Partial Knowledge Data-driven Event Detection for Power Distribution Networks

Yuxun Zhou[1], Reza Arghandeh[2], *Member, IEEE*, Costas J. Spanos[3] *Fellow, IEEE*

***Abstract*—The power system has been incorporating increasing amount of unconventional generations and loads such as distributed renewable resources, electric vehicles, and controllable loads. The induced dynamic and stochastic power flow require high-resolution monitoring technology and agile decision support techniques for system diagnosis and control. This paper discusses the application of micro-phasor measurement unit (μPMU) data for power distribution network event detection. A novel data-driven event detection method, namely Hidden Structure Semi-Supervised Machine (HS^{3}M), is established. HS^{3}M only requires partial expert knowledge: it combines unlabeled data and partly labeled data in a large margin learning objective to bridge the gap between supervised learning, semi-supervised learning, and learning with hidden structures. To optimize the non-convex learning objective, a novel global optimization algorithm, namely Parametric Dual Optimization Procedure (PDOP), is established through its equivalence to a concave programming. Finally, the proposed method is validated on an actual distribution feeder with installed μPMUs, and the result justifies the effectiveness of the learning-based event detection framework, as well as its potential to serve as one of the core algorithms for power system security and reliability.**

***Index Terms*—Event Detection, Phasor Measurement Unit, Distribution, Machine Learning.**

NOMENCLATURE

$\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$	Lagrangian multipliers for the three types of constraints in the primal; Decision variables in the Dual
$\boldsymbol{\eta}$	Indicator variables for partly labeled data
$\boldsymbol{\lambda}$	Unified hidden variable in the Dual
$\boldsymbol{\theta}$	Unified decision variable in the Dual
$\boldsymbol{\zeta}$	Indicator variables for unlabeled data
$\boldsymbol{C}^\theta, \boldsymbol{C}^\lambda, \boldsymbol{C}^0$	Matrices encoding the coefficients of $\boldsymbol{\theta}$,$\boldsymbol{\lambda}$ and the constant term in the constraints.
$\boldsymbol{e}$	Column vector of all 1s
$\boldsymbol{Q}$	Kernel matrix in the Dual
$\boldsymbol{T}$,$\boldsymbol{H}$,$\boldsymbol{P}$	Composite matrices defined in Theorem 1
$\boldsymbol{w}$	Parameter of the classifier
$\boldsymbol{x}_i$	The i^{th} data sample (system measurement)
$\kappa(\cdot,\cdot)$	Kernel function
Λ,Θ	Feasible set for $\boldsymbol{\lambda}$ and $\boldsymbol{\theta}$, respectively
$\mathcal{A}$	Index set of active constraints
$\mathcal{H}$	Hilbert function space
$\mathcal{J}(\boldsymbol{\theta})$	Value of the dual objective as a function of $\boldsymbol{\theta}$
$\mathcal{L}^+$	Index set of all samples having $y_i = +1$
$\mathcal{L}^-$	Index set of completely labeled samples having $y_i = -1$
$\mathcal{L}_H^+$	Index set of partly labeled samples having $y_i = +1$
$\mathcal{L}_H^-$	Index set of partly labeled samples having $y_i = -1$
$\mathcal{U}$	Index of unlabeled data samples
$\widehat{y}_i$	Tentative label for sample $i \in \mathcal{U}$
$\widetilde{\boldsymbol{y}}$	Augmented label vector, including tentative and dummy labels
c ,c_1-c_3	Classification loss penalties
K	Number of event classes
k	Index for the k^{th} base classifier
y_i	Indicator for nominal (0) or event (1)
z_i	Event type indicator

[1]Yuxun Zhou is with Electrical Engineering Computer Sciences, University of California, Berkeley, 94720 CA, USA. yxzhou@berkeley.edu

[2]Reza Arghandeh is an Assistant Professor in the Electrical and Computer Engineering Department, Florida State University, FL 32306, USA. arghandehr@gmail.com

[3]Costas J. Spanos is the Andrew S. Grove Distinguished Professor of Electrical Engineering Computer Sciences, University of California, Berkeley, 94720 CA, USA. spanos@berkeley.edu

I. INTRODUCTION

A. Background

HISTORICALLY, power distribution networks are behind transmission networks regarding transparency, measurement frequency, and sophisticated monitoring systems. However, the growth of distributed renewable energy resources, electric vehicles, power electronic based grid edge controllers, and controllable loads introduces more short-term and unpredictable disturbances in the electricity flow [1]. It suggests a need for more accurate measurement devices with higher resolution. As utilities have expanded their monitoring systems and sensor networks, the data management, analysis, and inference have become the most significant challenges with the shift from a conventional data acquisition system depended on domain experts *(supervised)* to an automated intelligent analysis system *(unsupervised)*. This paper specifically discusses high-precision synchrophasors, or micro-phasor measurement units (μPMUs) for high-fidelity measurement of voltage and current waveforms [2]. All μ-PMUs measurements are GPS time stamped to provide time-synchronized observability for three-phase voltage and current magnitude and phase angle with a 0.05% Total Vector Error [3]. Figure 1 shows the actual μ-PMU device and the overall system architecture. The μ-Pnet is the equivalent of the Phasor Data Concentrator (*PDC*) in a distribution network. More technical information about the μ-PMU device is available in [4]. Topology detection[5], phase labeling [6] and linear state estimation[7] are among applications of μ-PMU data that are explored so far. Moreover, the accuracy and resolution available from such μ-PMU monitoring network enable operators to detect short-time events that would otherwise be unobservable in distribution systems.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2681962, IEEE Transactions on Smart Grid

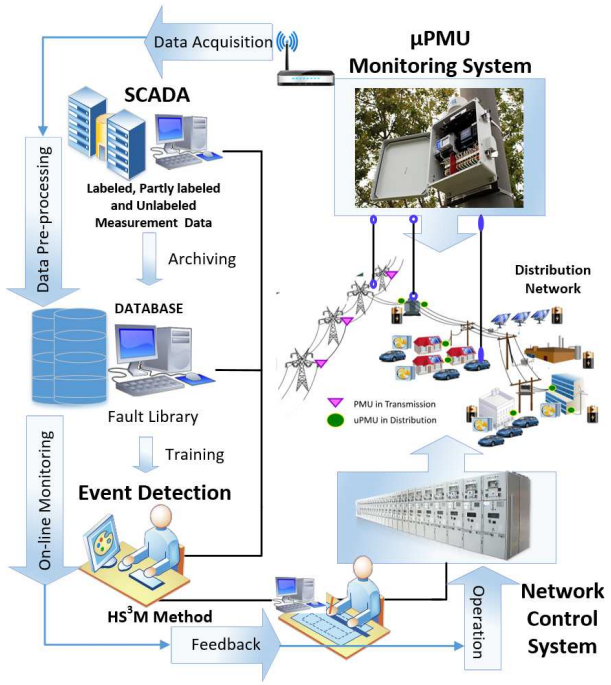


Fig. 1: The integration of the proposed event detection tool with existing μ-PMU monitoring and SCADA system.

Events of interest in distribution networks are classified into different categories based on their duration, cause, and location. Two sample events, captured by a μ-PMU, are shown in Figure 2. Events in power systems present themselves as oscillations, transients, short duration variations, long-duration variations, and waveform distortion in voltage, current and frequency values [8]. The causes are usually faults, switching actions, topology changes, controllers, load behavior and source dynamics. Related standards for characterizing events in distribution network including but are not limited to [9], [10], [11]. Moreover, the initial fault or event occurring at one or more components of the grid usually triggers the faults or events of others [12], [13], [14], [15]. For the sake of power systems reliability, stability, security, and resilience, it is crucial to monitor the operating states in real time and detect anomalies quickly to avert disturbances and disruptions[16]. Besides μ-PMU based event detection system in distribution networks, the proposed data-driven tool in this paper is compatible with different types of measurement such as traditional measurement devices, power quality recorders, protection relays, smart meters, and synchrophasors. Within a larger scope, this paper aims to leverage advances in machine learning techniques, global optimization methods and time-series data analysis for data-driven event detection in power distribution networks.

B. Learning Based Event Detection Methods

A novel framework for event detection is proposed in this paper using μ-PMU data streams. A large body of available literature has been addressing the problem of event detection, in particular, its simplified case of novelty/fault detection with various methods. Available works on event detection can be divided into two main categories and their combinations, i.e., model-based method and model-less or data-driven method.

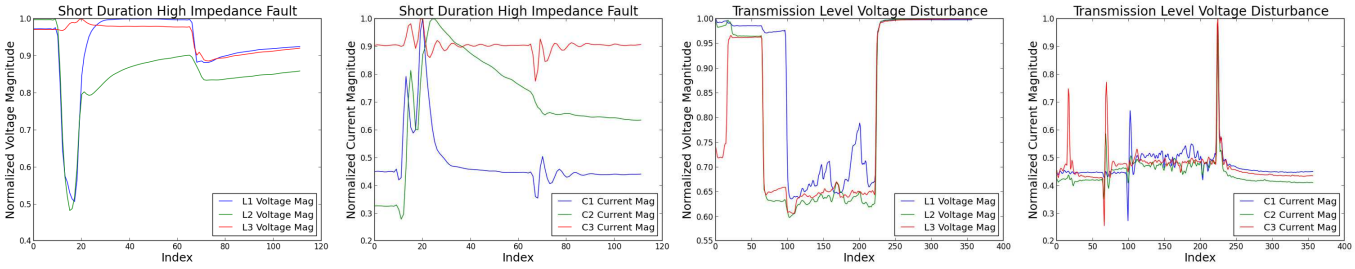
The basic idea of model-based approaches is to compare the system behavior, estimated by a dynamic model, to the expected behavior when the system is in a certain state [17]. To list a few, model-based applications range from cyber attacks identification in power system [18], fault diagnosis for switching converters [19], and fault-detection of electric machine [20]. Recently, model-based approaches have also been used in combination with time series analysis to establish semi-model-based algorithms [21]. This type of methods is largely based on the correctness of the dynamic model of the system, as well as system analytic tools such as real-time state estimators and parity equations. Their limitations are evident as dynamics of a system may be (1) hard to specify and (2) nonlinear/coupled in structure. Also, as more and more applications are dealing with a complex system with randomness, the high dimensionality and inherent uncertainty significantly deteriorate the reliability of dynamic models.

