Learning Behavior of Distribution System Discrete Control Devices for Cyber-Physical Security

Ciaran Roberts, Anna Scaglione, Mahdi Jamei, Reinhard Gentz, Sean Peisert, Emma M. Stewart, Chuck McParland, Alex McEachern, and Daniel Arnold

***Abstract*—Conventional cyber-security intrusion detection systems monitor network traffic for malicious activity and indications that an adversary has gained access to the system. The approach discussed here expands the idea of a traditional intrusion detection system within electrical power systems, specifically power distribution networks, by monitoring the physical behavior of the grid. This is achieved through the use of high-rate distribution Phasor Measurement Units (PMUs), alongside SCADA packets analysis, for the purpose of monitoring the behavior of discrete control devices. In this work we present a set of algorithms for passively learning the control logic of voltage regulators and switched capacitor banks. Upon detection of an abnormal operation, the operator is alerted and further action can be taken. The proposed learning algorithms are validated on both simulated data and on measured PMU data from a utility pilot deployment site.**

NOMENCLATURE

δ^i	Time series difference for event i
Δ	Summation of individual δ^i series
τ_i	Time delay for event i
η	Vector of residuals
Cap_State	Binary variable indicating whether capacitor bank is connected to the network
BW	Controller deadband bandwidth.
E	Set of all recorded events for a device.
I_{CT}	Primary rated current of current transformer.
Inv_Init	Binary variable indicating whether time delay has been initialized in case of inverse time delay
$\boldsymbol{i}^u/\boldsymbol{v}^u$	Vector of current/voltage measurements corresponding to voltage step-up operations
$\boldsymbol{i}^d/\boldsymbol{v}^d$	Vector of current/voltage measurements corresponding to voltage step-down operations
m	Total number of events for a particular device
N	Voltage deviation from V_{Target} normalized to $BW/2$
N_{PT}	Potential transformer ratio.
R_{Drop}	Voltage drop due to network resistance.
$\mathcal{S}$	Set of events for a device.
$\mathcal{S}_{down}$	Set of step-down events for a device.
$\mathcal{S}_{up}$	Set of step-up events for a device.
T_{user}	User set time delay.
T_{error}	Error of controller time delay
t_a^i	Time at which devices actuates for event i
t_c^i	Time at which variable crosses upper-/lower-threshold for event i
$t_{sampling}$	Sampling period of controller
t_w	Time window for events
t_{timer}	Countdown timer to device actuation
$t_{frac\,remain}$	Fraction of t_{timer} remaining when operating with an inverse time delay
V_{lower}	Voltage lower threshold.
V_{upper}	Voltage upper threshold.
V_{Target}	Regulator target voltage.
V_{set}	User set target voltage at zero loading conditions.
X_{Drop}	Voltage drop due to network reactance.
Z_{Drop}	Voltage drop due to network impedance.

C. Roberts (cmroberts@lbl.gov), D. Arnold, R. Gentz, S. Peisert and C. McParland are with Lawrence Berkeley National Laboratory.

A. Scaglione and M. Jamei are with Arizona State University.

E. M. Stewart is with Lawrence Livermore National Laboratory.

A. McEachern is with Power Standards Laboratory.

This research was supported in part by the Director, Cybersecurity, Energy Security, and Emergency Response, Cybersecurity for Energy Delivery Systems program, of the U.S. Department of Energy, under contract DE-AC02-05CH11231. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsors of this work.

I. INTRODUCTION

The increasing penetration of Distributed Energy Resources (DER) is transforming the role of the distribution grid in modern power networks. Historically, the distribution grid exhibited very well understood and predictable behavior due to slow time-varying aggregated demand profiles. This, however, is no longer universally the case with significant levels of distributed solar and electrical vehicles, among others, proliferating distribution networks, potentially causing large power variations and power quality problems. The rising level of variability within distribution networks will require a more active management of the network via an Advanced Distribution Management System (ADMS), with two-way communication, greater levels of automation and Fault Location, Isolation, & Service Restoration (FLISR) capabilities. This increased level of automation and communication, however, opens potential entry points for adversaries seeking to disrupt operations.

In order to minimize the risk of cyber-attacks, a holistic cyber-physical security approach, marrying both the physical operating behavior of the grid as well as the superimposed communication/control layer, is required. The use of physical voltage and current measurements alongside SCADA data has been explored for the transmission grid for detecting false data attacks and subsequent malicious mis-operation of protection equipment [1], [2], for detecting false data attacks

against automatic generation control [3] and for detecting abnormal voltage and/or current measurements [4], [5]. At the distribution level, a similar approach was adapted in [6], [7] where the authors were interested in unauthorized control actions of DER. This work differs in that it focuses on regulation equipment and adapts a model-based approach, similar to [3], which typically requires less training data than machine learning approaches. This reduces the re-learning period following an intentional change in control settings.

This work seeks to utilize distribution Phasor Measurement Units (PMUs), which are receiving increasing attention in recent literature [8]–[10], as an independent isolated read-only sensor network to complement existing intrusion detection systems (IDSs), that focus on monitoring SCADA traffic [11]–[13]. We use PMUs to monitor the physical behavior of control devices on the distribution gird, namely On-Load Tap Changing (OLTC) transformers and capacitor banks. Given a sufficient density of these control devices, an adversary could use them to force the voltage into the voltage ride-through range of DER [14]. This can subsequently cause the tripping of a large population of DER and/or affect customer power quality and target specific sensitive loads.

Additionally, our work is motivated by seeking to detect adversaries within a SCADA network prior to the initiation of an attack, namely when they may be carrying out *reconnaissance*. *Reconnaissance* is where an adversary would gain information about a network or verify controllability, but their action(s) would not trigger an alarm or alert, and would likely go unnoticed by the operator. In a distribution grid, an example could include altering the static time delay or deadband bandwidth of an On-Load Tap Changing (OLTC) transformer, for the case where it is remotely configurable, to confirm controllability. In the case of the Ukrainian cyber-attack, it was noted that the adversaries likely gained access to the network up to six months prior to the attack and conducted reconnaissance during this period [15]. This work seeks to detect abnormal operation of these control devices that may go unnoticed by a system operator, particularly if the SCADA data from the device in question is being spoofed. Once these abnormalities have been detected, the operator would be notified that there may be an adversary in their SCADA network.

The primary contribution of this work is a set of algorithms to passively learn and monitor the control logic of distribution system regulation equipment, specifically OLTC transformers and capacitor banks. We first detect and assign control actions to specific control devices. Following this, a coarse estimation of the time delay of the device, static or dynamic, is estimated. Then we optimize locally around this coarse estimation of the time delay to more accurately estimate the delay of the device and the upper- and lower-thresholds, outside of which the device actuates, are estimated. Finally, a threshold for classifying operations as normal or abnormal is proposed.

The paper is organized as follows; Section II describes the general architecture for monitoring discrete switching devices from a Cyber-Physical security standpoint, Section III presents an overview of the possible control schemes that regulators and capacitor banks can operate under, Section IV describes the proposed methodology for identifying and parametrizing their control logic, Section V validates this proposed methodology through both simulation and data from a utility network and empirically analyzes its performance and Section VI concludes with a summary and potential future work.

Notation: Throughout the paper, unless otherwise noted, all variables are real. Bold lower case symbols are used to denote vectors, bold upper-case symbols are used to denote matrices and a hat over a parameter corresponds to its estimated value.

II. Cross-Corroboration with SCADA

As described in Section I, one of the primary motivations for this work is the use of an independent, isolated sensor network to corroborate SCADA, specifically regarding the behavior of discrete control devices. In order to monitor SCADA we deploy the Bro network monitoring tool [16] that passively listens to network traffic and sends reports to our monitoring framework, where these reports are correlated and further analyzed with corresponding PMU data. This fused analysis is beneficial for us because it allows the analysis of "broad data" (from large amounts of SCADA enabled devices) with "high detail data" (from high sample rate PMU data). Furthermore, the additional cross-checking helps to validate whether any of the devices are reporting false or otherwise incorrect data, e.g. is an adversary spoofing data to hide abnormal behavior from a control device. Fig. 1 outlines the process of passively monitoring control devices and alerting operators upon the detection of abnormal operation.

