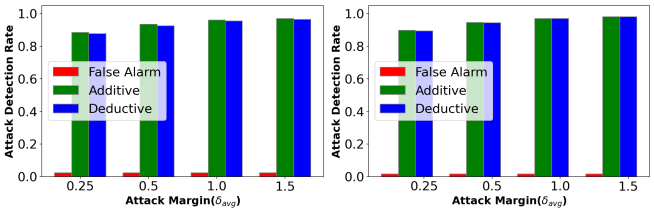
We capture the data from a PMU of the selected time period and used the mirror image of the captured data for the same amount of time and we implemented 100 such attacks. The detection rate with various attack period length is shown in Fig. 17a and Fig. 17b.



(a) (b)

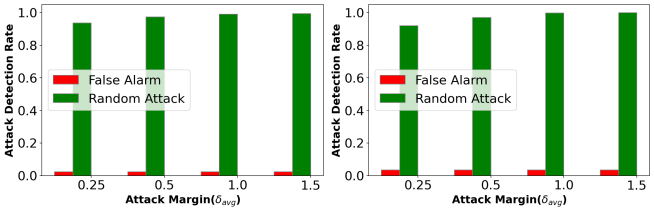
Fig. 18: Performance under Step Strategy: (a) For EPFL Data; (b) For LBNL Data.

Fig. 18a and Fig. 18b shows the true detection averaged over all the PMUs for EPFL and LBNL dataset respectively for step attack strategy. It is evident from the plots that our method achieves detection accuracy ≥ 0.95 for both the deductive and additive attack types on both datasets. Similarly, Fig. 19a and Fig. 19b shows the detection rate under Ramp attack strategy averaged over all the PMUs for EPFL and LBNL dataset respectively. Now, Fig. 20a and 20b depicts the performance against random attacks for both the dataset.



(a) (b)

Fig. 19: Performance under Ramp Strategy: (a) For EPFL Data; (b) For LBNL Data.

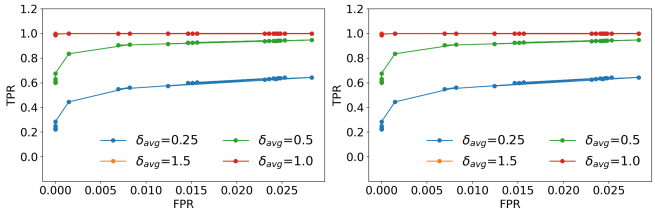


(a) (b)

Fig. 20: Performance under Random Strategy: (a) For EPFL Data; (b) For LBNL Data.

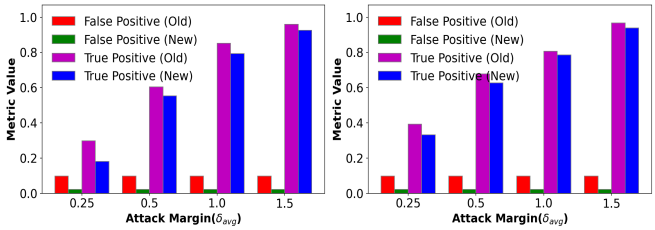
3) Scale Sensitivity: Figs. 21a. and 21b prove that the attack detection still works with very low false alarms even when 2 out of the 3 PMUs in the cluster are compromised with false data injection.

4) **Improvement with Resilient Learning:** We have compared our previous [1] and the proposed noise resilient learning found improved performance. We have simulated the same attack strategy and attack margin on the EPFL dataset and compared the ROC curves.



(a) (b)

Fig. 21: Performance when 2 PMUs attack simultaneously (EPFL): (a) Additive Attack; (b) Deductive Attack.



(a) (b)

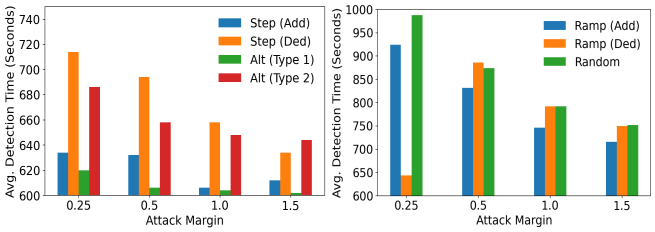
Fig. 22: Comparison Between Old and New Method on EPFL dataset: (a) For Additive Attack; (b) For Deductive Attack.