On the other hand, the data-driven approaches use methods of machine learning to conduct statistical inference or decision making on available system measurements. As massive amount of data is provided by the advancement of sensor network and information technology, this approach is receiving increasing attention in both application and research domains. The problem of distinguishing abnormalities from the normal states has been extensively studied in literature and is referred to as novelty (or abnormal) detection. Indeed, many classic machine learning tools, such as kernel Principle Component Analysis (kPCA), Partial Least Squares (PLS), and Fisher Discriminant Analysis (FDA) have been widely applied in various fields. Readers are referred to [22] and the references therein for a comprehensive survey.

The event detection problem becomes more challenging as the objective is extended from making a binary decision to distinguishing the type of events. Recently several attempts were made for this purpose, including cost sensitive Support Vector Machine (SVM) for fault type identification in semiconductor manufacturing process [23] and the hierarchical kernel method for building cooling system fault diagnosis [24]. These methods belong to the *supervised* multi-class classification which require properly labeled data with detailed event types as the training set. However, the occurrence of events might be rare in many real-world applications, and the availability of fully labeled data set is often limited. In the situation dealing with μ-PMU data, complete expert knowledge for power system diagnosis and event labeling can be expensive and scant. Lacking labeled data may cause insufficient supervised learning and eventually lead to degraded performance.

In this work, a novel data-driven event detection method, namely the Hidden Structure Semi-Supervised Machine (HS^{3}M), is proposed for the purpose of event type identification. To resolve the scarcity of labeled training data, HS^{3}M also incorporates information from *partly labeled* data and *unlabeled* data. The inclusion of this partial information breaks the convexity of traditional large margin learning formulation. As such a new global optimization algorithm is developed to solve the non-convex learning problem. The proposed method

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2681962, IEEE Transactions on Smart Grid



(a) Voltage Magnitude: HI (b) Current Magnitude: HI (c) Voltage Magnitude: VD (d) Current Magnitude: VD

Fig. 2: Real time μ-PMU measurements that contains events. (a)&(b): Voltage/Current measurements when a short duration high impedance fault occurs. (c)&(d)Voltage/Current measurements when a transmission level voltage disturbance occurs.

is applied to a concrete case of power distribution network event detection with μ-PMU data. The integration of the event detection tool as a complement of the existing monitoring and control system is shown in Figure 1.

The contributions are highlighted as follows:

- HS^{3}M provides a unified framework to incorporate data of different natures for event detection. The proposed method bridges the gap between semi-supervised learning and learning with hidden structures.
- The newly developed PDOP algorithm constitutes a promising substitute for previous methods such as Concave-Convex Procedure and alternating optimization.
- HS^{3}M yields significant improvement compared to state-of-the-art methods, showing promising results on distribution network event detection with μ-PMU data.

The paper is organized as follows, in section II we introduce the formulation of HS^{3}M learning objective and the intuition behind it. Section III is devoted to transforming the training problem into a parametric quadratic program. In Section IV, the PDOP algorithm is established for HS^{3}M learning. The case study is given in section V.

II. HS^{3}M Event Detection: Motivation and Formulation

Traditionally, there have been two fundamentally different paradigms of machine learning (ML). The first one is the supervised learning, with the goal of learning a mapping from some input $\boldsymbol{x}$ to output y. Usually the observations $(\boldsymbol{x}_i, y_i), i = 1, \cdots, n$ are called samples, $\boldsymbol{x}_i$ are referred to as features of sample i, and $y_i \in \mathcal{Y}$ are called labels or targets. To find the "optimal" mapping f, the learning task is commonly formulated as "regularized empirical risk minimization" [25]

$$\min_{f \in \mathcal{H}} \frac{1}{2}||f||^2_{\mathcal{H}} + c \sum_i L(y_i, f(\boldsymbol{x}_i)) \tag{1}$$

in which the second term measures the "goodness of fit" of the classifier f with some loss function L, and the first regularization term controls the complexity of the mapping f to avoid over-fitting[1][26]. By tuning the "hyper-parameter" c (called "model selection" in the jargon of ML), one is able to balance training fitness and model complexity, hence finding the optimal classifier that generalizes well to unseen data set.

[1]Known as the Occam's razor principle. In some context the regularization can also help alleviate ill-posed problem and induce sparsity.

The second task of ML is unsupervised learning [27]. Under this setting, only the unlabeled observations $X = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\}$ are given. Typically, the goal of unsupervised learning is to identify interesting structures in the data X, such as clusters, quantiles, support, low-dimensional embedding, or more generally the patterns related with the distribution of the data.

The presence of both labeled and unlabeled data motivates the so-called semi-supervised learning [28]. The hope is that, by combining both types of available data sets, semi-supervised learning could find better models/classifiers, and reduce the cost of expert engagement [29]. In the context of event detection using μ-PMU measurement, data with detailed event labels is precious but scant - power system experts are needed to inspect the measurement and mark events. On the other hand partly labeled data with incomplete labels, e.g., nominal/fault, may be obtained less costly by using decision support systems. Moreover, unlabeled data can be acquired in large quantity simply by collecting μ-PMU measurement. Given the characteristics of the different types of data available for power system event detection, a unified ML framework is proposed to leverage all information sources. The following subsection explains the construction of the proposed HS^{3}M step-by-step.

A. Motivation and Intuition of HS^{3}M

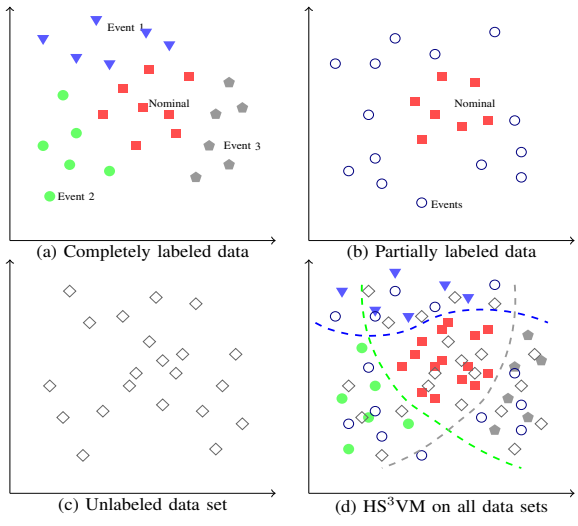
To formalize the information availability in different scenarios, consider data of the following three formats:

1 Completely labeled data samples, denoted as $\{\boldsymbol{x}_i, y_i, z_i\}$, where i is the sample index, $\boldsymbol{x}_i$ is the system measurement, y_i is a indicator for "nominal/stable state" ($y = +1$) or an "event" ($y = -1$). If $y = -1$, an event type indicator $z_i \in \{1, \cdots, K\}$ is also associated.

2 Partly labeled data samples, denoted as $\{\boldsymbol{x}_i, y_i, \cdot\}$, where y_i is still the indicator for "event", but event type is unavailable as differentiating event types is costly.

3 Unlabeled data samples, denoted as $\{\boldsymbol{x}_i, \cdot, \cdot\}$, where only system measurement $\boldsymbol{x}_i$ is accessible.

An illustration of different situations is given in Fig. 3. Intuitively, partly labeled data should be helpful: at least it provides discriminating information between stable state and events. The role of unlabeled data might be ambiguous since it does not carry any expert knowledge. However, it does contain distributional information of measurement, which



(a) Completely labeled data (b) Partially labeled data

(c) Unlabeled data set (d) HS^{3}VM on all data sets

Fig. 3: Different data format and the intuition of HS^{3}M

could be exploited with a proper formulation. In the sequel, an unified learning framework is designed to combine all three information sources to improve detection performance.

To begin with, the regularized empirical risk minimization framework is substantiated by hing loss and L_2 regularization:

$$\min_{\boldsymbol{w}} \frac{1}{2}||\boldsymbol{w}||_2^2 + c\sum_i [1 - y_i g(\boldsymbol{w}, \boldsymbol{x}_i)]_+ \tag{2}$$

where c is the loss penalties, $[1 - y_i g(\boldsymbol{w}, \boldsymbol{x}_i)]_+$ is the hinge loss for the miss classification, and $\boldsymbol{w}$ is the parameters of the classifier $g(\boldsymbol{w}, \boldsymbol{x}_i)$. The regularization term $||\boldsymbol{w}||_{\mathcal{H}}^2$ also favors the large margin separating hyperplanes. The HS^{3}M formulation resembles this "regularization+hinge loss" minimization, but both partly labeled data and unlabeled data are incorporated as additional information, by constructing a new classifier having a geometric property suitable for power system event detection purposes.

B. A Geometrically Constrained Classifier

Intuitively, data generated from the nominal (stable) state are often "concentrated" in some distribution, whereas data of diverse events are "scattered around" since they deviated from the nominal in different ways. This distributional asymmetric is illustrated in the top-left panel of Figure 3. To encode this particular "geometric" information of event detection, it is natural to describe the stable state by the intersection of the acceptance region of multiple nonlinear decision rules, while the events class by the union of their complements, as is shown in Figure 3(d). Mathematically, a composed classifier $g(\cdot)$ can be written in terms of multiple base classifiers $f_k(\cdot)$ as follows:

$$g(\boldsymbol{w}, \boldsymbol{x}) > 0 \Leftrightarrow \min\{f_1(\boldsymbol{w}, \boldsymbol{x}), \cdots, f_K(\boldsymbol{w}, \boldsymbol{x})\} > 0$$

where K is the number of all possible event types. The construction of the classifier inherently emphasizes the sensitivity to event class because an event is detected by any one of the base classifiers. The classifier also maintains the specificity to stable state class, as all base classifiers have to "agree" for a positive prediction. Given a feature mapping $\phi : \mathbb{R}^d \to \mathcal{H}$, a hyperplane classifier in the new Hilbert space can be written as $f(\boldsymbol{w}, \boldsymbol{x}) = \langle \boldsymbol{w}, \phi(\boldsymbol{x}) \rangle_{\mathcal{H}} + b \triangleq \boldsymbol{w} \cdot \phi(\boldsymbol{x})$ for short hand notation. Then the proposed classifier is

$$g(\boldsymbol{w}, \boldsymbol{x}) = \min_k \{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x})\}. \tag{3}$$

1) Incorporating Labeled Data: Based on the composed classifier, the hinge loss $[1 - y_i g(\boldsymbol{w}, \boldsymbol{x}_i)]_+$ is computed for each format of data to incorporate their "contribution" to the classifier. With detailed event type being available, the hinge loss for labeled data set can be directly obtained as $[1 + \boldsymbol{w} \cdot \phi_{z_i}(\boldsymbol{x}_i)]_+$ when $y_i = -1$ and $[1 - \min_k\{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x_i})\}]_+$ when $y_i = +1$.