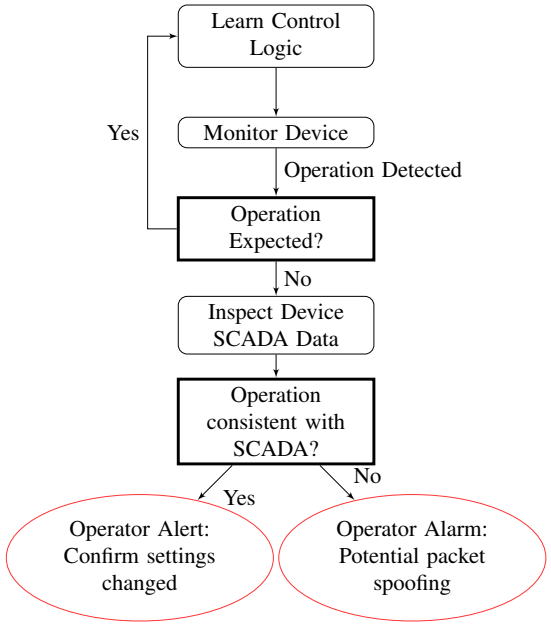

Fig. 1. Logic for detecting reconnaissance attacks.

Of particular focus for this work will be the activity of learning the control logic of these devices. While in principle the behavior of each component on the grid should be known to the operators, it is desirable to minimize the amount of a-priori knowledge of the cyber-physical environment and configuration that is assumed in the analysis. Learning it from

the measurements themselves makes the architecture read-only, isolating the cyber-physical data analytics from outdated and incorrect data entry as well as manipulation. Our work proves that, abstracting the functionality of devices in a certain class and extracting its parameters, one can learn from PMU data the devices normal configuration, having a baseline for the detection of anomalies. The learning approach is based on the devices expected behavior. This is preferable to a black box approach, which requires significant more training data and it is not amenable to interpretation. Section IV outlines the methodology for learning this logic and concludes with a subsection discussing how to differentiate with normal and abnormal behavior.

III. An Overview of Control Logic Schemes

Within this work we focus on discrete regulation equipment. A similar approach can be adapted for DER with Volt/Var and/or Volt/Watt functionality; however, this is beyond the scope of this work. In this section we review the control logic of voltage regulators and capacitor banks, which constitute the targets of our anomaly detection algorithms.

A. Voltage Regulators

Operational voltage regulation control schemes can be generally characterized by two distinct properties; 1) their upper- and lower-voltage thresholds and 2) their user-defined time delay. In order for our approach to be generalizable, it must be able to both identify the specific implementation and parameterize the corresponding control scheme. Given this, a brief description of the modes of operation are outlined below.

1) Voltage Threshold Boundaries: Voltage regulators typically operate with upper and lower voltage thresholds, outside of which they will actuate. These thresholds may be static, which means that, irrespective of loading conditions, the upper and lower thresholds delimit a fixed range within which the regulator tries to maintain the voltage. Another mode of operation is where a regulator estimates, and regulates, the voltage at a certain bus corresponding to a customer load center, as a linear function of i) local measurements at the secondary terminals of the transformer, namely current magnitude and power factor, and ii) its respective network properties [17]–[20]. This mode of operation is called Line Drop Compensation (LDC). In LDC operation the difference between the customer voltage and the voltage measured at the regulator terminals is attributable to a voltage drop along lines and other transformers. A graphical representative of the static threshold and LDC is shown in Fig. 2, where BW is the deadband bandwidth of the regulator. It is important to note that a regulator may use measurements that are not at the regulator site, but from another node in the network. For our algorithms we assume that we have access to the same measurement values that the regulator controller is acting upon. This can achieved either by direct measurement at the same node as the controller or by using nearby sensors to estimate the controller measurements, assuming sufficient observability. In the case of the latter, errors in line impedance and/or systematic errors in the respective instrument transfers

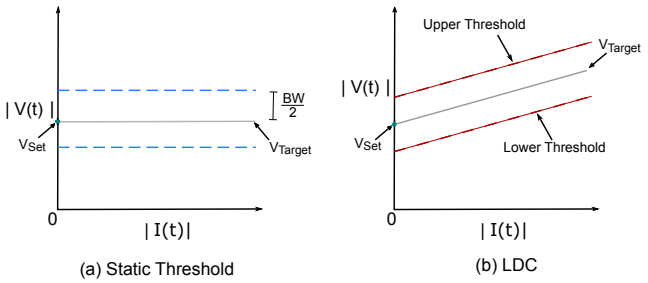

Fig. 2. Static Threshold and LDC Voltage Thresholds.

may adversely impact the ability to accurately estimate the controller measurements. Particular care must be taken to appropriately account for these sources of error [21].

When operating in LDC mode, a regulator may employ LDC-R&X or LDC-Z. The LDC-R&X scheme determines its target voltage as follows. Consider a line with resistance R_L and reactance X_L with a load at its end. R_{Drop} and X_{Drop} (whose units are volts) are the expected voltage drops due to the line characteristics, calculated as follows:

$$\begin{aligned} R_{Drop} &= \frac{I_{CT}}{N_{PT}} R_L \\ X_{Drop} &= \frac{I_{CT}}{N_{PT}} X_L \end{aligned} \tag{1}$$

where I_{CT} is the primary rated current of the current transformer and N_{PT} is the potential transformer ratio. Let $pf(t)$ denote the power factor and $qf(t)$ denote the load reactive power factor, where $qf(t) = \sin(\arccos(pf(t)))$ [17], measured at the secondary terminals of the transformer. The target voltage for the LDC-R&X scheme is then given by:

$$V_{Target}(t) = V_{set} + (pf(t)R_{Drop} + qf(t)X_{Drop})\frac{I(t)}{I_{CT}} \tag{2}$$

Within this work we assume that we have available measurements of both voltage and current phasors, from which we can obtain the necessary magnitude quantities and power factor.

The LDC-Z method is implemented in scenarios where there is no dominant single line, or path of lines, from which a suitable R_L and X_L can be determined. The LDC-Z control logic is a simple linear relationship between the measured voltage and current magnitudes, as shown in (3):

$$V_{Target}(t) = V_{set} + Z_{Drop}\frac{I(t)}{I_{CT}} \tag{3}$$

where Z_{Drop}, whose units is in volts, is some expected voltage drop at rated current, estimated analytically and/or through simulation.

The upper- and lower-thresholds, outside of which the regulator will actuate, are then given by the following (4) for both the LDC-R&X and LDC-Z:

$$\begin{aligned} V_{upper}(t) &= V_{Target}(t) + BW/2 \\ V_{lower}(t) &= V_{Target}(t) - BW/2 \end{aligned} \tag{4}$$

The parameters to be estimated for the case of static thresholds are simply V_{upper} and V_{lower} while both LDC-Z and LDC-R&X require estimating V_{upper} and V_{lower} at zero loading conditions, and additionally, R_{Drop} and X_{Drop} for LDC-R&X

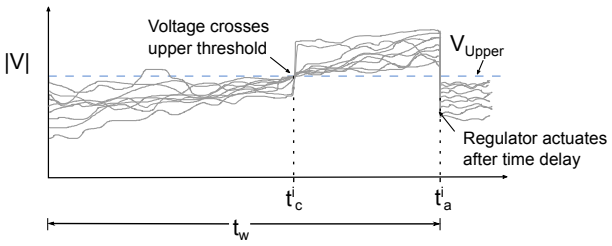

Fig. 3. Defining t_a^i, t_c^i and t_w for fictitious time series voltage profiles.

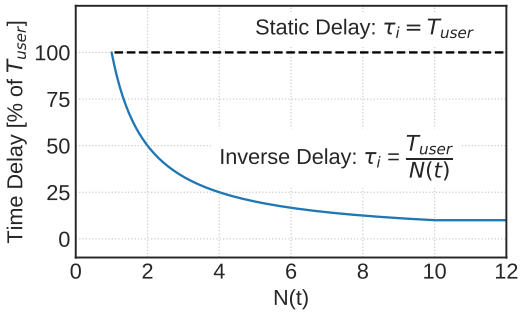

Fig. 4. Static and inverse regulator time delay.

and Z_{Drop} for LDC-Z. Throughout the paper we refer to the time instant immediately before the actuation of a switching event i, from either a regulator or switched capacitor bank, as t_a^i, the time at which its voltage crosses its upper-/lower-threshold as t_c^i and the time duration of our event sample as t_w. These are graphically shown in Fig. 3 for some fictitious voltage time series profiles. The controller time delay associated with this switching action is then given as $\tau_i = t_a^i - t_c^i$. Estimating the threshold that the voltage magnitude crossed requires detecting the times of control device actuation, t_a^i, and finding the crossing time, t_c^i, which requires identifying the controller time delay.