Fig. 22a and 22b shows the comparison of the previous and the new method for additive and deductive attacks respectively. It is evident that although our detection rate (or true positive) drops a little compared to the old method, we can still reduce false alarms (false positive) significantly. As we mentioned in Section IV.D, since the actual probability of a PMU system being under attack is less (base rate fallacy), therefore, we should not ignore the importance of the false alarms improvement.

Our new method produces much better false alarm performance compared to our old model. As depicted in Fig. 22a and Fig. 22b, the new model reduces the average false alarm rate by more than 80% (from .105 to .019). However, we have observed minuscule degradation in missed detection performance of the new method. For any attack margin $>= 1$ amps, we have seen only 4.9% (for additive type of attacks) and 4.2% (for deductive type of attacks) drop in the average missed detection performance which is much lesser than the average improvement of the false alarm performance.

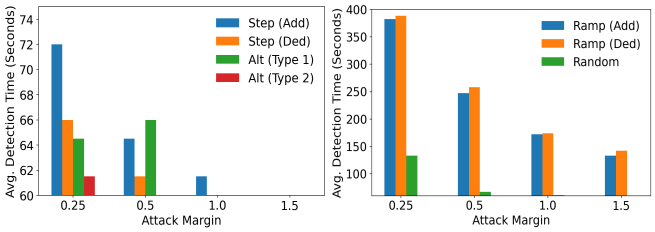
5) **Real Timeliness of Detection:** To demonstrate the real-time nature of the detection method, we have estimated the *Average Attack Detection Time* for different attack types and strategies. For the EPFL data, as we have shown in our parameter description table, the temporal granularity is estimated to be 600 seconds. The average detection time for this EPFL data is shown in Fig. 23. It is evident from Fig. 23a that even with our relatively higher temporal granularity, the average delay is less than 12 minutes even for low margin (.25 amps) of additive, deductive, or alternating attacks with step strategy. However, for a low margin of random attacks, our model produces an average delay of less than 17 minutes as shown in Fig. 23b.

For the LBNL data, the average detection time is even less. First of all, as the LBNL data is relatively stable, the temporal granularity is estimated to be 60 seconds. Based on that, the average delay in attack detection is shown in Fig. 24a for



(a) (b)

Fig. 23: Average Attack Detection Time for EPFL Data: (a) Step Strategy (b) Ramp and Random Strategy.

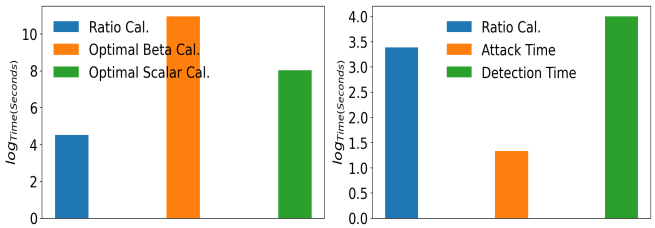


(a) (b)

Fig. 24: Average Attack Detection Time for LBNL Data: (a) Step Strategy (b) Ramp and Random Strategy.

additive, deductive, or alternating attacks with step strategy. For a low margin of ramp attacks, our model produces an average delay of less than 7 minutes as shown in Fig. 24b. This illustration corroborates the near real-time applicability and quick FDI attack identification of our proposed model.

6) **Run Time Complexity of Our Method:** For the demonstration of how light-weight our framework is we have explored the average running time (measured in seconds) on an Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz 16 GB RAM computer for training and testing of the proposed model. The running time of the crucial steps of training and testing is shown in Fig. 25. The 95% of the total training time is spent on the optimal τ_{max}, τ_{min} calculation for each β_+ and β_- respectively. However, once the training is done and all the parameters are learned, the complete testing phase with 4 weeks of data takes only 88.22 seconds. We can therefore conclude that the method does not much time or resource



(a) (b)

Fig. 25: Average Running Time: (a) Training (b) Testing.