2) Incorporating Partly Labeled Data: Comparing Figure 3(a) and 3(b), we see that partly labeled data could be viewed as data with "missing detailed labels" (event types). But the hidden clusters of each event class still exist. From a ML perspective, this is the problem of "learning with hidden structures". Composed from multiple base classifiers, the unified classifier g performs implicit "clustering" for the event class, so as to capture hidden subgroups in partly labeled data sets. Thus the hinge loss for a partly labeled data sample $\{\boldsymbol{x}_i, y_i, \cdot\}$ is just $[1 - y_i \min_k\{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x_i})\}]_+$ for $y_i = \pm 1$.

3) Incorporating Unlabeled Data: Given a classifer $g(\boldsymbol{w}, \boldsymbol{x})$, the predicted label is its sign, i.e., $\widehat{y}_i = \text{sign}(g(\boldsymbol{w}, \boldsymbol{x}))$. This inspires a tentative labeling strategy to include information provided by unlabeled data. More specifically, with $\widehat{y}_i = \text{sign}\left(\min_k\{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x}_i)\}\right)$, the corresponding hinge loss has the form

$$\left[1 - \widehat{y}_i \min_k\{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x_i})\}\right]_+ = \left[1 - \left|\min_k\{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x}_i)\}\right|\right]_+$$

C. Overall Learning Objective

Putting things together, the following regularized hinge loss minimization is formulated for event detection that incorporates all explicit and partial expert knowledge:

$$\begin{aligned} \min_{\boldsymbol{w}} \quad & \frac{1}{2}||\boldsymbol{w}||_{\mathcal{H}}^2 + c_1 \sum_{i \in \mathcal{L}^+} \left[1 - \min_k\{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x_i})\}\right]_+ \\ & + c_{21} \sum_{i \in \mathcal{L}^-} [1 + \boldsymbol{w} \cdot \phi_{z_i}(\boldsymbol{x}_i)]_+ \\ & + c_{22} \sum_{i \in \mathcal{L}_H^-} \left[1 + \min_k\{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x}_i)\}\right]_+ \\ & + c_3 \sum_{i \in \mathcal{U}} \left[1 - \left|\min_k\{\boldsymbol{w} \cdot \phi_k(\boldsymbol{x}_i)\}\right|\right]_+ \end{aligned} \tag{ORIG}$$

where $\mathcal{L}^+$ denotes the index set of all data samples that has $y_i = +1$, including both completely and partly labeled samples, $\mathcal{L}^-$ as the index set of completely labeled samples with $y_i = -1$ and event type z_i (hence the hinge loss only involves the corresponding individual classifier f_{z_i}). The index set $\mathcal{L}_H^-$ contains partly labeled samples that are "events", and $\mathcal{U}$ is the index of all unlabeled data samples. The loss penalty hyper-parameters c_1-c_3 weight each loss term differently, and should be chosen by taking into account the imbalanced cost for false positive and false negative error, sample size in each category, as well as for model selection considerations. Since the above formulation deals with both hidden structures and unlabeled samples in the available data, we call it Hidden Structured Semi-Supervised Machine (HS^{3}M).

III. Learning Objective Reformulation

The first three terms in the learning objective **ORIG** are convex in decision variables, the last two terms, however are not convex. Existing heuristics for this type of problems use either group alternating optimization or Concave-Convex Procedure (CCCP), which only converge to local minimals and may lead to deteriorated solution. The degradation especially happens as we are in a machine learning context [30].

In this work, a novel optimization algorithm is derived for HS^{3}M as well as for a class of "hidden structure" problems. A remarkable property of the algorithm is that it can approach global optimum by iteratively improving local solutions. The new method is based on an equivalent optimization problem and two important parametric property of its dual. As the first step, we transform **ORIG** by introducing additional "hidden" variables, and write the learning objective in the following joint optimization form:

Proposition 1. ***ORIG*** *is equivalent to* ***(OPT1)***

$$\begin{aligned}\min_{\boldsymbol{\eta},\boldsymbol{\zeta}}\min_{\boldsymbol{w}} \quad & \frac{1}{2}||\boldsymbol{w}||_{\mathcal{H}}^2 + c_1\sum_{i\in\mathcal{L}^+}\Big[1-\min_k\{\boldsymbol{w}\cdot\phi_k(\boldsymbol{x_i})\}\Big]_+ \\ & + c_{21}\sum_{i\in\mathcal{L}^-}\left[1+\boldsymbol{w}\cdot\phi_{z_i}(\boldsymbol{x}_i)\right]_+ \\ & + c_{22}\sum_{i\in\mathcal{L}_H^-}\sum_{k=1}^{K}\eta_{ik}\left[1+\boldsymbol{w}\cdot\phi_k(\boldsymbol{x}_i)\right]_+ \\ & + c_3\sum_{i\in\mathcal{U}}\sum_{k=1}^{K}\zeta_{ik}\left[1+\boldsymbol{w}\cdot\phi_k(\boldsymbol{x}_i)\right]_+ \\ & + c_3\sum_{i\in\mathcal{U}}\zeta_{i(K+1)}\max_j\{0,1-\boldsymbol{w}\cdot\phi_j(\boldsymbol{x}_i)\}\end{aligned}$$

subject to $\quad \boldsymbol{\eta}_i\in\mathbb{S}^K,\ \forall i\in\mathcal{L}_H^-; \quad \boldsymbol{\zeta}_i\in\mathbb{S}^{K+1},\ \forall i\in\mathcal{U}$

In addition, the two minimization is interchangeable.

The introduced variables η and ζ can be thought of as hidden state indicators for partially labeled data and unlabeled data, respectively. The corresponding dual of the **inner optimization** of OPT1 is

$$\begin{aligned}\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}} \ & -\frac{1}{2}\left\|\sum_{k,i\in\mathcal{I}}\alpha_{ik}y_i\phi_k(\boldsymbol{x}_i)+\sum_{i\in\mathcal{L}^-}\beta_i\phi_{z_i}(\boldsymbol{x}_i)+\sum_{k,i\in\mathcal{U}^+}\gamma_{ik}\phi_k(\boldsymbol{x}_i)\right\|_{\mathcal{H}}^2 \\ & + \sum_{k,i\in\mathcal{I}}\alpha_{ik}+\sum_{i\in\mathcal{L}^-}\beta_i+\sum_{k,i\in\mathcal{U}^+}\gamma_{ik}\end{aligned}$$

subject to

$$\begin{cases}\alpha_{ik}\geq 0; \quad \sum_k\alpha_{ik}\leq c_1 \quad \forall i\in\mathcal{L}^+ \\ 0\leq\beta_i\leq c_{21} \quad \forall i\in\mathcal{L}^- \\ 0\leq\alpha_{ik}\leq c_{22}\eta_{ik} \quad \forall i\in\mathcal{L}_H^- \\ 0\leq\alpha_{ik}\leq c_3\zeta_{ik} \quad \forall i\in\mathcal{U}^- \\ \gamma_{ik}\geq 0; \quad \sum_k\gamma_{ik}\leq c_3\zeta_{i(K+1)} \quad \forall i\in\mathcal{U}^+ \\ \sum_{k,i\in\mathcal{I}}y_i\alpha_{ik}+\sum_{k,i\in\mathcal{L}^-}y_i\beta_i+\sum_{k,i\in\mathcal{U}^+}y_i\gamma_{ik}=0\end{cases} \quad \text{(Inner Dual)}$$

where the Lagrangian multipliers $\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}$ are now decision variables. Note that the unlabeled data set $\mathcal{U}$ is used as two dummy copies with tentative labels $y_i=+1$ for $i\in\mathcal{U}^+$ and $y_i=-1$ for $i\in\mathcal{U}^-$, respectively. Also for short hand notation, we denote the index set of all samples by $\mathcal{I}$, and a unified decision variable in the Dual by

$$\boldsymbol{\theta}=\left[\boldsymbol{\alpha}^T,\boldsymbol{\beta}^T,\boldsymbol{\gamma}^T\right]^T$$

where $\boldsymbol{\alpha}\triangleq\left[\alpha_{11},\cdots,\alpha_{|\mathcal{I}|1},\cdots,\alpha_{|\mathcal{I}|K}\right]^T$, $\boldsymbol{\beta}\triangleq\left[\beta_1,\cdots,\beta_{|\mathcal{L}^-|}\right]^T$ and $\boldsymbol{\gamma}\triangleq\left[\gamma_{11},\cdots,\gamma_{|\mathcal{I}|1},\gamma_{12}\cdots,\gamma_{|\mathcal{I}|K}\right]^T$. It is immediate that the norm in the Hilbert space reduces to inner products. Hence the objective of the dual (pulling out a minus sign) can be equivalently written as

$$-\min_{\boldsymbol{\theta}}\frac{1}{2}\sum_{i,j}\theta_i\langle\phi(\boldsymbol{x}_i,y_i,i),\phi(\boldsymbol{x}_j,y_j,j)\rangle\theta_j-\sum_i\theta_i \tag{4}$$

in which the so called kernel trick could be used for direct computation of the inner product without the need to compute the explicit feature mapping $\phi(\cdot)$, i.e.,

$$\langle\phi(\boldsymbol{x},y,i),\phi(\boldsymbol{x}',y',j)\rangle=\kappa_{(iy)(jy')}(\boldsymbol{x},\boldsymbol{x}') \tag{5}$$