2) Controller Time Delay: Typically, once a regulator has determined that the measured voltage has exceeded an allowable range, it begins a countdown timer before taking an action aimed at restoring the voltage. If the voltage re-enters the allowable range prior to this timer reaching zero, the timer is reset, and no action is taken. Otherwise, the regulator will execute a tap change operation.

This controller time delay can be a constant set by the user, T_{user}, or it can be varied; the general indication is that the delay is inversely proportional to the ratio of the voltage deviation relative to the deadband BW, as shown in Fig. 4 where N(t) is defined as:

$$N(t) = \frac{2|V(t) - V_{Target}|}{BW}. \tag{5}$$

The control logic, which is executed once per controller sampling period, for a regulator operating with a static time delay is trivial. Once the voltage exits the allowable range a countdown timer is initiated. If the voltage does not re-enter the allowable range before the countdown timer has elapsed, a control action is taken. The logic for an inverse time delay, however, is not as straight forward. Although the specific implementation of the inverse time delay can differ across controllers, following consultation with vendors an example implementation of an inverse time delay is outlined in Algorithm 1 where $t_{sampling}$ is the controller sampling period, t_{timer} is the remaining time left on the countdown timer in seconds, $t_{\text{frac remain}}$ is the fraction of time remaining and Inv_Init is a binary variable indicating whether the time delay has been initialized, Within this work we assume that $N(t)$ settles at a quasi-steady state value following a discrete jump in the voltage timer series and it is this value for $N(t)$ which will be used for estimation purposes. In that case, the controller time delay τ_i associated with the ith event, from when the voltage magnitude first crossed the threshold to the switching event, is therefore:

$$\tau_i = \mathrm{T}_{user}/N(t) \tag{6}$$

Upon detection of a voltage excursion, regulators may determine the required tap position to restore the voltage within allowable bounds and do so in one tap movement or more gradually, by multiple tap movements [17]. In the case of the latter, and when operating under a definite time delay, the regulator will typically have an *inter-tap time delay* which determines the time delay between subsequent tap executions once an initial operation has taken place. Initially we will filter out possible tap-operations under the influence of an inter-tap by only considering events sufficiently separated in time, e.g. events must be separated by at least 2 minutes. If the regulator operated with an inter-tap delay, this can easily be learned online.

Algorithm 1: Voltage regulator logic for stepping down the voltage with inverse time delay

```
1  V_upper(t) and V_lower(t) are given by (3) and (4);
2  if V(t) > V_upper(t) then
3      if Inv_Init=0 then
4          Compute N(t) using (5) ;
5          Initialize t_timer = τ_i using (6) ;
6          Inv_Init = 1
7      else
8          t_timer = t_timer − t_sampling ;
9          if t_timer ≤ 0 then
10             Step-down the voltage;
11             Inv_Init = 0 ;
12         else
13             t_frac remain = t_timer/τ_i ;
14             Compute N(t) using (5) ;
15             Compute τ_i using (6) ;
16             t_timer = t_frac remain × τ_i ;
17 else if V(t) < V_lower(t) then
18     Mirrors logic in lines 3-16
19 else
20     Inv_Init=0 ;
```

B. Switched Capacitor Banks

Switched capacitor banks can be operated under a number of possible modes, and in some cases can operate under multiple modes simultaneously, provided there is an associated priority list [22], [23]. The most common variables upon a switched capacitor bank can be programmed to operate under include:

- Measured voltage
- Computed VAR demand
- Measured Current
- Temperature (to compensate for inductive AC loads)
- Time schedule (based on time of day)

Each individual mode of operation is less complex in comparison to a regulator, with a simple upper and lower threshold specified[1] and an associated time delay. In the case of switched capacitor banks, however, there can be an additional internal time delay in the operation logic. This additional *safety switching delay* begins counting down once the user-defined time delay has elapsed. This safety switching delay allows the operators to move away from the capacitor bank before switching in/out, operating in either manual or autonomous mode, given that the potential for fault is higher during a switching action. This safety switching delay can be the same for switching in/out or can be different. The capacitor bank will perform a switching action regardless of whether or not the quantity of interest re-enters its allowable range once the *safety switching delay* countdown timer has been initiated, unless this action is manually overridden by a physical button on the controller interface. Algorithm 2 describes the logic of a capacitor bank controller whereby Cap_State is a binary variable indicating whether the capacitor bank is connected the network. The logic outlined in Algorithm 2 should be executed once per timestep.

Algorithm 2: Capacitor Bank Switching Logic for Voltage

```
if t_timer > 0 then
    if V(t) > V_upper and Cap_State=1 then
        t_timer = t_timer − t_sampling ;
    else if V(t) < V_lower and Cap_State=0 then
        t_timer = t_timer − t_sampling ;
    else
        t_timer = T_user ;
else if t_timer ≤ 0 then
    t_safety = t_safety − t_sampling ;
    if t_safety ≤ 0 then
        Capacitor bank switching performed
        Reset t_safety
        t_timer = T_user
```

IV. Methodology for Learning Device Behavior

A. Measurement Requirements and PMU Placement

In order to learn the behavior of individual devices, discrete control actions must first be detected in distribution PMU data and then attributed to individual devices on the network. Detection can be a simple threshold algorithm determined by the physical properties of the controller, e.g. distribution OLTC transforms typically regulate voltage $\pm 10\%$ with ± 16 tap positions. Therefore each tap change changes the voltage by 0.625% with an actuation time period of 40-60 ms [24]. Given that the number of control devices are typically orders of magnitude less than the number of nodes within a distribution network this can be achieved with a minimal number of PMUs. We assume that each PMU provides phasor measurements of both the voltage and current, i.e. magnitude and angle, from the same point at which the controller of interest is measuring. As noted in Section III-A this can be done through direct measurement or by estimation using nearby sensors, assuming sufficient observability and a thorough treatment of impedance and/or instrument transformer error [21]. From these measurements we have both voltage and current magnitudes and can compute both pf and qf from the angle values. Using the relative topological information of these PMUs and the control devices of interest, step changes in voltage, and reactive power in the case of switched capacitor banks, can be used to detect and classify discrete control actions [25]. An example of a placement at a pilot site deployment is shown in Fig. 5 where PMU 1 and PMU 2 are used in conjunction to determine whether a step change in voltage magnitude was a local action of OLTC 1 or OLTC 2 respectively or whether it was an event on the sub-transmission grid. An example of both a local event recorded on PMU and a global event, observed in both PMU 1 and PMU 2 is shown in Fig. 6. It is the high accuracy GPS synced time-stamped measurements that are exploited when attributing control actions to devices. For the example shown in Fig. 6 an event would only be misclassified as global if there was an event on the subtransmission network 1) that caused the voltage that each respective device is regulating to cross its upper- or lower-threshold at the same time instant, and 2) the controllers for both OLTC 1 and OLTC 2 have the same user defined delay and 3) the time delay error of each controller is sufficiently small that it is undetectable by the measurements. Although these conditions are possible, they becoming increasingly unlikely as the number of measurements for cross-comparison from parallel feeders connected through the same subtransmsission feeder increases.

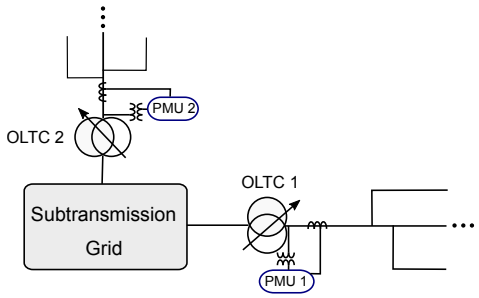

Fig. 5. Monitoring device behavior with PMUs.

[1]The voltage mode can also use a second threshold, an extreme threshold, for which the user set time delay is zero.