VI. Related Work

Most of the works related to PMU data falsification can be broadly divided into two categories - state estimation based detection, and machine learning based detection. State estimation based approaches are typically model driven. For example, in [12], a mechanism based on continuous monitoring of phase-wise voltage in an equivalent transmission line was proposed, for detecting data falsification in PMUs. However, they require two PMUs deployed at both ends of the transmission line. More importantly, we found that the PMU data streams at transmission level were inherently stable making anomaly detection a less challenging problem.

In [38], the authors have proposed a multisensor track-level fusion-based model prediction (TFMP) to estimate electromechanical power oscillations modes during data-injection attacks. However, it did not discuss the detection of FDI attacks, instead, the algorithm is utilized to remove bias and noise while accurately extracting the system parameters. Also the proposed method only monitors phase angles and is tested on a IEEE 39-Bus New England System. Similarly, The [39] proposed a Bayesian-based approximated filter (BAF) to improve the immunity of the monitoring of power oscillation against data-injection attacks. The predictive distribution property of the this algorithm supports monitoring power oscillation in the presence of FDI with high probabilities of information loss. As both of these papers only monitors the phase angles, they are not suitable to detect FDI attacks in other streams such as current or voltage magnitude.

In [40], the authors presented an algorithm to detect line outage by measuring the phase angles and then comparing it to a threshold value to identify the event. However, they use actual topology of the network as one of the input parameters which is very critical information and very hard to obtain. Also, their approach is said to be applicable in transmission line network whose measurement data is inherently stable compared to distribution line network, which we have considered. In [41], the authors proposed a convex optimization method for detecting line failures and exploit Bayesian regression to develop an algorithm for probabilistically detecting line failures after an attack using partial noisy measurements. Their model is numerically validated against DC power flow which mainly focuses on the event on line tripping which is a special case of FDI attack.

In [32], first it is demonstrated that the conventional bad data detectors are less effective against sophisticated FDI attacks mainly because they are based on the classical weighted least-square estimation where redundancy is a must in detecting the bad data [36]. Then a decentralized homomorphic computation paradigm is proposed along with a hierarchical knowledge sharing algorithm to facilitate the secure ciphertext calculation of state estimation. However, the method is computationally heavy and no real data is used to demonstrate the efficacy of the proposed approach. In [33], an FDI detection method based on the Kullback–Leibler distance (KLD) which calculates the distance between two distributions, p and q. p is derived from the measurement variation between the current and the previous time step and q is derived from the historical data. However, the minimum attack margin that is demonstrated is comparatively higher than our approach and it is not explained how the proposed method can be implemented in a PMU-PDC setting. Similarly, in [21] the authors designed statistical testing with sliding windows that detect anomalies in a single stream by generating an equivalent circuit to identify a specific

event. Please note, each PMU generates 12 streams of data when deployed in a 3-phase grid network, thus creating a huge computational overhead for stream specific model driven approaches.In [37], a novel approach is presented to enhance the resilience of wide-area control systems against the FDI attacks. The temporal prediction attribute of the proposed model is able to parry the FDI attacks while estimating and controlling the voltage magnitude. However, the proposed model only validated against the voltage magnitude, so there is a need to implement models/methods for the other data streams as well.