For a more compact form of the dual, let us further define a matrix $\boldsymbol{Q}$ with elements $\boldsymbol{Q}_{(iy)(jy')}=\kappa_{(iy)(jy')}(\boldsymbol{x},\boldsymbol{x}')$, a column vector of all hidden variables $\boldsymbol{\lambda}\triangleq\{\boldsymbol{\eta}_1;\cdots;\boldsymbol{\eta}_{|\mathcal{L}_H^-|},\boldsymbol{\zeta}_1,\cdots,\boldsymbol{\zeta}_{|\mathcal{U}|}\}$, and an augmented label vector (with dummy copies of unlabeled set) as

$$\widetilde{\boldsymbol{y}}=[\underbrace{1,\cdots,1}_{\mathcal{L}^+},\underbrace{-1,\cdots,-1}_{\mathcal{L}_H^-},\underbrace{-1,\cdots,-1}_{\mathcal{U}^-},\underbrace{-1,\cdots,-1}_{\mathcal{L}^-},\underbrace{1,\cdots,1}_{\mathcal{U}^+}]^T$$

together with a matrix encapsulated inequality constraints, the (negative) inner optimization becomes

$$\begin{aligned}&\min_{\boldsymbol{\theta}}\mathcal{J}(\boldsymbol{\theta})=\frac{1}{2}\boldsymbol{\theta}^T\boldsymbol{Q}\boldsymbol{\theta}-\boldsymbol{e}^T\boldsymbol{\theta} \\ &\text{subject to}\begin{cases}\boldsymbol{C}^\theta\boldsymbol{\theta}\leq\boldsymbol{C}^\lambda\boldsymbol{\lambda}+\boldsymbol{C}^0 \\ \widetilde{\boldsymbol{y}}^T\boldsymbol{\theta}=0.\end{cases}\end{aligned} \quad \textbf{(OPT\_D)}$$

where $\boldsymbol{C}^\theta,\boldsymbol{C}^\lambda,\boldsymbol{C}^0$ are constant matrices with $K|\mathcal{L}_H^-|+(K+1)|\mathcal{U}|$ rows. Similar to other kernel methods in machine learning, HS^{3}M is restricted to Mercer kernels, thence $\boldsymbol{Q}$ is positive definite, and **(OPT_D)** is in a convex quadratic program. The learning objective **ORIG** proposed in section II now becomes

$$\max_{\boldsymbol{\lambda}\in\Lambda}\min_{\theta\in\Theta(\boldsymbol{\lambda})}\mathcal{J}(\boldsymbol{\theta}) \tag{6}$$

Next the inner optimization of (6) is analyzed in some depth, and establish the theoretical foundations of the novel global optimization algorithm.

IV. Structures of the Dual Optimality and a Novel Global Optimization Method

First of all, it is helpful to consider the inner optimization (**OPT_D**) as if "parameterized" by the outer optimization variable $\boldsymbol{\lambda}$. Thus solving it with some fixed $\boldsymbol{\lambda}$, one can get the optimal solution as a function of the "parameters". Denote the optima as $\boldsymbol{\theta}^*(\boldsymbol{\lambda})$ to emphasize this dependence, then the original learning objective becomes $\min_{\boldsymbol{\lambda}\in\Lambda}\mathcal{J}\left(\boldsymbol{\theta}^*(\boldsymbol{\lambda})\right)$.

Two important properties of $\mathcal{J}\left(\boldsymbol{\theta}^*(\boldsymbol{\lambda})\right)$ will be characterized: (1) Locally (in a well defined neighborhood called critical regions) $\boldsymbol{\theta}^*(\boldsymbol{\lambda})$ has an explicit form. (2) Globally the inner optimal objective $\mathcal{J}\left(\boldsymbol{\theta}^*(\boldsymbol{\lambda})\right)$ is convex piece-wise quadratic in λ. These two observations serve as the underpinning for the new optimization algorithm.

The problem of analyzing the dependence between optimal solution and involved parameters have been previously addressed in operational research and our communities, with

the terminology Parametric Programming (PP) or Sensitivity Analysis (SA). In particular, the study of Parametric Quadratic Programming (PQP) can be dated back to [31], and following the pioneer work [32], it has been widely applied for model predictive control problems [33][34]. However, to the best of our knowledge, PQP has not yet been explored for machine learning problems. More importantly, technical difficulties arise because existing results usually rely on the so called Linear Independence Constraint Qualification (LICQ) assumption, which in the current case of (**OPT_D**) cannot be satisfied due to the existence of the equality constraint, $\widetilde{\boldsymbol{y}}^T\boldsymbol{\theta} = 0$. In fact, in the jargon of PP or SA, the problem at hand corresponds to a degenerate case for which existing solution is still lacking.

In subsequent parts, we will bridge the gap by firstly replacing the LICQ condition with a sample partition property, and then show that the local explicit form of $\boldsymbol{\theta}^*(\boldsymbol{\lambda})$ can still be obtained under a mild condition. But before that, it is useful to define the following terms.

Definition 1. ***(Active Constraint)*** *Assume that an optimal solution of (**OPT_D**) has been obtained as $\boldsymbol{\theta}^*(\boldsymbol{\lambda})$. Then the i^{th} row of the constraints is said to be active at $\boldsymbol{\lambda}$, if $\boldsymbol{C}_i^\theta\boldsymbol{\theta}^*(\lambda) = \boldsymbol{C}_i^\lambda\boldsymbol{\lambda} + \boldsymbol{C}_i^0$, and inactive if $\boldsymbol{C}_i^\theta\boldsymbol{\theta}^*(\lambda) < \boldsymbol{C}_i^\lambda\boldsymbol{\lambda} + \boldsymbol{C}_i^0$. Let index set of all active inequality constraints i be denoted by $\mathcal{A}$, and all inactive inequality constraints by $\mathcal{A}^C$. $\boldsymbol{C}_\mathcal{A}$ denotes the $\boldsymbol{C}$ matrix with only rows that correspond to the active constraints, and $\boldsymbol{C}_{\mathcal{A}^C}$ as the matrix with only rows that correspond to inactive constraints.*

Definition 2. ***Samples Partition*** *: Based on the value of $\boldsymbol{\theta}^*$, i.e., $\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*, \boldsymbol{\gamma}^*$ at optimal,*

- *A j^{th} sample in $\mathcal{L}^+ \cup \mathcal{L}^- \cup \mathcal{U}^+$ is called:*
  Support vector if $\alpha_{jk} > 0$, $\beta_j > 0$ or $\gamma_{jk} > 0$ respectively. Unbounded support vector if in addition $\sum_k \alpha_{jk} < c_1$, $\beta_j < c_{21}$, or $\sum_k \gamma_{jk} < c_3$, and bounded support vector if the upper bound (equality) is reached.
- *A j'^{th} sample in $\mathcal{L}_H^+ \cup \mathcal{U}^-$ is called:*
  Support vector of subgroup k if $\alpha_{j'k} > 0$. Unbounded support vector of subgroup k if in addition $\alpha_{j'k} < c_{22}$ for $j' \in \mathcal{L}_H^-$ or $\alpha_{j'k} < c_3$ for $j' \in \mathcal{U}^-$, and bounded support vector if the upper bound (equality) is reached.

This definition is extended from classic binary Support Vector Mathine (SVM) and has very similar geometric interpretations. Next, we define a term based on the characteristics of the sample partition, which will serve as a sufficient condition for the existence of the parametric solution of (**OPT_D**).

Definition 3. ***(Qualified Solution)*** *We say that the solution of the dual problem is Qualified if its corresponding sample partition contains at least one unbounded support vector in both $\mathcal{L}^+ \cup \mathcal{L}^- \cup \mathcal{U}^+$ and $\mathcal{L}_H^+ \cup \mathcal{U}^-$*

It is worth noting that requiring qualified solution is a very mild condition. Indeed since the unbounded support vectors are essentially the sample points that lie on the decision boundaries that construct the classifiers (and its interception). In order to have meaningful classification in practice this condition is necessary and is expected to be satisfied with even a few samples. The following theorem characterizes the solution structure and provides explicit forms.

Theorem 1. *Assume that the solution of (**OPT_D**) is qualified and induces a set of active and inactive constraints $\mathcal{A}$ and $\mathcal{A}^\mathcal{C}$, respectively. Denote the composed matrices*

$$\boldsymbol{H} \triangleq \frac{\boldsymbol{Q}^{-1}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\boldsymbol{Q}^{-1}}{\widetilde{\boldsymbol{y}}^T\boldsymbol{Q}^{-1}\widetilde{\boldsymbol{y}}} - \boldsymbol{Q}^{-1}$$

$$\boldsymbol{T} \triangleq \boldsymbol{H}(\boldsymbol{C}_\mathcal{A}^\theta)^T; \quad \boldsymbol{P} \triangleq \boldsymbol{C}_\mathcal{A}^\theta\boldsymbol{H}(\boldsymbol{C}_\mathcal{A}^\theta)^T. \tag{7}$$

and the vector $\widetilde{\boldsymbol{e}} \triangleq \boldsymbol{C}_\mathcal{A}^\theta\boldsymbol{H}\boldsymbol{e}$. Then we have

1. *The matrix $\boldsymbol{H}$ is symmetric negative semi-definite, and $\boldsymbol{P}$ is symmetric strictly negative definite hence is invertible.*
2. *The optimal solution is a continuous piecewise affine function of $\boldsymbol{\lambda}$. And in the critical region defined by*

$$\begin{cases} \boldsymbol{P}^{-1}(\boldsymbol{C}_\mathcal{A}^\lambda\boldsymbol{\lambda} + \boldsymbol{C}_\mathcal{A}^0 + \widetilde{\boldsymbol{e}}) \geq 0 \\ \boldsymbol{C}_{\mathcal{A}^C}^\lambda\boldsymbol{\lambda} + \boldsymbol{C}_{\mathcal{A}^C}^0 - \boldsymbol{C}_{\mathcal{A}^C}^\theta\boldsymbol{T}\boldsymbol{P}^{-1}(\boldsymbol{C}_\mathcal{A}^\lambda\boldsymbol{\lambda} + \boldsymbol{C}_\mathcal{A}^0 + \widetilde{\boldsymbol{e}}) \geq 0 \end{cases} \tag{8}$$

   *the optimal solution $\boldsymbol{\theta}^*$ of (**OPT_D**) admits a closed form*

$$\boldsymbol{\theta}^*(\boldsymbol{\lambda}) = \boldsymbol{T}\boldsymbol{P}^{-1}(\boldsymbol{C}_\mathcal{A}^\lambda\boldsymbol{\lambda} + \boldsymbol{C}_\mathcal{A}^0 + \widetilde{\boldsymbol{e}}) \tag{9}$$

3. *The optimal objective is a continuous piece-wise quadratic (**PWQ**) function of $\boldsymbol{\lambda}$.*

In essence the theorem indicates that each time the inner optimization (**OPT_D**) is solved, full information (closed form solution) in a well-defined neighborhood (critical region) can be retrieved as a function of outer optimization variables, i.e., those newly introduced "hidden variables".