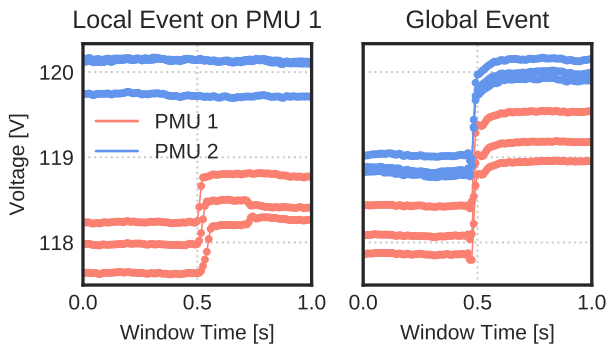

Fig. 6. Example of a recorded *local* and *global event* for all 3 phases.

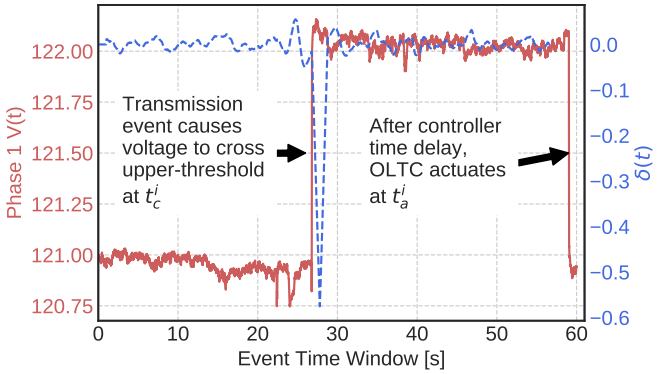

Fig. 7. Detected event from pilot deployment site and its $\boldsymbol{\delta}$ time series profile for n=1s.

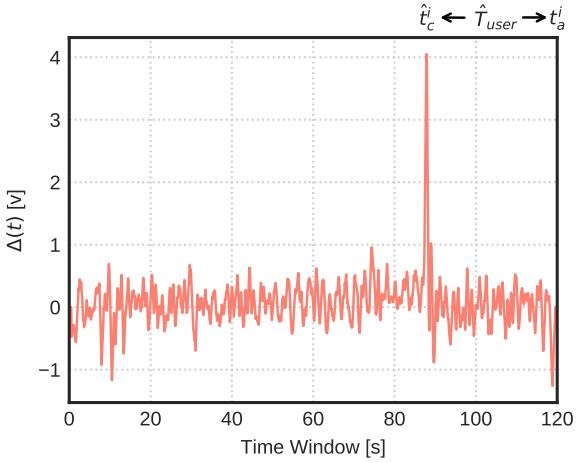

Fig. 8. Example of a $\boldsymbol{\Delta}$ time series, given by (9), from a pilot site deployment.

B. Learning Regulator Control

The learning of regulator control logic can be broken up into two steps: 1) learning the associated time delay and 2) learning the upper- and lower-threshold boundaries. There are two causes for the voltage to exit the deadband of a regulator or switched capacitor bank. These are 1) a ramp up/down in net-load that pushes the voltage outside its allowable range in a continuous manner, or 2) a discrete jump in the voltage caused by either a large change in net-load load or an event on the transmission/subtransmission grid, e.g. a switching action or the actuation of subtransmission regulation equipment. It is the latter, case 2, which we seek to exploit in order to obtain an initial coarse estimate of T_{user}.

For the case of ramps in the net-load we can expect that normally $V(t) \approx V(t-1)$ for measurements one second apart. However, when a discrete jump in the voltage profile occurs this will not hold true $\forall t$ and we will seek to estimate the time, t^i, for which $V(t^i) \not\approx V(t^i+1)$ for some events i. Therefore we will estimate $\hat{t}_c^i$ by detecting this discontinuity with an approach similar to the Early-Late Gate timing recovery block in signal processing [26] where for each event $i \in \mathcal{S}$ and averaging period n, its time series $\boldsymbol{\delta^i}$ is given by (7):

$$\boldsymbol{\delta^i}(t) = V^i(t_-) - V^i(t_+) \quad \forall t \in [t_a^i - t_w + n, t_a^i - n] \quad (7)$$

where

$$V^i(t_-) = \frac{1}{n}\sum_{j=t-n}^{t} V(j) \qquad V^i(t_+) = \frac{1}{n}\sum_{j=t}^{t+n} V(j) \quad (8)$$

This averaging will attenuate high frequency voltage fluctuations due to motor starts and/or switching in of large inverter interfaced resources. A sample field-recorded voltage profile and its corresponding $\boldsymbol{\delta}$ time series can be seen in Fig. 7.

We then sum over the m individual $\boldsymbol{\delta}$ time series vectors to obtain a new time series, $\boldsymbol{\Delta}$:

$$\boldsymbol{\Delta} = \sum_{i \in \mathcal{S}_{up}} \delta^i - \sum_{i \in \mathcal{S}_{down}} \delta^i \quad (9)$$

where the negative sign accounts for the fact that $\boldsymbol{\delta}$ time series for voltage step-down events will exhibit a large negative peak, as shown in Fig. 7, while $\boldsymbol{\delta}$ time series for voltage step-up events will exhibit a large positive peaks. For the case of a static time delay, i.e. $t_a^1 - \hat{t}_c^1 \approx \ldots \approx t_a^n - \hat{t}_c^n$, this summation will amplify individual δ^i values around $\hat{t}_c^i$, i.e. when all voltage time series are increasing as they cross their respective upper-threshold or decreasing as they cross their respective lower-threshold, and attenuate all other values due to the stochastic nature of voltage fluctuations. An example of a $\boldsymbol{\Delta}$ time series from a pilot deployment site is shown in Fig. 8. The time series profile exhibits a distinct singular peak close to 90s, where the end of the window corresponds to t_a^i, and consequently this particular OLTC has a static time delay, $\hat{T}_{user}$, close to 30s. This distinct peak is due to the summation of individual peaks, similar to those in Fig. 7, given by (9).

In the event of a distinct singular peak in the $\boldsymbol{\Delta}$ time series profile, as is the case in Fig. 8, we can conclude that the regulator is operating under a static time delay. For the case of an inverse-time delay, we can expect multiple peaks in $\boldsymbol{\Delta}$ and this case will be addressed in Section IV-B2. First, however, we must introduce the methodology for estimating the upper- and lower-thresholds.

1) Parameterizing a regulator with a static time delay: For a regulator operating with a static time delay, a coarse estimate of the time at which the voltage exited its allowable range, $\hat{t}_c^i$, is given by $t_a^i - \hat{\mathrm{T}}_{user}$. This is a coarse estimation of t_c^i as the error of the time delay on a regulator can vary. In

order to robustify the estimation process against measurement noise, a measurement at time t is taken to be the mean of a 10 cycle window centered at t. Given a set of voltage step-up operations, S_{up} of length p we define the ordered vector $\boldsymbol{v}^u(t^i)$ to be $[V^1(t^i),...,V^p(t^n)]^T$, where $V^1(t^i)$ denotes the voltage at some time $t^i \in [t_a^1 - t_w, t_a^1]$ for event $1 \in S_{up}$. A similar vector is defined for the current, $\boldsymbol{i}^u(t^i)$, and for both the voltage and current, $\boldsymbol{v}^d(t^i)$ and $\boldsymbol{i}^d(t^i)$ respectively, in the set of voltage step-down operations, S_{down}.