On the other hand, machine learning based approaches such as [22] propose to use clustering of the sets of events distinguish different events from one another. However, the designs are built on a single data stream, thus, can result in computational overhead. The survey article, [35], discussed different ML models, both supervised and semi-supervised, and unsupervised ML methods. However, the high computational overhead and the lack of validation on real data raises the question of reliability and the possibility of using these methods in real time. The [15] proposed a decision tree based anomaly detection scheme to differentiate between normal tripping and malicious tripping by training on specific attack samples. However, it is not feasible to generate 100% of all the possible legitimate line tripping cases for training in [15]. In [18] a density-based local outlier factor (LOF) analysis was used to detect the anomalies among the data, to describe spatio-temporal outliers among all the synchrophasor measurements from the grid. However, this method might not be able to detect attacks in real time, and in their proposed method the authors have only considered an attack on voltage magnitude. The [27] tries to combine both the approaches together by proposing the extraction of events signatures such as line faults and trips, generation and load fluctuations to every cycle of state estimation and then use a Convolutional Neural Network (CNN) data filter to validate the PMU data. Another work, [23], uses a rule-based mechanism to detect perturbations in each data stream independently by identifying optimal placements of PMUs. Most of these works use a single dataset for a few minutes or days or use simulated datasets from controlled testbeds that have lesser noise.

A critical analysis of all previous works on the detection of PMU data falsification revealed that current data falsification for PMU streams was not investigated. Furthermore, we found that, unlike transmission level PMUs, the distribution level PMU's current synchrophasor data shows high dynamic variations in benign conditions, making anomaly detection challenging. Finally, all previous defenses are stream specific in the sense that they only work for either voltage or current falsification. Since each PMU contains 4 streams and has 3 phases, a stream specific defense will require 12 different defense models that need complex cross-coordination.

VII. Discussion

In this section, we discuss the broader scope, limits of applicability, and how to scale up our framework for large scale PMU infrastructure as the electric grids become more and more PMU enabled.

A. Scope of the Framework

First, the presence of a positive correlation among the PMUs sending data to the same PDC/LCC is a required property for this framework to work. Intuitively, if a group of PMUs are connected to the same feeder or serving proximate geographical areas with similar types of customers, then the current drawn exhibit some level of positive correlation. Since voltage and phase angle needs to be regulated within a specified range, the positive correlation in the current implies a positive correlation for the active power on which our invariant is built. The framework will work on distribution level PMU infrastructure, where and when there is the presence of groups of PMU that are positively correlated.

Second, our framework is useful for distribution level PMUs rather than transmission level PMUs, where the data varies more readily due to direct integration with loads and positive correlation may be observed. This does not mean our proposed method would not work on transmission levels PMU clusters. However, as the PMU data is more stable in the transmission line network and the PMUs connected to the same feeder are strongly correlated, our previous method, which we proposed in our conference version [1] would work. Other simple conventional methods based on statistical learning, moving average can also be applied to the FDI attack detection on transmission line PMU-PDC architecture.

Third, in our datasets we observed weak cyclostationarity in the PMU data streams. Since the active power at the distribution level is directly affected by cyclical patterns of human behavior in terms of using electricity on a day-to-day basis. For example, on the weekdays the power measurements peak during mid-day but on the weekends they stay relatively stable throughout the day (See Fig. 6a).

However, if the PMUs connected to the same PDC/LCC are not positively correlated, then either the clusters can not be created or clusters will be formed based on very weak or no positive correlation among the PMUs. This will cause the invariants to have a high variation even under benign conditions, which will prevent us from distinguishing between benign behavior and attacks. Similarly, the positive correlation is also somewhat tied to the cyclical nature of the use of electricity by customer loads. This allows us to use the time context successfully, by using a historical mean value of the ratio invariant on a certain time window of the day that is not arbitrarily different. If for some reason, this is not present, the performance may degrade. Finally, if all PMUs within a cluster are compromised and the attackers have the same attack type, strength, and strategy, then our method cannot detect because if every node in the cluster is compromised, the misalignment in the space time covariance structure [4] required to create deviation is not possible.