Besides the local explicitness result, the next theorem describes the overall geometric structure of the optimality, showing that globally the optimal objective is convex in $\boldsymbol{\lambda}$, which inspires the proposed learning algorithm for HS^{3}M.

Theorem 2. *Still assuming qualified solution,*

1. *The dual optimization has finite number of polyhedron critical regions $CR_1, \cdots, CR_{N_r}$ which constitute a **partition** of the feasible set of $\boldsymbol{\lambda}$, i.e., each feasible $\boldsymbol{\lambda}$ belongs to one and only one critical region.*
2. *The optimal objective $\mathcal{J}(\boldsymbol{\theta}^*(\boldsymbol{\lambda}))$ is a **convex** PWQ function of $\boldsymbol{\lambda}$, and is almost everywhere differentiable.*

With the local explicit solution and global convex PWQ structure of $\mathcal{J}(\boldsymbol{\theta}^*(\boldsymbol{\lambda}))$ revealed by the theoretical analysis, the learning problem is reduced to *maximizing (minimizing) a non-smooth but convex (concave) function* in the space $\boldsymbol{\lambda} \in \Lambda$. Concave minimization is well-known to be NP-hard. Despite the hardness, there exist several global optimality conditions. In this work one of these conditions [35] and a level set idea are adopted to establish a an optimization algorithm that is able to approach global optima. The derivation and convergence analysis of the algorithm is long and technical, hence is postponed to Appendix for interested readers. It's worth pointing out that the proposed optimization strategy can be extended to a much broader class of non-convex machine learning problems, and the parametric analysis done in the section complements prior studies on PQP. These two points constitutes the theoretical contribution of this paper.

V. Evaluation

The authors are collaborating in the "Micro-Synchrophasor for Power Distribution Networks" project [2] to develop a network of μ-PMU devices (μ-Pnet) at the distribution level. In this paper, data from μ-PMUs are used to validate the proposed method. Each μ-PMU provides 120 samples per second for three-phase voltage and current magnitude and phase angle [3],[4].

A. Data Collection and Feature Selection

The raw data sets collected from the μ-PMU measurement are multi-stream time series, which are transmitted and stored with the sMAP protocol [36]. Notation-wise, raw data is written as $\{X_1, \cdots, X_T\}$, where each X_t is a $M \times C$ dimensional vector. In the current experiment, $M = 5$ is the number of μ-PMUs and $C = 12$ is the number of channels of each μ-PMU. Because the raw data is in millisecond's resolution and almost all practical events happen at a larger time scale, one can safely use a sliding window to extract useful information. The window size L is set to 12 according to the time scale of the event of interest. We denote $w_t^i \triangleq \{x_t^i, \cdots, x_{t+L}^i\}$ as the t^{th} window of stream i. For the purpose of detecting different types of events, miscellaneous single stream and inter-stream feature extraction are performed. A summary of computed feature are given in Table I.

With the presented feature extraction procedure, a total number of 312 features have been obtained. However, some of them may be redundant as there are significant similarities among extracted features, for example, when the three phases are balanced, their single stream mean, variation, etc., are almost the same. From a machine learning point of view, adding redundant features does not help event detection, but instead introduces extra noise and cause computational difficulties. In this work, we adopt a method developed in [37], called Minimum-redundancy-maximum-relevance (mRMR). The procedure uses mutual information as the metric of goodness of a feature set, and resolve the trade-off between relevancy and redundancy. For each event, mRMR is conducted to choose 20 most informative features [38][39]. Also note that all numerical experiments in the following are conducted on a workstation having dual Xeon X5687 CPUs and 72GB memory.

B. Performance of HS^{3}M

The overall task is not only identifying the occurrence of events versus stable state, i.e., the binary classification, but also distinguishing 4 types of events including Voltage Disturbance (VD) and Voltage Sag (VS), Motor Start (MS), High Impedance fault (HI), i.e., the multi-class classification. The training set contains about 40000 μPMU records with detailed labels (completely labeled data). The testing data set contains the similar events and has around 30000 data points, but is collected at a different time. For the training of HS^{3}M which enables the inclusion of partial knowledge, another 36000 partially labeled data and 108000 unlabeled data are also used (the effect of the size of these data sets will be discussed later).

TABLE I: Extracted Candidate Features

Single Stream	Statistics	$\text{mean}(w_t^i), \text{var}(w_t^i), \text{range}(w_t^i)$ $\text{median}(w_t^i), \text{entropy}(w_t^i), \text{hist}(w_t^i)$
	Difference	$u_t^i = \text{Diff}(x_t^i)$; Statistics
	Transformation	$\text{fft}(w_t^i), \text{wavelet}(w_t^i)$
Inter Stream	Deviation	$x^i - x^j \quad \forall i, \forall j \in \mathcal{N}(i)$
	Correlation	$corr(x^i, x^j) \quad \forall i, \forall j \in \mathcal{N}(i)$

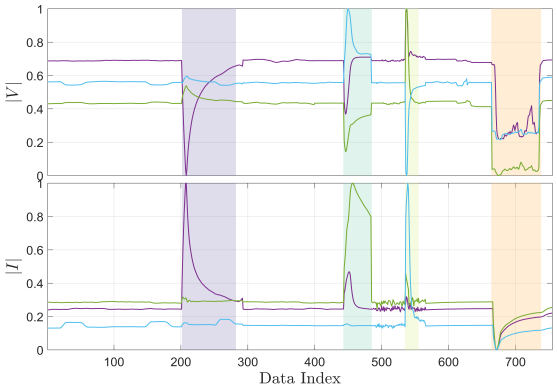


Fig. 4: Detection results over Time. Note that some periods of stable state are shrunk for visualization purpose.

The performance of HS^{3}M is compared with other popular multi-class event detection methods, including cost sensitive versions of Ada Boost, Decision Tree, and Gaussian Process Classification. The hyper-parameters of those models, e.g., the cost balance coefficient, are chosen with 10-folds cross validation (CV). A brief introduction of those methods and their implementation details are listed below:

- Decision tree method: It create a tree-like model that predicts the value of a target variable by learning simple decision rules inferred from the data features. The conditional inference tree algorithm is implemented in this work, which uses multiple significance tests to grow the tree. For cross validation, the maximum tree depth is varied from 10 to 50 by factors of 10, and the splitting threshold is varied from 0.1 to 0.9 with 0.1 intervals.
- Ada Boosting Method: Boosting generates a prediction model by combining many weak classifiers into a stronger classification committee. The AdaBoost procedure is implemented to combine basic tree classifiers for ensemble learning. We vary the maximum tree depth from 10 to 50 by factors of ten. The number of boosting iterations range from 100 to 500 by a step size of 50.
- Gaussian Process Classification (GPC): Instead of directly parameterizing a latent function for classification, GPC models it with a generic Gaussian process. The posterior of the process is updated with training data set, and is squashed through a logistic function for classification. The GPC is implemented with the help of kernlab package [2], which includes several approximation algorithms for acceleration. The radial basis kernel is used with kernel width tuned from 2^{-5} to 2^4.

The binary detection results are shown in Figure 4 and its performance are listed in Table II together with other

[2]http://www.jstatsoft.org/v11/i09/

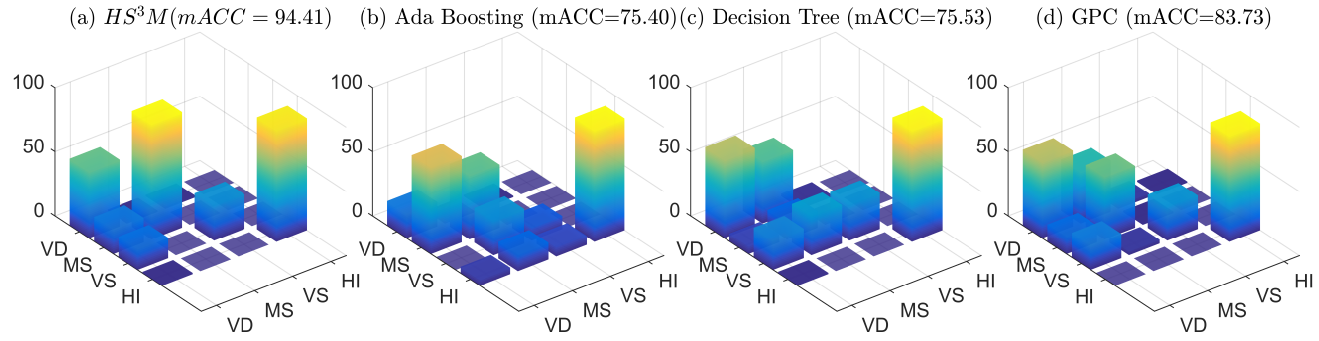


Fig. 5: Confusion Matrix for different methods. Diagonal terms are correct identifications and off-diagonal ones are misclassifications. mACC for multi-class detection accuracy. Note that the class of stable state is not included for better visualization.

methods under comparison. We observe that HS^{3}M performs extremely well in distinguishing stable state and events, with 99.13% binary accuracy (bACC), only 2.33% false positive (false alarm) rate and 0% false negative (miss detection) rate. Compared to other methods, the proposed HS^{3}M outperforms the rest, by at least 9% of bACC and 10% of false positive with respect to the runner-up. The computational cost in terms of training and testing (or prediction) time/memory usage are also listed in Table II. It appears that HS^{3}M requires longer time and about median memory usage in the training phase. This is expected since HS^{3}M is a more comprehensive method incorporating partial information. On the other hand, the testing time/memory usage of HS^{3}M are one of the shortest/smallest. This is due to the solution sparsity (Definition 2) of the HS^{3}M classifier. In practice, testing cost is of major concern because event prediction should be done in real time on distributed systems, while training can be performed "offline" on powerful computers. In this regards, the proposed HS^{3}M is promising also when computational cost is a concern.