For the case of LDC-Z the equations describing the control logic are given by:

$$\begin{aligned} V_{upper}(t_c^i) &= V_{set} + \frac{BW}{2} + \frac{I(t_c^i)}{I_{CT}} Z_{Drop} \\ &= \hat{v}_{upper} + I(t_c^i)\hat{z}_{Drop} \\ V_{lower}(t_c^i) &= V_{set} - \frac{BW}{2} + \frac{I(t_c^i)}{I_{CT}} Z_{Drop} \\ &= \hat{v}_{lower} + I(t_c^i)\hat{z}_{Drop} \end{aligned} \tag{10}$$

where $\hat{v}_{upper} = V_{set} + BW/2$, $\hat{v}_{lower} = V_{set} - BW/2$ and $\hat{z}_{Drop} = Z_{Drop}/I_{CT}$. For LDC-Z, the optimization is given by:

$$\begin{aligned} \underset{x,t^i}{\text{minimize}} \quad & ||\boldsymbol{y}(t^i) - \boldsymbol{A}(t^i)\boldsymbol{x}||_2^2 \\ \text{subj. to} \quad & |\hat{t}_c^i - t^i| \le T_{error} \quad \forall i \\ & \hat{t}_c^i - t^i = \alpha \quad \forall i \end{aligned} \tag{11}$$

where: $\boldsymbol{y}(t^i) = [\boldsymbol{v}^u(t^i), \boldsymbol{v}^d(t^i)]^T$, $\boldsymbol{x} = [\hat{z}_{Drop}, \hat{v}_{upper}, \hat{v}_{lower}]^T$

$$\boldsymbol{A}(t^i) = \begin{bmatrix} \boldsymbol{i}^u(t^i) & \mathbf{1} & \mathbf{0} \\ \boldsymbol{i}^d(t^i) & \mathbf{0} & \mathbf{1} \end{bmatrix},$$

where $A(t^i)$ is an $m \times 3$ matrix, $y(t^i)$ is a vector of length m, T_{error} is the specified accuracy of the regulator time delay and $\mathbf{1}$ and $\mathbf{0}$ are vectors of 1's and 0's respectively. The MI-QP optimization in (11) can be reduced to a single-variable optimization by exploiting the closed form solution for x as a function of t^i, as given by (12). The first constraint in (11) allows the optimization to sweep over the neighborhood of $\hat{t}_i^c$ in order to compensate for the inaccuracy in the internal time delay of the OLTC controller. This allows us to refine our coarse estimation of $\hat{t}_c^i$ from (7) - (9). The second constraint enforces that a uniform offset from $\hat{t}_c^i$ across all events in order to ensure that the optimization remains computationally tractable. Given that it is only being evaluated over a small discretized time period, with each evaluation having a closed form solution given by (12), the problem remains computationally light.

$$\boldsymbol{x}_{opt}(t^i) = (\boldsymbol{A}(t^i)^T\boldsymbol{A}(t^i))^{-1}\boldsymbol{A}(t^i)^T\boldsymbol{y}(t^i). \tag{12}$$

Assuming that the controller operates on moving averaged measurements [18], with measurement errors that are Gaussian i.i.d. random variables, we can consider the controller to be acting upon the true value. The error, ϵ, in the model

$$\boldsymbol{y}(t^i) = \boldsymbol{A}(t^i)\boldsymbol{x} + \boldsymbol{\epsilon} \tag{13}$$

is then given by the variance of the voltage during the period $\hat{t}_i^c \pm T_{error}$, where T_{error} is the error of the controller time delay. Thus we can express the error as

$$\epsilon \sim \mathcal{N}(0, \sigma^2\boldsymbol{I}) \quad \sigma^2 = f(Var(V(t)), \mathrm{T}_{error}) \tag{14}$$

and the covariance of the estimated parameter, $\hat{x}_i$, is given by

$$Var(\hat{x}_i) = \sigma^2(A^TA)^{-1}_{i,i} \tag{15}$$

Consequently we have that for controllers with large time delay inaccuracies and/or sites with large voltage volatility, we require a larger library of events to converge to the true value of x.

In order to estimate parameters under the hypothesis that LDC-R&X is in operation, we define the additional ordered vector for voltage step-up operations $\boldsymbol{i}_{pf}^u(t^i)$, of length p, as $[I_{pf}^1(t^i),...,I_{pf}^p(t^i)]^T$ where

$$I_{pf}^1(t^i) = pf^1(t^i)I^1(t^i) \tag{16}$$

and similar vectors for $\boldsymbol{i}_{qf}^u(t^i)$,$\boldsymbol{i}_{pf}^d(t^i)$ and $\boldsymbol{i}_{qf}^d(t^i)$ Given that, we can now perform the following optimization:

$$\begin{aligned} \underset{x,t^i}{\text{minimize}} \quad & ||\boldsymbol{y}(t^i) - \boldsymbol{A}(t^i)\boldsymbol{x}||_2^2 \\ \text{subj. to} \quad & |\hat{t}_c^i - t^i| \le T_{error} \quad \forall i \\ & \hat{t}_c^i - t^i = \alpha \quad \forall i \end{aligned} \tag{17}$$

where: $\boldsymbol{y}(t^i) = [\boldsymbol{v}^u(t^i), \boldsymbol{v}^d(t^i)]^T$,
$\boldsymbol{x} = [\hat{r}_{Drop}, \hat{x}_{Drop}, \hat{v}_{upper}, \hat{v}_{lower}]^T$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{i}_{pf}^u(t^i) & \boldsymbol{i}_{qf}^u(t^i) & \mathbf{1} & \mathbf{0} \\ \boldsymbol{i}_{pf}^d(t^i) & \boldsymbol{i}_{qf}^d(t^i) & \mathbf{0} & \mathbf{1} \end{bmatrix}$$

where $\boldsymbol{A(t^i)}$ is an $m \times 4$ matrix, $\boldsymbol{y(t^i)}$ is a vector of length m and the optimal estimate of $\boldsymbol{x(t^i)}$ is given by (12). Determining whether LDC-Z or LDC-R&X is in operation is carried out by comparing the RMSE value from (11) and (17). The case of static thresholds is simply LDC-Z or LDC-R&X with $\hat{z}_{Drop} \approx 0$ and $\hat{r}_{Drop} \approx \hat{x}_{Drop} \approx 0$ respectively. The optimization in (11) and (17) should only be preformed for a set of events where all the operations occurred under conventional power flow direction or where all the operations occurred under reverse power flow conditions, due to these modes of operation not necessarily having the same LDC settings.

2) Parameterizing a regulator with an inverse time delay: In the event that there exist multiple positive peaks in $\boldsymbol{\Delta}$, then it is possible that the regulator is operating under an inverse time delay. In that case we need an initial estimate of the upper- and lower-voltage thresholds to parameterize the inverse time delay function. We perform this initial estimation for x at t_a^i using (12). Given that once a voltage time series crosses its threshold, it must remain outside the allowable range in order for that crossing to result in a tap-operation (if it did not, the timer would reset upon re-entering the allowable range and it would have to exit this range again and consequently have a new $\hat{t}_c^i$). Thus we have that for all voltage step-down events $V(t_a^i) > V(\hat{t}_c^i)$ and voltage for all

step-up events $V(t_a^i) < V(\hat{t}_c^i)$. Therefore, estimation of the parameters at t_a^i will result in upper- and lower-bounds on the upper-threshold and lower-threshold respectively. Once an estimate of the upper- and lower-thresholds, and consequently the target voltage and bandwidth as given by (19), has been obtained we can estimate the user defined time delay. We do so be detecting large jumps in the individual voltage time series profiles by (7). Given k time series profiles with a discrete jump in their voltage time series profile at time $\hat{t}_c^i$, we define τ_i as $t_a^i - \hat{t}_c^i$ and estimate the user specified delay by re-arranging (6) as:

$$\mathrm{T}_{user} = \frac{1}{k}\sum_{i=1}^{k} \frac{2\tau^i |V(\hat{t}_c^i) - V_{Target}(\hat{t}_c^i)|}{BW} \tag{18}$$

where $V_{Target}(\hat{t}_c^i)$ is given by (10) and BW is computed by:

$$BW = \hat{v}_{upper} - \hat{v}_{lower} \tag{19}$$

Given that such estimation of the bandwidth will be an overestimation, due to performing the optimization at t_a^i and the relationship between the bandwidth and thresholds given by (19), we will tend to underestimate T_{user}. We can however go through an refinement algorithm in order to obtain a more accurate estimation. This is summarized in Algorithm 3. Due to inaccuracies in the operational inverse time delay, $\pm 10\%$ [18], we take a measurement at time t to be the mean of a one second measurement window centered oat time t when parameterizing a regulator with an inverse time delay.