B. Applicability for Tackling Large Scale PMU infrastructure

For our framework to scale to larger system than the small testbed deployments, we need to group the large scale system into positively correlated clusters with a causal link. The causal link could be come from topology knowledge, i.e. if two PMUs are connected to completely different feeders with

completely different loads (e.g. residential versus industries or DER microgrids versus non-DER integrated customers). This is because a positive correlation may not have a causal link. So first group the large scale distribution PMU system, such that PMUs within each group has similar customer types/load types. Within each such group say C, further divide it into clusters c_k, such that each cluster maximizes the positive correlation between the active power of the PMUs that are part of that cluster. Such positive correlation maximization is related to achieving high invariance under benign conditions but deviations under attacks as shown in our previous work [4], [14]. Let $Cor(p_i, p_j)$ represent the correlation between the active power data from any two PMUs i and j and let $p^{(min)}$ be a threshold that represents a minimum lower bound on positive correlation such that PMUs can only be eligible to become part of the cluster if their correlation is higher than this threshold.

$$\begin{aligned} \max \quad & \sum_{c_k \in C} \sum_{\{p_i, p_j\} \in c_k} Cor(p_i, p_j) \\ \text{s.t.} \quad & Cor(p_i, p_j) > p^{(min)} \end{aligned} \tag{22}$$

where the c_k is any candidate cluster. The above problem can be solved by converting the PMU system into a graph where the edges represent the correlation level and the vertices represent PMUs. We showed in our previous work in a different context of city scale transportation systems [14], that this clustering could be done via a region growing approximation algorithm, although the optimization problem in Eqn. 22 is NP-hard. Once done, our framework proposed in this paper, can be applied to each of these clusters formed separately (i.e. train and test on each cluster separately). In this way, our method can apply to large scale PMU systems.

VIII. Conclusions

In this work, we presented a noise resilient real time learning framework for attack detection that can detect current magnitude falsification in distribution level PMU data streams. We showed that winsorized harmonic to arithmetic mean ratios can allow a feature space whose benign operating region may be reliably learned even in the presence of noise via robust estimation theoretic approaches. We proved that given the characteristics of the reduced feature space in the form of stateless residuals, we obtain the Cauchy-Lorentz loss function as the best learning estimator for learning the attack classification criterion. We found that our method shows detection accuracy of more than 95% for false alarm rates of less than 5% for various attack types with two different PMU datasets. The performance reported is with a dataset of 4 weeks, which is a longer time horizon compared to other works in this area. Furthermore, this work validates with two different datasets giving credence to the generality of the approach in the context of distribution level PMUs.

Acknowledgments: This work was supported by National Science Foundation grants: SATC-2030611, SATC-2030624, OAC-2017289, DGE-1433659, CNS-1818942.

References

[1] P. Roy, S. Bhattacharjee and S. K. Das, "Real Time Stream Mining based Attack Detection in Distribution Level PMUs for Smart Grids," IEEE Global Communications Conference, pp. 1-6, 2020.

[2] M.S. Mahmoud, H. M. Khalid, and M. M. Hamdan, "Cyberphysical Infrastructures in Power Systems: Architectures and Vulnerabilities", Academic Press, 2021.

[3] S. Ashraf, M.H. Shawon, H.M. Khalid, S.M. Muyeen. "Denial-of-service attack on IEC 61850-based substation automation system: A crucial cyber threat towards smart substation pathways." Sensors 21, no. 19, 2021.

[4] S. Bhattacharjee, S. K. Das, "Detection and Forensics against Stealthy Data Falsification in Smart Metering Infrastructure,", IEEE Trans. on Dependable Computing, Vol. 18(1), pp. 356-371, Jan. 2021.

[5] S. Bhattacharjee, A. Thakur, S. K. Das, "Towards Fast Semi-supervised Identification of Smart Meters Launching Data Falsification Attacks", ACM Asia Conference on Computer and Communications Security, pp. 173-185, 2018.

[6] L.M. Branscomb, R.D. Klausner, "Making the nation safer: the role of science and technology in countering terrorism", National Research Council, 2002.

[7] Defense Use Case, "Analysis of the cyber attack on the Ukrainian power grid," Electricity Information Sharing and Analysis Center (E-ISAC), vol. 388, 2016.

[8] T.M. Chen, "Stuxnet, the real start of cyber warfare?[Editor's Note]" *IEEE Network, vol. 24, no. 6*, pp. 2-3, 2010.