To compare the performance on distinguishing event types, the confusion matrices (contingency table) for all methods are shown in Figure 5. Each row of the sub-figure represents the samples in predicted class while each column represents the samples in actual (true) class. The overall multi-class accuracy (mACC) is summarized in the title of each sub-figure. We see from the confusion matrices that significant improvement is achieved: HS^{3}M provides 94.41% mACC, outperforming the best of the other methods by around 11%, while the classic tree based method only yields 75% classification accuracy. Moreover, HS^{3}M gives improvements in differentiating all event types, especially VS, MS, and HI with an accuracy at least 90%. The only issue is that it tends to confuse VD with VS, which is somewhat expected as the criteria for distinguishing VD and VS events are thresholding on the voltage magnitude. In short, the results justified the effectiveness of the proposed HS^{3}M, as well as the idea of incorporating partial information for event detection.

C. Effect of Partial Information

Last but not least, the benefit of including additional partially labeled data and unlabeled data is investigated. To do this, HS^{3}M is evaluated with 0, 7200, 21600 and 36000

TABLE II: Comparison of (Binary) Detection Performance and Computational Cost. bACC for binary Accuracy.

Method	HS^{3}M	Ada Boost.	Decision Tree	GPC
bACC (%)	99.13	90.11	89.20	92.73
False Positive (%)	2.33	12.3	16.87	10.1
False Negative (%)	0.0	5.32	0.01	2.31
Time Train (min)	35.7	17.4	11.8	28.6
Time Test (sec)	53.7	206.6	41.2	221.9
Mem. Train (MB)	193	166	59	299
Mem. Test (MB)	0.79	2.91	0.52	292

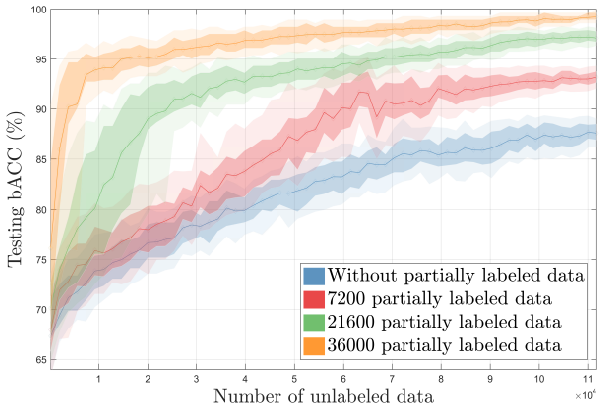


Fig. 6: The incorporation of partially and unlabeled data

partially labeled data samples and 0 - 108000 unlabeled data points. The testing accuracy (bACC) of each test, averaged over 50 random sampled experiments, is plotted in Figure 6 (diamond line), together with the 0.75 and 0.90 confidence intervals (shaded area). Note that when no partially labeled data is included (blue line), HS^{3}M reduces to the semi-supervised (kernelized) support vector machine [40]. In the case where unlabeled data is not incorporated (the beginning of each line), HS^{3}M can be viewed as a multi-class variations of the consensus learning recently proposed in [41]. In general, it is observed that the performance improves as more partially labeled data and unlabeled data are added, while the improvements exhibit a "diminishing return" property, i.e., the marginal benefit of including more and more partially/unlabeled data is decreasing. Besides, it appears clearly that HS^{3}M, by leveraging both source of information, signif-

icantly outperforms previous semi-supervised and consensus learning.

VI. Conclusion and Future Work

In summary, with the help of high resolution μ-PMU measurement, a pure data-driven framework is designed for distribution network event detection by incorporating all types of accessible labeled, partially labeled and unlabeled data. More importantly, we developed a new optimization algorithm, by revealing the locally explicit and globally convex property of the dual solution. The experimental results on a real-world μ-PMU dataset justified the effectiveness of the proposed HS^{3}M and the PDOP global optimization method, as well as the benefit of incorporating partial knowledge for event detection in power distribution network.

For future work, we will implement HS^{3}M methods for a large volume of μ-PMUs data streams, as well as developing online learning HS^{3}M method. More recorded events will be used to train our algorithm and perform feature selection. More importantly, the spatial-temporal characters of large scale events in power systems will be investigated to detect possible cascading effect.

Appendix

A. Derivation of the PDOP algorithm

With a slight abuse of notation, we omit the intermediate variable and denote the non-smooth convex function $\mathcal{J}(\boldsymbol{\theta}^*(\boldsymbol{\lambda}))$ by $\mathcal{J}(\boldsymbol{\lambda})$. From previous discussion, we consider the following non-smooth convex maximization:

$$\max_{\boldsymbol{\lambda}\in\mathbb{P}} \mathcal{J}(\boldsymbol{\lambda}) \tag{10}$$

We first provide a global optimality condition for maximizing non-smooth convex function, then we propose a corresponding optimization algorithm that approaches global solution with level set augmentation.

The global optimality condition for maximizing smooth convex function, in particular convex quadratic functions, has been studied before [35]. However, none of them consider non-smooth piecewise defined functions. In this work, we extend Strekalovsky's condition to non-smooth cases. First of all, the notion of level set is defined as the set of variables that produces the same function values, i.e.,

Definition 4. *The level set of function $\mathcal{J}$ at $\boldsymbol{\lambda}$ is defined by*

$$E_{\mathcal{J}(\boldsymbol{\lambda})} = \{q \in \mathbb{R}^n \mid \mathcal{J}(\boldsymbol{\lambda}) = \mathcal{J}(\boldsymbol{\lambda})\}$$

The following result gives sufficient and necessary condition for a point $\boldsymbol{\lambda}^*$ to be the global maximizer of non-smooth convex function $\mathcal{J}(\boldsymbol{\lambda})$

Theorem 3. *Assume that $\mathcal{J}$ is not constant, then $\boldsymbol{\lambda}^*$ is a global optimal solution of problem (10) if and only if for all $\boldsymbol{\lambda} \in \mathbb{P}$, $\boldsymbol{q} \in E_{\mathcal{J}(\boldsymbol{p}^*)}$, $g(\boldsymbol{q}) \in \partial\mathcal{J}(\boldsymbol{q})$*

$$(\boldsymbol{\lambda} - \boldsymbol{q})^T g(\boldsymbol{q}) \le 0 \tag{11}$$

where $\partial\mathcal{J}(\boldsymbol{q})$ is the set of subgradients of $\mathcal{J}$ at $\boldsymbol{\lambda}$.

By virtue of Theorem 3, we can check the optimality of any point $\boldsymbol{\lambda}$ by solving

$$\Delta(\boldsymbol{\lambda}) \triangleq \max_{\substack{\boldsymbol{q}\in E_{\mathcal{J}(\boldsymbol{\lambda})},\ \boldsymbol{\lambda}'\in\mathbb{P} \\ g(\boldsymbol{q})\in\partial\mathcal{J}(\boldsymbol{q})}} (\boldsymbol{\lambda}' - \boldsymbol{q})^T g(\boldsymbol{q}) \tag{12}$$

We call the above maximization auxiliary problem at $\boldsymbol{\lambda}$. This seems to be a hard problem because (1) it is bilinear in decision variables and (2) usually the level set $E_{\mathcal{J}(\boldsymbol{\lambda})}$ cannot be calculated explicitly. Next we study solution method for (12) based on approximating the level set with a collection of representative points. To begin with, let us formally define the notion

Definition 5. *Given a user specified approximation degree m, the approximation level set for $E_{\mathcal{J}(\boldsymbol{\lambda})}$ is defined by*

$$A^m_{\boldsymbol{\lambda}} = \left\{\boldsymbol{q}^1, \boldsymbol{q}^2, \cdots, \boldsymbol{q}^m \mid \boldsymbol{q}^i \in E_{\mathcal{J}(\boldsymbol{\lambda})}\ i = 1, 2, \cdots, m\right\}$$

Now consider solving the auxiliary problem approximately by replacing $E_{\mathcal{J}(\boldsymbol{\lambda})}$ with $A^m_{\boldsymbol{\lambda}}$, we obtain that for each $\boldsymbol{q}^i$, problem (12) becomes

$$\max_{\boldsymbol{\lambda}\in\mathbb{P},\ g(\boldsymbol{q}^i)\in\partial\mathcal{J}(\boldsymbol{q}^i)} (\boldsymbol{\lambda} - \boldsymbol{q}^i)^T g(\boldsymbol{q}^i) \tag{13}$$

Since $\mathcal{J}(\boldsymbol{\lambda})$ is almost everywhere differentiable, in most cases $g(\boldsymbol{q}^i)$ is unique and equals to the gradient $\nabla\mathcal{J}(\boldsymbol{q}^i)$. Then the auxiliary problem is a simple linear program. In the cases when $\boldsymbol{q}^i$ is on the boundary of critical regions, $\partial\mathcal{J}(\boldsymbol{q}^i)$ becomes a convex set and the auxiliary problem becomes a bilinear program. Generally bilinear program is hard, but fortunately (12) has disjoint feasible sets and one can easily check that it is equivalent to

$$\max_{\boldsymbol{\lambda}\in\mathbb{P}} \left\{ \max_{g(\boldsymbol{q}^i)\in V(\partial\mathcal{J}(\boldsymbol{q}^i))} (\boldsymbol{\lambda} - \boldsymbol{q}^i)^T g(\boldsymbol{q}^i) \right\} \tag{14}$$

which uses the fact that the optimal solution to (13) must be on the vertex of the feasible polyhedron, i.e., $g^*(\boldsymbol{q}^i)$ must be one of the vertices of $\partial\mathcal{J}(\boldsymbol{q}^i)$. Moreover, since $\mathcal{J}(\boldsymbol{\lambda})$ can be viewed as a pointwise maximum, the subgradient is the convex hull of one side derivatives of the neighboring critical regions. The above analysis suggests an enumerative method for (12) by solving a set of linear programs each with an element in $A^m_{\boldsymbol{\lambda}}$ and a vertex of $\partial\mathcal{J}(\boldsymbol{q}^i)$.