Algorithm 3: Parameterizing an OLTC under operation of inverse time delay

1 Initialize estimation of parameters at t_a^i using (12)
2 Compute $V_{target}(\hat{t}_c^i)$ and BW using (10) and (19)
3 Detect discrete voltage jumps in time series profiles with (7)
4 Using subset of events with discrete jumps at $\hat{t}_c^i$, define τ_i as $t_a^i - \hat{t}_c^i$ and estimate T_{user} using (18)
5 Compute $\hat{t}_c^i$ for each individual profile as follows,

$$\hat{t}_c^i = t_a^i - \mathrm{T}_{user} \times \min\left(1, \frac{BW/2}{\max_{t^i}(|V(t^i) - V_{Target}(\hat{t}_c^i)|)}\right)$$

6 Perform the optimization outlined in (11) at $\hat{t}_c^i$
7 Compute V_{target} and BW using (19)
8 Repeat *Steps 4 to 7* while $|T_{user} - T_{user}^{prior}| > \epsilon$

C. Learning Switched-Capacitor Control

The task of inferring the control logic of switched-capacitor banks is similar to that of regulators, but with an additional step. Given that a switched capacitor bank can operate under multiple control modes simultaneously, this additional step is necessary for determining which mode was responsible for each individually recorded switching action. Once events have been attributed to their respective modes, the parameterization of the control logic of these modes can be carried out. The following approach does not apply to switched capacitors on a time schedule. Due to the deterministic nature of time-scheduled switching relative to the stochastic time varying of the other triggering features (e.g. voltage and Var demand), identifying a switched capacitor under the operation of a time-schedule is a trivial exercise. For the remainder of the modes, classifying events with respect to their triggering mode is carried out first constructing the matrix, Ψ, shown in (20):

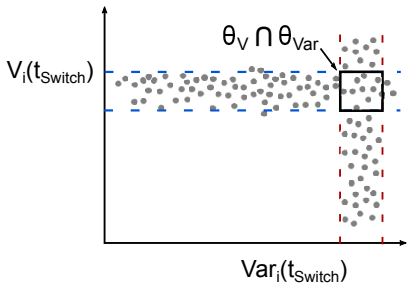

Fig. 9. Example of control triggering modes identified using subspace clustering on feature matrix in (20).

$$\Psi = \begin{bmatrix} V_1(t_a^1) & Var_1(t_a^1) & I_1(t_a^1) \\ V_2(t_a^2) & Var_2(t_a^2) & I_2(t_a^2) \\ \vdots & \vdots & \vdots \\ V_i(t_a^i) & Var_i(t_a^i) & I_i(t_a^i) \end{bmatrix} \tag{20}$$

and then performing subspace clustering on this matrix [27]. Subspace clustering, as employed here, is the task of determining axis-parallel affine subspaces. If we denote these affine spaces as Θ_γ, where γ denotes the feature variable of interest, e.g. voltage, we can then attribute a subset of the events contained within these affine spaces to the corresponding triggering mode. In fact, the subset of the events associated to mode γ is given by:

$$Events_\gamma = \Theta_\gamma \backslash (\cap_{i=1}^{n} \Theta_i)\,. \tag{21}$$

A graphical representation of classifying events for the case of a two-dimensional feature space is shown in Fig. 9.

Once events have been assigned to their respective modes, the thresholds can be determined in a similar manner to the regulator, whereby the time delay for voltage switching is estimated by peak detection in $\boldsymbol{\Delta}$ time series. As noted in Section III, given that a switching in/out action may have a different safety switching delay, there may have different time delays, given by $T_{user} + t_{safety}$. Therefore, peak detection needs to be performed on $\boldsymbol{\Delta}_{up}$ and $\boldsymbol{\Delta}_{down}$ as given by (22) and the threshold boundaries are estimated by (23). A similar approach is adapted for current and var mode.

$$\begin{aligned} \boldsymbol{\Delta}_{up} &= \sum_{i \in S_{up}} \delta^i \\ \boldsymbol{\Delta}_{down} &= -\sum_{j \in S_{down}} \delta^j \end{aligned} \tag{22}$$

$$\begin{aligned} \underset{x,t^i}{\text{minimize}} \quad & ||\boldsymbol{y}(t^i) - \boldsymbol{A}\boldsymbol{x}||_2^2 \\ \text{subj. to} \quad & |\hat{t}_c^i - t^i| \le T_{error} \quad \forall i \\ & \hat{t}_c^i - t^i = \alpha \quad \forall i \end{aligned} \tag{23}$$

where: $\boldsymbol{y} = [\boldsymbol{v}^u(t^i), \boldsymbol{v}^d(t^i)]^T$, $\boldsymbol{x} = [\hat{v}_{upp}, \hat{v}_{low}]^T$

$$\boldsymbol{A} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix},$$

D. Monitoring Control Devices

In the event that an operation has been detected in PMU data that is inconsistent with a device's historical behavior, the corresponding SCADA communications will be inspected. A suitable method for classifying an operation as abnormal or unexpected is a threshold approach. We denote our vector of residuals from performing the optimization given by (11) or (17) as $\boldsymbol{\eta}$. Then, we can set the threshold for the voltage as 3σ of $\boldsymbol{\eta}$, where σ denotes the standard deviation of the vector.

V. Application to Test Cases

A. Simulated Control Schemes

For simulating different control schemes we replay field recorded voltage and current phasor measurements through a simulated controller whose properties mirror that of [18], specifically a static time delay error of $\pm 0.5\%$ and inverse time delay of $\pm 10\%$. There were 3 control schemes implemented:

1) OLTC with static thresholds and a fixed time delay
2) OLTC with LDC-Z and a fixed time delay
3) OLTC with LDC-R&X and an inverse time delay

For input to the learning algorithm a total of 40 events were used for each case, equally divided between voltage step-up and step-down operations. The estimation values along with the actual values are shown in Table I. As can be seen, the algorithm was successfully able to accurately estimate the parameters, with the case of the inverse time delay logic yielding the least accurate estimates. The underestimation of T_{user} was partly due to the overestimation of BW as well as a small number of events who experienced a discrete jump in their voltage time series profile. The inaccuracy in estimating R_{Drop} and X_{Drop} highlights the fact that in practice there may be insufficient excitation of the system to accurately estimate both parameters independently due to the objective suffering from multiple local minima. In such a case the system is under-determined. For the purpose of monitoring a regulator for abnormal operation it is less critical that each independent parameter be correctly identified but rather that we can determine whether a tap is consistent with historical behavior or not. For model validation purposes, however, we would require more events in order to obtain a unique optimal value.

TABLE I
Comparing parameter estimations for simulated data

		V_{Target}	BW	T_{user}	Z_{Drop}	R_{Drop}	X_{Drop}
Case 1	**Estimated**	119.98 V	2 V	20.06s	0.02	-	-
	Actual	120 V	2 V	20 s	0	-	-
Case 2	**Estimated**	118.9 V	2.12 V	19.71 s	4.16	-	-
	Actual	119 V	2 V	20 s	4	-	-
Case 3	**Estimated**	118.73 V	2.2 V	16.83 s	-	2.28	3.21
	Actual	119 V	2 V	20 s	-	2	4

In order to empirically investigate the performance of the algorithm for various system conditions for the case of a static time delay we consider two cases. In the first of these cases we examine the performance of the algorithm under different feeder voltage profiles, with varying volatility. We use field recorded voltage measurements from two different distribution feeders whose one-second voltage difference variance is shown in Table II. For the second case we use measurements from one of these sites, Dataset 1, and examine the performance of the algorithm under two different regulator controller time delay errors, 0.5% and 5%. We assume that the time delay errors are independently identically distributed, drawn from Gaussian distribution with variance ϵ, where ϵ is the specified accuracy bounds. As discussed in Section III-A, we assume that the controller acts upon averaged measurements to minimize measurement error [18]. For each case the OLTC controller took as inputs both magnitude and angle of voltage and current measurements and simulated the control logic presented in Section III.

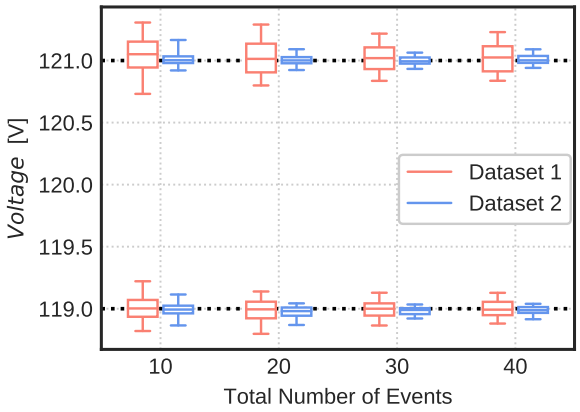

Fig. 10. $\hat{v}_{upper}$ and $\hat{v}_{lower}$ estimates for different voltage volatility levels.