[9] J. Jiang, X. Zhao, S. Wallace, E. Cotilla-Sanchez, and R. Bass., "Mining PMU Data Streams to Improve Electric Power System Resilience," ACM BDCAT, pp. 95-102, 2017.

[10] P. Gopakumar, M. Balimidi, MJB Reddy, and DK Mohanta. "Remote monitoring system for real time detection and classification of transmission line faults in a power grid using PMU measurements." Protection and Control of Modern Power Systems 3, no. 1, pp. 1-10, 2018.

[11] T. Morris, S. Pan, J. Lewis, J. Moorhead, B. Reaves, N. Younan, R. King,M. Freund, and V. Madani., "Cybersecurity testing of substation phasor measurement units and phasor data concentrators" 7th Annual ACM CSIIRW, pp. 12-14, 2011.

[12] S. Pal, B. Sikdar, and J.H. Chow., "Classification and detection of PMU data manipulation attacks using transmission line parameters," IEEE Transactions on Smart Grid, vol. 9, no. 5, pp. 5057-5066, 2017.

[13] D.P. Shepard, T.E. Humphreys, and A.A. Fansler., "Evaluation of the vulnerability of phasor measurement units to GPS spoofing attacks," International Journal of Critical Infrastructure Protection, vol. 5, no. 3-4, pp. 146-153, 2012.

[14] J. Islam, J.P. Talusan, S. Bhattacharjee, F. Tiausas, S. M. Vazirizade, A. Dubey, K. Yasumoto, and S. K. Das. "Anomaly based Incident Detection in Large Scale Smart Transportation Systems." ACM/IEEE 13th International Conference on Cyber-Physical Systems (ICCPS), pp. 215-224. IEEE, 2022.

[15] V.K. Singh and M. Govindarasu., "Decision Tree Based Anomaly Detection for Remedial Action Scheme in Smart Grid using PMU Data," IEEE Power & Energy Society General Meeting, pp. 1-5, 2018.

[16] E. Stewart, A. Liao, and C. Roberts., "Open μpmu: A real world reference distribution micro-phasor measurement unit data set for research and application development," LBNL Tech. Rep. 1006408, 2016.

[17] D.I. Urbina, J.A. Giraldo, A.A. Cardenas, N.O. Tippenhauer, J. Valente,M. Faisal, J. Ruths, R. Candell, and H. Sandberg., "Limiting the impact of stealthy attacks on industrial control systems," ACM CCS, pp. 1092-1105, 2016.

[18] M. Wu and L. Xie., "Online detection of false data injection attacks to synchrophasor measurements: A data-driven approach," HICSS, 2017.

[19] Z. Zhang, S. Gong, A.D. Dimitrovski, and H. Li., "Time synchronization attack in smart grid: Impact and analysis," IEEE Transactions on Smart Grid, vol. 4, no. 1, pp. 87-98, 2013.

[20] M. Pignati, M. Popovic, S. Barreto Andrade, R. Cherkaoui, D. Flores, J.-Y. Le Boudec, M. M. Maaz, M. Paolone, P. Romano, S. Sarri et al., "Real-time state estimation of the epfl-campus medium-voltage grid by using pmus," Sixth Conference on Innovative Smart Grid Technologies (ISGT2015), no. EPFL-CONF-203775, 2014.

[21] F. Mohammad, A. Shahsavari, Emma M. Stewart, and H. Mohsenian-Rad. "Locating the source of events in power distribution systems using micro-PMU data." IEEE Transactions on Power Systems 33, no. 6 (2018): 6343-6354.