Having solved the approximate auxiliary problem, we can immediately determine if an improvement can be made with current approximate level set. Let $\{(\boldsymbol{u}^i, \boldsymbol{s}^i),\ i = 1, \cdots, m\}$ be the solution of (13) on the approximate level set $A^m_{\boldsymbol{\lambda}}$, i.e.,

$$(\boldsymbol{u}^i - \boldsymbol{q}^i)^T \boldsymbol{s}^i = \max_{\boldsymbol{\lambda}\in\mathbb{P},\ g(\boldsymbol{q}^i)\in V(\partial\mathcal{J}(\boldsymbol{q}^i))} (\boldsymbol{\lambda} - \boldsymbol{q}^i)^T g(\boldsymbol{q}^i) \tag{15}$$

and define

$$\Delta(A^m_{\boldsymbol{\lambda}}) = \max_{i=1,\cdots,m} (\boldsymbol{u}^i - \boldsymbol{q}^i)^T \boldsymbol{s}^i \tag{16}$$

Then with the convexity of $\mathcal{J}$ we have

Proposition 2. *For any $\boldsymbol{\lambda} \in \mathbb{P}$, if there exist $\boldsymbol{q}^i \in A^m_{\boldsymbol{\lambda}}$, $g(\boldsymbol{q}^i) \in V(\partial\mathcal{J}(\boldsymbol{q}^i))$, and $\boldsymbol{u}^i$ defined in (15), such that $(\boldsymbol{u}^i - \boldsymbol{q}^i)^T g(\boldsymbol{q}^i) > 0$, then $\mathcal{J}(\boldsymbol{u}^i) > \mathcal{J}(\boldsymbol{\lambda})$.*

Now the remaining problem is to construct approximate level set given current point $\boldsymbol{\lambda}$ and the degree m. The following

Algorithm 1 Parametric Dual Optimization Procedure

Choose $\boldsymbol{\lambda}^{(0)} \in \Lambda$; set $k = 0$; compute $\boldsymbol{\lambda}_*$ with subgradient descent.
while $k \leq$ iter_max **do**
  Starting from $\boldsymbol{\lambda}^{(k)}$, find local minimizer $\boldsymbol{r}^{(k)} \in \Lambda$ with existing methods.
  Construct approximate level set $A^m_{\boldsymbol{r}^{(k)}}$ at $\boldsymbol{r}^{(k)}$
  for $\boldsymbol{q}^i \in A^m_{\boldsymbol{r}^{(k)}}$ **do**
    for $\boldsymbol{g}^j \in V(\partial\mathcal{J}(\boldsymbol{q}^i))$ **do**
      Solve $\boldsymbol{u}_{ij} = \text{argmax}_{\boldsymbol{\lambda}\in\Lambda}\ (\boldsymbol{\lambda} - \boldsymbol{q}^i)^T\boldsymbol{g}^j$ #linear programming
    end for
    Let $j^* = \text{maxIndex}_j\{\boldsymbol{u}_{ij}\}$; $(\boldsymbol{u}^i, \boldsymbol{s}^i) = (\boldsymbol{u}_{ij^*}, \boldsymbol{g}^{j^*})$;
  end for
  Let $i^* = \text{maxIndex}_{i=1,\cdots,m}\{(\boldsymbol{u}^i - \boldsymbol{q}^i)^T\boldsymbol{s}^i\}$;
  $\Delta(A^m_{\boldsymbol{\lambda}}) = (\boldsymbol{u}^{i^*} - \boldsymbol{q}^{i^*})^T\boldsymbol{s}^{i^*}$; $\boldsymbol{u}^{(k)} = \boldsymbol{u}^{i^*}$;
  if $\Delta(A^m_{\boldsymbol{\lambda}}) > 0$ **then**
    Set $\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{u}^{(k)}$; $k = k + 1$; # improvement found
  else
    Terminate and output $\boldsymbol{\lambda}^{(k)}$; # Global optimality condition
  end if
  Collecting explored critical region and explicit forms in Theorem 1.
end while

results shows that this is theoretically possible with the help of a global minimizer.

Lemma 1. *Let the global **minimizer** of $\mathcal{J}(\boldsymbol{\lambda})$ be $\boldsymbol{\lambda}_*$, then for any $\boldsymbol{h} \in \mathbb{R}^n$ and $\boldsymbol{\lambda} \neq \boldsymbol{\lambda}_*$, there exist a unique positive scalar γ, such that $\boldsymbol{\lambda}_* + \gamma\boldsymbol{h} \in E_{\mathcal{J}(\boldsymbol{\lambda})}$.*

With this guarantee, we write approximate level set by

$$A^m_{\boldsymbol{\lambda}} = \left\{\boldsymbol{q}^1, \boldsymbol{q}^2, \cdots, \boldsymbol{q}^m \mid \boldsymbol{q}^i = \boldsymbol{\lambda}_* + \gamma_i\boldsymbol{h}^i \in E_{\mathcal{J}(\boldsymbol{\lambda})}\right\} \quad (17)$$

To explore directions for improvement, a natural choice of $\boldsymbol{h}$ is a set of orthogonal basis. Specifically, we could start with a random $\boldsymbol{h}^1$ and use Gram-Schmidt algorithm to extend it to m orthogonal basis. For each $\boldsymbol{h}^i$, the corresponding γ_i is found by solving:

$$\Phi(\gamma_i) \triangleq \mathcal{J}(\boldsymbol{\lambda}_* + \gamma_i\boldsymbol{h}^i) - \mathcal{J}(\boldsymbol{\lambda}) = 0 \quad (18)$$

As stated in Lemma 1, the above function has a unique root, which can be computed efficiently with line searching method such as the Bisection algorithm. While another problem is to solve $\boldsymbol{\lambda}_* = \text{argmin}\mathcal{J}(\boldsymbol{\lambda})$, which is a convex minimization problem. We adopt a sub-gradient descent method since in each critical region, the gradient can be calculated explicitly with Theorem 1. By further considering the convex PWQ structure of the problem, we have

Lemma 2. *Let $\sup_{\boldsymbol{\lambda}} ||\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}|| = B$, and the Lipschitz constant of $\mathcal{J}$ be G, then sub-gradient descent with iteration T and optimal step size $\tau_i = B/G\sqrt{T}$ converges to global minimum within $O\left(n/\sqrt{T}\right)$. To be specific, let $\mathcal{J}_*$ be the global minimum then*

$$\mathcal{J}(\boldsymbol{\theta}^*(\boldsymbol{\lambda}^{(T)}_{best})) - \mathcal{J}_* \leq \frac{BG}{\sqrt{T}} \leq O\left(\frac{n}{\sqrt{T}}\right), \qquad \textit{where}$$

$$\mathcal{J}(\boldsymbol{\theta}^*(\boldsymbol{\lambda}^{(T)}_{best})) \triangleq \min\left\{\mathcal{J}(\boldsymbol{\alpha}^*(\boldsymbol{\lambda}^{(1)})), \cdots, \mathcal{J}(\boldsymbol{\theta}^*(\boldsymbol{\lambda}^{(T)}))\right\}$$

The overall PDM procedure for $\max_{\boldsymbol{\lambda}\in\mathbb{P}}\mathcal{J}(\boldsymbol{\lambda})$ is summarized in Algorithm 1. Given current solution $\boldsymbol{\lambda}^{(k)}$, the algorithm first tries to improve it with existing methods such as AO, CCCP, and SGD. After finding a local minimizer $\boldsymbol{r}^{(k)}$, approximate level set $A^m_{\boldsymbol{r}^{(k)}}$ is obtained by solving (18) and constructing (17). With $A^m_{\boldsymbol{r}^{(k)}}$ and vertices of current sub-gradient set, a series of linear programming is solved to pick up vector $\boldsymbol{u}^{(k)}$ and subgradient $\boldsymbol{s}$ that maximize condition (11) of Theorem 3. If this maximal value, i.e., $\Delta(A^m_{\boldsymbol{\lambda}})$, is greater than 0, then by Proposition 2, $\boldsymbol{u}^{(k)}$ must be strict improvement of $\boldsymbol{r}^{(k)}$. The algorithm iterate until no improvement could be found at current point with the associated approximate level set. Combining Theorem 3 and Proposition 2, we have

Theorem 4. *Algorithm I generates a sequence $\{\boldsymbol{\lambda}^{(1)}, \cdots, \boldsymbol{\lambda}^{(k)}, \cdots\}$ that converges to the global maximizer of $\mathcal{J}(\boldsymbol{\lambda})$ in a finite number of steps or finds an approximate maximizer at user specified degree m.*

REFERENCES

[1] Alexandra Von Meier, David Culler, Alex McEachern, and Reza Arghandeh. Micro-synchrophasors for distribution systems. In *Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5. IEEE, 2014.

[2] Lawrence Jones. Every moment counts: Synchrophasors for distribution networks with variable resources. In *Renewable Energy Integration*. Academic Press, Boston, 2014.

[3] Alexandra von Meier and Reza Arghandeh. Diagnostic applications for micro-synchrophasor measurements, 2014.

[4] http://www.powersensorsltd.com/PQube3.php. Micro synchrophasor catalog, 2015.