TABLE II
Field measured one second voltage variance

	Var(V(t)-V(t-1s)) [%]
Dataset 1	0.056
Dataset 2	0.048

Fig. 10 and Fig. 11 consider the impact of volatility and time delay error respectively. As predicted in Section IV more volatile voltage profiles and/or higher inaccuracies in the time delay result in larger variances in the parameter estimations. It is notable that although there is initially a difference in parameter estimation variances in Fig. 10, this difference stabilizes above 30 events. A difference in the accuracy of the time delay, however, has a more pronounced impact with a offset in the median of the estimated parameters.

Under the operation of a controller with a time delay inaccuracy of $\pm 0.5\%$ [18], and given that the median number of operations per year for substation OLTC's is 3,000 [28], Fig. 10 indicates that the expected learning period is on the order of days. For capacitors, who typically switch less-frequently, the learning period may be longer.

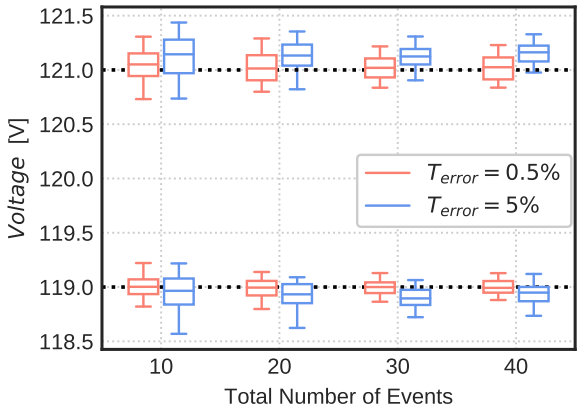

Fig. 11. $\hat{v}_{upper}$ and $\hat{v}_{lower}$ estimates for different controller time delay errors.

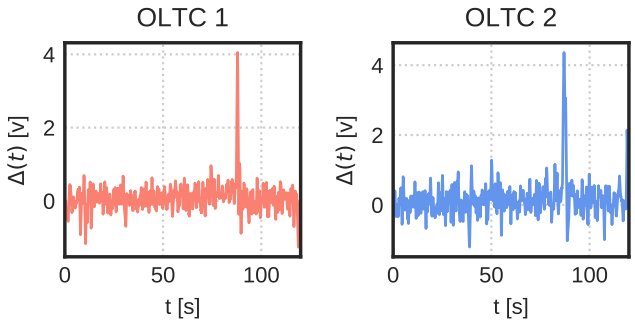

Fig. 12. Estimating T_{user} via Delta time series profile.

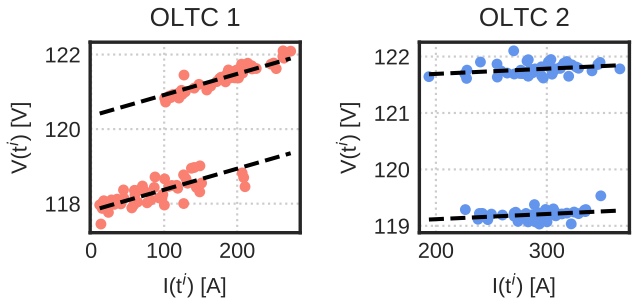

Fig. 13. Estimated lower- and upper-thresholds.

B. Utility Recorded Data

In order to validate our approach using real data we consider two OLTC transformers at pilot site deployments whose secondary terminals are being measured by a distribution PMU reporting at 120 Hz. Both locations are independent distribution feeders electrically connected through a sub-transmission network. In order to identify operation of the OLTC, both sensors were used to classify events as either global, i.e. originating from the transmission or subtransmission network, or local, i.e. OLTC transformers. Once a set of tap operations has been identified, the control logic is identified and parameterized. For each site, initially a total of 100 events were randomly chosen, equally divided between voltage step-up and voltage step-down operations. Voltages are presented on a 120 V basis. In both cases, the $\mathbf{\Delta}$ time series profile exhibited one distinct peak, as can be seen in Fig. 12, and correspondingly an initial estimate for T_{user} was obtained. Estimated and reported quantities are shown in Table III for both devices. For the case of OLTC 1, when the optimization to determine the upper- and lower-thresholds was carried out there was only a minor difference between the RMSE's as shown in Table IV. The primary reason for this is that 74% of the events had a value for $pf(\hat{t}_c^i) > 0.95$, and consequently we can see from (2) and (3) that $R_{Drop}pf(t_c^i) + X_{Drop}qf(t_c^i) \approx Z_{Drop}$. For different feeders with more variations in power factor, this difference in RMSE is expected to increase. The estimated LDC relationship for both OLTC 1 and OLTC 2 are shown is Fig. 13. The RMSE for OLTC 2 was noticeably less than that for OLTC 1. This difference is attributed to the difference in measured current levels. As can be seen in Fig. 13, there were very low current levels measured for OLTC 1. This is due to a large penetration of photovoltaic installations on that particular feeder which supplies a significant portion locally and thus reduces the transformer loading. These low loading conditions impact the parameter estimation due to the non-linear errors introduced by the current transformer through which we are stepping down the current for measurement. At such low current levels, the errors associated with the current transformer may no longer be assumed to be a stable bias offset. It is unclear how the error manifests itself but it is worth noting that the measurements for OLTC 1 in Fig. 13 are significantly denser at higher current levels.

TABLE III
COMPARING PARAMETER ESTIMATIONS FOR UTILITY MEASURED DATA

		V_{Target}	BW	T_{user}
OLTC 1	**Estimated**	119.07 V	2.54 V	32.26 s
	Reported	119 V	3 V	30 s
OLTC 2	**Estimated**	120.21 V	2.56 V	31s
	Reported	120 V	3 V	30 s

TABLE IV
RMSE FROM MODEL FITS ON UTILITY DATA

	LDC-Z	**LDC-R& X**
OLTC 1	3.43 V	3.56 V
OLTC 2	0.80 V	0.74 V

Minor discrepancies between estimated and reported quantities are within the scale calibration accuracy of the controller [29]. Table V-B shows an estimate of the threshold for abnormal event detection as described in Section IV-D. Again, the uncertainty for OLTC 1 is larger due to OLTC operations at low current levels, well below the rated current of the current transformer.

In order to gain further understanding into the performance of the algorithm on field measured data we investigate its performance by i) varying the number of events that exhibit a

	Abnormal Threshold
OLTC 1	0.56 V
OLTC 2	0.27 V

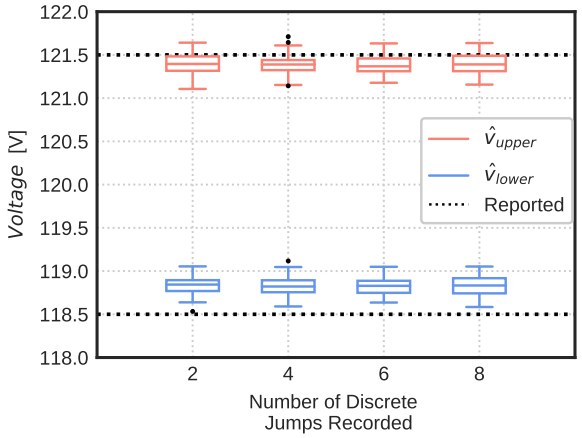

Fig. 14. Algorithm performance for $m = 80$ events with varying number of series with discrete jumps for OLTC 2.

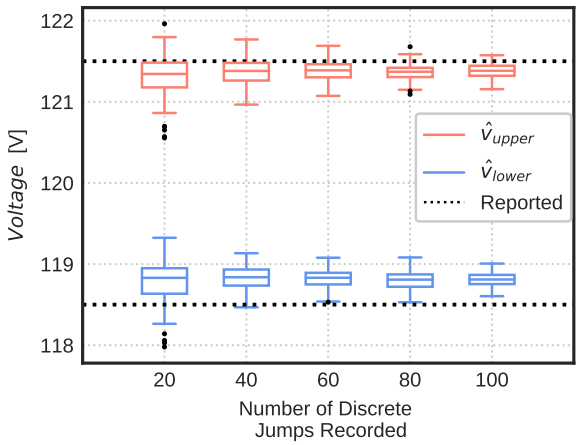

Fig. 15. Algorithm performance for varying number of events with 8 number of time series with discrete jumps for OLTC 2.

discrete jump in their voltage profile while keeping the total number of events constant and ii)varying the total number of events for a given estimate of T_{user}, the time delay. A boxplot plot showing the range of estimations for both the upper- and lower-threshold for these cases are shown in Fig. 14 and Fig. 15. For each scenario we randomly choose events with replacement from our library of events and carry out the estimation. This is repeated a total of 100 times for each condition of interest.