[22] D. B. Arnold, C. Roberts, O. Ardakanian, and E. M. Stewart. "Synchrophasor data analytics in distribution grids." IEEE Power Energy

Society Innovative Smart Grid Technologies Conference (ISGT), pp. 1-5. IEEE, 2017.
[23] J. Mahdi, A. Scaglione, C. Roberts, E. Stewart, S. Peisert, C. McParland, and A. McEachern. "Anomaly Detection Using Optimally Placed μPMU Sensors in Distribution Grids." IEEE Transactions on Power Systems 33, no. 4 (2017): 3611-3623.
[24] A. Armin, M. Farajollahi, and H. Mohsenian-Rad. "Unsupervised learning for online abnormality detection in smart meter data." IEEE Power Energy Society General Meeting (PESGM), pp. 1-5. IEEE, 2019.
[25] K. Fu, W. Xu, "Communications of the ACM", Vol. 61(2), pp. 20-23, 2018.
[26] M. Brown, M. Biswal, S. Brahma, S. J. Ranade and H. Cao, "Characterizing and quantifying noise in PMU data," IEEE Power and Energy Society General Meeting (PESGM), 2016, pp. 1-5.
[27] S. Basumallik, R. Ma, and S Eftekharnejad. "Packet-data anomaly detection in PMU-based state estimator using convolutional neural network." International Journal of Electrical Power & Energy Systems 107, pp. 690-702, 2019.
[28] Sumarni, C., et al."Robustness of location estimators under t-distributions: a literature review." IOP conference series: earth and environmental science. Vol. 58. No. 1. IOP Publishing, 2017.
[29] Massey Jr, Frank J. "The Kolmogorov-Smirnov test for goodness of fit." Journal of the American statistical Association 46, no. 253, pp. 68-78, 1951.
[30] G. Hug and J. A. Giampapa, "Vulnerability assessment of ac state estimation with respect to false data injection cyber-attacks," IEEE Transactions on Smart Grid, vol. 3, pp. 1362–1370, 2012.
[31] D. Urbina, J. Giraldo, A. Cardenas, J. Valente, M. Faisal, N. Tippenhauer, J. Ruths, R. Candell, and H. Sandberg, "Survey and New Directions for Physics-Based Attack Detection in Control Systems" NIST Report, 2016.
[32] B. Li, R. Lu, G. X., L. Tao, and K. R. Choo, "Detection of false data injection attacks on smart grids: A resilience-enhanced scheme." IEEE Transactions on Power Systems, vol. 37, no. 4, pp. 2679-2692, 2022.
[33] G. Chaojun, P. Jirutitijaroen, and M. Motani, "Detecting false data injection attacks in AC state estimation." IEEE Transactions on Smart Grid 6, no. 5, pp. 2476-2483, 2015.
[34] V.K. Singh, A. Ozen, and M. Govindarasu. "A hierarchical multi-agent based anomaly detection for wide-area protection in smart grid." IEEE Resilience Week (RWS), pp. 63-69. 2018.
[35] U. Inayat, M.F. Zia, S. Mahmood, H.M. Khalid, M. Benbouzid. "Learning-Based Methods for Cyber Attacks Detection in IoT Systems: A Survey on Methods, Analysis, and Future Prospects." Electronics 11, no. 9 (2022): 1502.
[36] A. Abur, A.G. Exposito, "Power system state estimation: Theory and Implementation", *CRC press*, 2004.
[37] A.S. Musleh, H.M. Khalid, S.M. Muyeen, A. Al-Durra. "A prediction algorithm to enhance grid resilience toward cyber attacks in WAMCS applications." IEEE Systems Journal 13, no. 1, pp.710-719, 2017.
[38] H.M. Khalid, J.C-H. Peng. "Immunity toward data-injection attacks using multisensor track fusion-based model prediction." IEEE Transactions on Smart Grid 8, no. 2, pp. 697-707, 2015.
[39] H.M. Khalid, J.C-H. Peng. "A Bayesian algorithm to enhance the resilience of WAMS applications against cyber attacks." IEEE Transactions on Smart Grid 7, no. 4, pp. 2026-2037, 2016.
[40] J.E. Tate, and T.J. Overbye. "Line outage detection using phasor angle measurements." IEEE Transactions on Power Systems 23, no. 4, pp. 1644-1652, 2008.
[41] S. Soltan, P. Mittal, and H.V. Poor. "Line failure detection after a cyber-physical attack on the grid using Bayesian regression." IEEE Transactions on Power Systems 34, no. 5, pp. 3758-3768, 2019.
[42] W. Li,, D. Deka, M. Chertkov, and M. Wang. "Real-time faulted line localization and PMU placement in power systems through convolutional neural networks." IEEE Transactions on Power Systems 34, no. 6, pp. 4640-4651, 2019.
[43] K. Chen, J. Hu, Y. Zhang, Z. Yu, Jinliang He. "Fault location in power distribution systems via deep graph convolutional networks." IEEE Journal on Selected Areas in Communications 38(1), pp. 119-131, 2019.
[44] S. Brahma,, R. Kavasseri, H. Cao, N. R. Chaudhuri, T. Alexopoulos, and Y. Cui. "Real-time identification of dynamic events in power systems using PMU data, and potential applications—models, promises, and challenges." IEEE transactions on Power Delivery 32, no. 1, pp. 294-301, 2016.