[5] Reza Arghandeh, Martin Gahr, Alexandra von Meier, Guido Cavraro, Monika Ruh, and Goran Andersson. Topology detection in microgrids with micro-synchrophasors. In *Power & Energy Society General Meeting*. IEEE, 2015.

[6] Miles Wen, Reza Arghandeh, Alexandra von Meier, Kameshwar Poolla, and Victor Li. Phase identification in distribution networks with micro-synchrophasors. In *Power & Energy Society General Meeting*. IEEE, 2015.

[7] Luca Schenato, Grazia Barchi, David Macii, Reza Arghandeh, Kameshwar Poolla, and Alexandra Von Meier. Bayesian linear state estimation using smart meters and pmus measurements in distribution grids. In *International Conference on Smart Grid Communications (SmartGridComm)*, pages 572–577. IEEE, 2014.

[8] Mark F McGranaghan and Surya Santoso. Challenges and trends in analyses of electric power quality measurement data. *EURASIP Journal on Advances in Signal Processing*, (1), 2007.

[9] IEEE recommended practice for monitoring electric power quality. *IEEE Std 1159-2009 (Revision of IEEE Std 1159-1995)*, June 2009.

[10] IEEE draft guide for the use of IEEE Std 1641, standard for signal and test definition. *IEEE P1641.1/D3*, Aug 2012.

[11] Alexander. Eigeles. Ieee standard 1459: a long overdue document [power quality]. In *IEEE Technical Conference on Industrial and Commercial Power Systems*, May 2003.

[12] Yihai Zhu, Jun Yan, Yan Lindsay Sun, and Haibo He. Revealing cascading failure vulnerability in power grids using risk-graph. *IEEE Transactions on Parallel and Distributed Systems*, 25(12):3274–3284, 2014.

[13] Yihai Zhu, Jun Yan, Yufei Tang, Yan Lindsay Sun, and Haibo He. Resilience analysis of power grids under the sequential attack. *IEEE Transactions on Information Forensics and Security*, 9(12):2340–2354, 2014.

[14] Yihai Zhu, Jun Yan, Yufei Tang, Yan Lindsay Sun, and Haibo He. Joint substation-transmission line vulnerability assessment against the smart grid. *IEEE Transactions on Information Forensics and Security*, 10(5):1010–1024, 2015.

[15] M Vaiman, Keith Bell, Y Chen, B Chowdhury, I Dobson, P Hines, M Papic, S Miller, and P Zhang. Risk assessment of cascading outages: Methodologies and challenges. *IEEE Transactions on Power Systems*, 27(2):631, 2012.

[16] Sakis Meliopoulos, George Cokkinides, Renke Huang, Evangelos Farantatos, Sungyun Choi, and Yonghee Lee. Wide area dynamic monitoring and stability controls. In *iREP Symposium onBulk Power System Dynamics and Control (iREP)-VIII (iREP)*. IEEE, 2010.

[17] Rolf Isermann. Model-based fault-detection and diagnosis–status and applications. *Annual Reviews in control*, 29(1), 2005.

[18] Borhan M Sanandaji, Eilyan Bitar, Kameshwar Poolla, and Tyrone L Vincent. An abrupt change detection heuristic with applications to cyber data attacks on power systems. In *American Control Conference (ACC)*. IEEE, 2014.

[19] Xuemei Ding, Josiah Poon, Ivan Celanovic, and Alejandro D Dominguez-Garcia. Fault detection and isolation filters for three-phase ac-dc power electronics systems. *IEEE Trans. on Circuits and Systems I: Regular Papers*, 60(4), 2013.

[20] Anselm Schwarte, Frank Kimmidi, and Rolf Isermann. Model-based fault detection of a diesel engine with turbo charger-a case study. In *Fault Detection, Supervision and Safety of Technical Processes 2003 (SAFEPROCESS 2003): A Proceedings Volume from the 5th IFAC Symposium, Washington, DC, USA, 9-11 June 2003*, volume 1. Elsevier, 2004.

[21] Barry Mather. Quasi-static time-series test feeder for PV integration analysis on distribution systems. In *Power and Energy Society General Meeting*. IEEE, 2012.

[22] Joe Qin. Survey on data-driven industrial process monitoring and diagnosis. *Annual Reviews in Control*, 36(2), 2012.

[23] Jae Yeon Baek, Philippe Leray, Anne-Laure Charley, and Costas J Spanos. Real-time inspection system utilizing scatterometry pupil data. *Journal of Micro/Nanolithography, MEMS, and MOEMS*, 13(4), 2014.

[24] Dan Li, Yuxun Zhou, Guoqiang Hu, and Costas J Spanos. Fault detection and diagnosis for building cooling system with a tree-structured learning method. *Energy and Buildings*, 2016.

[25] Vladimir Naumovich Vapnik and Vlamimir Vapnik. *Statistical learning theory*, volume 1. Wiley New York, 1998.

[26] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

[27] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. Unsupervised learning. In *The elements of statistical learning*, pages 485–585. Springer, 2009.

[28] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

[29] Xiaojin Zhu and Andrew B Goldberg. Introduction to semi-supervised learning. *Synthesis lectures on artificial intelligence and machine learning*, 3(1):1–130, 2009.

[30] Christodoulos Floudas. *Nonlinear and mixed-integer optimization: fundamentals and applications*. Oxford University Press, 1995.

[31] John. Boot. On sensitivity analysis in convex quadratic programming problems. *Operations Research*, 11(5), 1963.

[32] Petter Tndel, Tor Arne Johansen, and Alberto Bemporad. An algorithm for multi-parametric quadratic programming and explicit MPC solutions. *Automatica*, 39(3), 2003.

[33] Tor Johansen, Warren Jackson, Robert Schreiber, Petter Tondel, et al. Hardware architecture design for explicit model predictive control. In *American Control Conference*. IEEE, 2006.

[34] Martin Monnigmann and Matthias Jost. Vertex based calculation of explicit mpc laws. In *American Control Conference (ACC)*. IEEE, 2012.

[35] Alexander S Strekalovsky. Global optimality conditions for nonconvex optimization. *Journal of Global Optimization*, 12(4), 1998.

[36] Stephen Dawson-Haggerty, Xiaofan Jiang, Gilman Tolle, Jorge Ortiz, and David Culler. smap: a simple measurement and actuation profile for physical information. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2010.

[37] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(8), 2005.

[38] Yuxun Zhou, Reza Arghandeh, Ioannis Konstantakopoulos, Shayaan Abdullah, and Costas J Spanos. Data-driven event detection with partial knowledge: A hidden structure semi-supervised learning method. In *American Control Conference (ACC)*, pages 5962–5968. IEEE, 2016.

[39] Yuxun Zhou, Reza Arghandeh, Ioannis Konstantakopoulos, Shayaan Abdullah, Alexandra von Meier, and Costas J Spanos. Abnormal event detection with high resolution micro-pmu data. In *Power Systems Computation Conference (PSCC)*, pages 1–7. IEEE, 2016.

[40] Xilan Tian, Gilles Gasso, and Stéphane Canu. A multiple kernel framework for inductive semi-supervised svm learning. *Neurocomputing*, 90, 2012.

[41] Yuxun Zhou, Ninghang Hu, and Costas J Spanos. Veto-consensus multiple kernel learning. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[42] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Trans. on Intelligent Systems and Technology*, 2, 2011.

ACKNOWLEDGMENT

The authors would like to thank Prof. Alexandra von Meier for her helpful advice and insights. This research is sponsored in part by the U.S. NSF award 1640587 and by the Republic of Singapore's National Research Foundation through a grant to the Berkeley Education Alliance for Research in Singapore (BEARS) for the Singapore-Berkeley Building Efficiency and Sustainability in the Tropics (SinBerBEST) Program.

Yuxun Zhou is currently a Ph.D candidate at Department of EECS, UC Berkeley. He obtained a Diplome d'Ingenieur in Applied Mathematics from Ecole Centrale Paris in 2012, and a B.S. degree in Electrical Engineering from Xian Jiaotong University in 2009. Yuxun's research interest includes statistical learning theory and paradigms for modern information rich, large-scale and human-involved systems.

Reza Arghandeh (SM'01-M'13) is an assistant professor in the ECE Dept and the Center for Advanced Power System. He has been a postdoctoral scholar at the University of California, Berkeley's California Institute for Energy and Environment 2013-2015. He has five years industrial experience in power and energy systems. He completed his Ph.D. in Electrical Engineering with a specialization in power systems at Virginia Tech. He holds Master's degrees in Industrial and System Engineering from Virginia Tech 2013 and in Energy Systems from the University of Manchester 2008. From 2011 to 2013, he was a power system software designer at Electrical Distribution Design Inc. in Virginia. Dr. Arghandehs research interests include, but are not limited to data analysis and decision support for smart grids and smart cities using statistical inference, machine learning, information theory, and operations research. He is a recipient of the Association of Energy Engineers (AEE) Scholarship 2012, the UC Davis Green Tech Fellowship 2011, and the best paper award from the ASME 2012 Power Conference and IEEE PESGM 2015. He is the chair of the IEEE Task Force on Big Data Application for Power Distribution Network and secretary of the IEEE Working Group on Distribution Power Quality.

Costas J. Spanos (M'77-SM'92-F'98) received the EE Diploma from the National Technical University of Athens, Greece in 1980 and the M.S. and Ph.D. degrees in ECE from Carnegie Mellon University in 1981 and 1985, respectively. In 1988 he joined the Faculty at the department of Electrical Engineering and Computer Sciences of the University of California at Berkeley. He has served as the Director of the Berkeley Microlab, the Associate Dean for Research in the College of Engineering and as the Chair of the Department of EECS. He works in statistical analysis in the design and fabrication of integrated circuits, and on novel sensors and computer-aided techniques in semiconductor manufacturing. He also works on statistical data mining techniques for energy efficiency applications. He has participated in two successful startup companies, Timbre Tech, (acquired by Tokyo Electron) and OnWafer Technologies (acquired by KLA-Tencor). He is presently the Director of the Center of Information Technology Research in the Interest of Society (CITRIS) and the Chief Technical Officer for the Berkeley Educational Alliance for Research in Singapore (BEARS).