As expected, both Fig. 14 and Fig. 15 support the intuition that with additional training data, the range of the estimates decrease. This growing library of historical events allows us to have greater confidence in labeling an action as normal or abnormal. Fig. 14 indicates that, for the case of a static delay, the number of discrete jumps has minimal impact of the performance. So long as there is a sufficient number to obtain a coarse estimation of T_{user}, allowing the optimization to search locally over t^i results in a stable estimation. Fig. 15 seem to indicate that for this particular feeder, estimation ranges stabilize above 60 total events, with minimal improvement beyond 80 events. However, for the purpose of detecting abnormal behavior the approach performs a lower number of events can be used with a minor degradation in performance, as indicated by the range of estimations in Fig. 15.

VI. Conclusions

In this work we proposed an online algorithm for passively learning and monitoring the control logic of distribution OLTC transformers and/or switched capacitor banks. The demonstrated ability of our approach to learn the control logic of devices negates the need for an operator to specify the settings a priori, facilitating the deployment of the cyber-physical data analytic engine described in Fig. 1. Also, the proposed approach does not inherit errors arising from inaccurate/outdated databases or human input and allows for further decoupling and isolation of the independent (read-only) sensor network, which increases its integrity. The proposed approach was validated on both simulated and utility recorded data.

A limitation of the proposed method is the requirement that there exist large discrete jumps in subset of the voltage profiles prior to the execution of a control action. These discrete jumps are exploited for obtaining a coarse estimation of the controller time delay, around which the optimization algorithm is allowed to explore. For the case of an inverse time delay in particular, feeders with fewer of these events will cause the performance of the learning algorithms to decrease. Follow-on work will seek to relax this requirement, by considering the statistical properties of a family of events prior to execution of a tap change or capacitor switching logic. The expected impact is that the learning period would be reduced for the case of infrequent discrete jumps in the voltage profile causing it to exit its allowable range. Future work could also seek to investigate an optimal placement algorithm that achieves sufficient observability while minimizing the total number of PMUs. Additionally, while the proposed learning algorithms are suitable for devices that operate on local measurements, they are not suitable for feeders that may employ a centralized volt/var optimization for determining control device behavior in the future. Follow-on work would seek to extend the underlying idea within this paper and adapt a purely data-driven learning algorithm for inferring such control logic.

References

[1] S. Sheng, W. L. Chan, K. K. Li, D. Xianzhong, and Z. Xiangjun, "Context information-based cyber security defense of protection system," *IEEE Transactions on Power Delivery*, vol. 22, no. 3, pp. 1477–1481, July 2007.

[2] S. Pan, T. Morris, and U. Adhikari, "Developing a hybrid intrusion detection system using data mining for power systems," *IEEE Transactions on Smart Grid*, vol. 6, no. 6, pp. 3104–3113, 2015.

[3] S. Sridhar and M. Govindarasu, "Model-based attack detection and mitigation for automatic generation control," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 580–591, 2014.

[4] Y. Yang, K. McLaughlin, S. Sezer, T. Littler, E. G. Im, B. Pranggono, and H. Wang, "Multiattribute scada-specific intrusion detection system for power networks," *IEEE Transactions on Power Delivery*, vol. 29, no. 3, pp. 1092–1102, 2014.

[5] J. Hong, C.-C. Liu, and M. Govindarasu, "Integrated anomaly detection for cyber security of the substations," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1643–1653, 2014.

[6] A. M. Kosek and O. Gehrke, "Ensemble regression model-based anomaly detection for cyber-physical intrusion detection in smart grids," in *Electrical Power and Energy Conference (EPEC), 2016 IEEE*. IEEE, 2016, pp. 1–7.

[7] A. M. Kosek, "Contextual anomaly detection for cyber-physical security in smart grids based on an artificial neural network model," in *2016 Joint Workshop on Cyber-Physical Security and Resilience in Smart Grids (CPSR-SG)*. IEEE, 2016, pp. 1–6.

[8] A. von Meier, E. Stewart, A. McEachern, M. Andersen, and L. Mehrmanesh, "Precision micro-synchrophasors for distribution systems: A summary of applications," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2926–2936, 2017.

[9] A. Borghetti, C. A. Nucci, M. Paolone, G. Ciappi, and A. Solari, "Synchronized phasors monitoring during the islanding maneuver of an active distribution network." *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 82–91, 2011.

[10] H. Mohsenian-Rad, E. Stewart, and E. Cortez, "Distribution synchrophasors: Pairing big data with analytics to create actionable information," *IEEE Power and Energy Magazine*, vol. 16, no. 3, pp. 26–34, May 2018.

[11] S. Cheung, B. Dutertre, M. Fong, U. Lindqvist, K. Skinner, and A. Valdes, "Using model-based intrusion detection for SCADA networks," in *Proceedings of the SCADA security scientific symposium*, vol. 46, 2007, pp. 1–12.

[12] B. Zhu and S. Sastry, "SCADA-specific intrusion detection/prevention systems: a survey and taxonomy," in *Proceedings of the 1st Workshop on Secure Control Systems (SCS)*, vol. 11, 2010, p. 7.

[13] H. Lin, A. Slagell, C. Di Martino, Z. Kalbarczyk, and R. K. Iyer, "Adapting Bro into SCADA: building a specification-based intrusion detection system for the DNP3 protocol," in *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop*. ACM, 2013.

[14] "IEEE standard for interconnection and interoperability of distributed energy resources with associated electric power systems interfaces," *IEEE Std 1547-2018 (Revision of IEEE Std 1547-2003)*, pp. 1–138, April 2018.

[15] G. Liang, S. R. Weller, J. Zhao, F. Luo, and Z. Y. Dong, "The 2015 Ukraine Blackout: Implications for False Data Injection Attacks," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 3317–3318, Jul. 2017.

[16] V. Paxson, "Bro: A System for Detecting Network Intruders in Real-Time," *Computer Networks*, vol. 31, no. 23, pp. 2435–2463, 1999.

[17] T. A. Short, *Electric Power Distribution Handbook*. CRC press, 2014.

[18] *KVGC202 Technical Manual: Voltage Regulating Control Relays*, GE, Publication Reference: KVGC202/EN/M/E11.

[19] E. Jauch, "Advanced tapchanger control features–and when to use them!(part one)," in *2005/2006 IEEE/PES Transmission and Distribution Conference and Exhibition*, 2006.

[20] F. A. Viawan, A. Sannino, and J. Daalder, "Voltage control with on-load tap changers in medium voltage feeders in presence of distributed generation," *Electric Power Systems Research*, vol. 77, no. 10, pp. 1314–1322, 2007.

[21] K. V. Khandeparkar, S. A. Soman, and G. Gajjar, "Detection and correction of systematic errors in instrument transformers along with line parameter estimation using pmu data," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 3089–3098, 2016.

[22] *CQ900 Capacitor controller Installation and Operation Manual*, ABB, Manual Rev 1.6.

[23] *CBC-8000 capacitor bank control installation and operation instructions*, EATON, October 2016.

[24] "Ieee standard requirements for tap changers," *IEEE Std C57.131-2012 (Revision of IEEE Std C57.131-1995)*, pp. 1–73, May 2012.

[25] D. B. Arnold, C. Roberts, O. Ardakanian, and E. M. Stewart, "Synchrophasor data analytics in distribution grids," in *Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), 2017 IEEE*. IEEE, 2017, pp. 1–5.

[26] B. B. Purkayastha and K. K. Sarma, *A digital phase locked loop based signal and symbol recovery system for wireless channel*. Springer, 2015.

[27] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.

[28] J. H. Harlow, *Electric Power Transformer Engineering*. CRC press, 2003.

[29] *Tapchanger Control M-0067E Specification*, BECKWITH ELECTRIC Co., INC, Specification 800-0067E-SP-00MC6 01/13.