Prithwiraj Roy received his PhD in Computer Science from Missouri University of Science and Technology, Rolla, USA in 2021, and BS in Electronics and Communication from Bengal Engineering and Science University in 2010. His current research interest includes smart grid security, artificial intelligence in cyber-physical systems, and information propagation in complex networks and he works as Data Scientist in Global Action Alliance Inc.

Shameek Bhattacharjee received his PhD and MS degrees from the University of Central Florida, Orlando, in 2015 and 2011, respectively and BS from the West Bengal Univ. of Tech, India, in 2009. He is currently an assistant professor with the Dept. of Computer Sc. at Western Michigan University. Between 2015-2018, he worked as a post-doctoral researcher with Missouri Univ. Sc & Tech. His current research interests include cybersecurity in cyber-physical systems, internet of things, and artificial intelligence, particularly in topics such as anomaly detection, trust models, dependable machine learning. He is a recipient of the Provost Fellowship and IEEE PIMRC Best Paper Award. He serves as a TPC member in various leading conferences such as IEEE ICDCS, IEEE SECON, IEEE/ACM Middleware, and a regular reviewer in leading journals such as IEEE Trans. on Mobile Computing, IEEE Trans. of Dependable and Secure Computing, IEEE J. of Sel. Areas in Commm.

Sahar Abedzadeh is currently a PhD student in the Department of Computer Science at Western Michigan University, Kalamazoo. Prior to this, she received her MS (in 2017)and BS degrees (in 2014) in Mathematics from K.N. Toosi University of Technology and Ferdowsi University of Mashhad in Iran. Here research interests include AI for cybersecurity, Cybersecurity Data Science, Applied Machine Learning.

Sajal K. Das is a professor of Computer Science and the Daniel St. Clair Endowed Chair at the Missouri Univ. of Sc. and Tech., where he was the Chair of Computer Science Dept. during 2013-2017. His research interests include cyber-physical systems, IoT, cybersecurity, pervasive and mobile computing, wireless sensor networks, and parallel computing, among others. He has made fundamental contributions to these areas and published extensively in high quality journals and peer-reviewed conference proceedings. He holds 5 US patents and coauthored 4 books, such as Handbook on Securing Cyber-Physical Critical Infrastructure: Foundations and Challenges, and Principles of Cyber-Physical Systems: An Interdisciplinary Approach. His h-index is 96 with more than 37,000 citations. He is a recipient of 12 Best Paper Awards at conferences like ACM MobiCom and IEEE PerCom, and numerous awards for teaching, mentoring and research including the IEEE Computer Society's Technical Achievement Award for pioneering contributions to sensor networks and mobile computing, and University of Missouri System President's Award for Sustained Career Excellence. Dr. Das serves as the founding Editor-in-Chief of Elsevier's Pervasive and Mobile Computing Journal, and as Associate Editor of the IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Mobile Computing, IEEE/ACM Transactions on Networking, and ACM Transactions on Sensor Networks. He is an IEEE Fellow.