A Game-Theoretic Data-Driven Approach for Pseudo-Measurement Generation in Distribution System State Estimation

Kaveh Dehghanpour, *Member, IEEE*, Yuxuan Yuan, *Student Member, IEEE*, Zhaoyu Wang, *Member, IEEE*, and Fankun Bu, *Student Member, IEEE*

***Abstract*—In this paper, we present an efficient computational framework with the purpose of generating weighted pseudo-measurements to improve the quality of distribution system state estimation (DSSE) and provide observability with advanced metering infrastructure (AMI) against unobservable customers and missing data. The proposed technique is based on a game-theoretic expansion of relevance vector machines (RVMs). This platform is able to estimate the nodal power consumption and quantify its uncertainty while reducing the prohibitive computational burden of model training for large AMI datasets. To achieve this objective, the large training set is decomposed and distributed among multiple parallel learning entities. The resulting estimations from the parallel RVMs are then combined using a game-theoretic model based on the idea of repeated games with vector payoff. It is observed that through this approach and by exploiting the seasonal changes in customers' behavior the accuracy of pseudo-measurements can be considerably improved, while introducing robustness against bad training data samples. The proposed pseudo-measurement generation model is integrated into a DSSE using a closed-loop information system, which takes advantage of a branch current state estimator (BCSE) to further improve the performance of the designed machine learning framework. This method has been tested on a practical distribution feeder model with smart meter data for verification.**

***Index Terms*—Pseudo-measurements, smart meters, relevance vector machines, game theory, state estimation.**

I. Introduction

ELECTRIC distribution systems have been undergoing radical changes in control and management. The driving force behind these changes can be attributed to higher penetration of distributed renewable resources and employment of Advanced Metering Infrastructure (AMI) in power distribution systems [1]. Thus, system operators' access to residential, commercial, and industrial customer metering data has presented an opportunity for using data-driven techniques for system monitoring and control [2]. While the AMI data history can be humongous in size, it does not necessarily provide full observability for distribution systems due to the limited number of smart meters compared to the huge size of the network and the common missing data problem [3], [4].

Pseudo-measurement generation techniques are used to improve the observability of distribution networks by performing data-driven power consumption estimation (in case of missing data, communication delays, and unobserved loads) [5]. Also, weights are assigned to these estimated values to define the operator's confidence in the accuracy of pseudo-measurements in the state estimation process. Since the efficiency of distribution system control and management can be negatively affected by the inaccuracy of the generated pseudo-measurement samples, it is of critical importance to design data-driven load estimation methods capable of providing accurate pseudo-measurement samples to improve the quality of distribution system monitoring [6].

Several papers have studied the problem of pseudo-measurement generation for distribution system monitoring and state estimation. The literature in this area can be roughly categorized into two groups based on the proposed solution approaches.

1) Statistical and probabilistic models: The previous works in this category rely on statistical and probabilistic analysis of the available AMI data history for constructing pseudo-measurement generation methods. Empirical Gaussian distributions have been conventionally used for estimating the Probability Density Functions (PDF) of consumer load profiles and generating pseudo-measurements [7]. In [8], empirical consumption PDFs are constructed employing Beta and log-normal distributions, which show improved performance over single Gaussian approach. These PDFs are then used for generating estimated power consumption data samples. Gaussian Mixture Models (GMM) have also been shown to be an improvement over mere fitting of a single distribution function to the available data [9], [10]. In a more recent work, data clustering has been combined with GMM to improve the pseudo-measurement generation process [11]. A weather-dependent empirical PDF construction scheme for distributed PV systems is proposed in [12], as pseudo-measurement generator, which is shown to have superior performance over conventional statistical methods. Statistical load profile and power loss estimation have been used in [13] and [14],

Manuscript received May 30, 2018; revised August 13, 2018, September 25, 2018, and December 5, 2018; accepted January 11, 2019. This work was supported by the Advanced Grid Modeling Program at the U.S. Department of Energy Office of Electricity under Grant DE-OE0000875. Paper no. TSG-00817-2018. *(Corresponding author: Zhaoyu Wang.)*

The authors are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: kavehd@iastate.edu; wzy@iastate.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2019.2893818

respectively, to model the uncertainty of load behavior and improve the observability of distribution networks.

2) Machine learning models: Another group of researchers have adopted machine-learning-based methods for distribution system load estimation. In comparison with the first group, these methods are able to further improve the accuracy of pseudo-measurements by exploiting the available real-time data samples. A Probabilistic Neural Network (PNN) is proposed in [15] for assigning load profiles to loads in distribution systems. In [16], an Artificial Neural Network (ANN) is used for generating pseudo-measurements using the real-time line power flow measurements. Missing data reconstruction using a neural network approach has also been employed in [4]. Using the concept of Parallel Distributed Processing networks (PDP) a load estimation mechanism has been developed in [17] to design a robust state estimator for distribution systems. An adaptive Nonlinear Auto-Regressive eXogenous (NARX) model is proposed in [18] for load estimation in distribution networks. While these works provide invaluable insights into distribution system monitoring, they have certain shortcomings, including: failure to capture seasonal correlations in customer behavior, not addressing the big-data challenge for large AMI datasets, and ignoring the possibility of using Distribution System State Estimation (DSSE) data for improving machine learning performance.

In this paper, we propose a novel nodal load estimation process that can be used for pseudo-measurement generation for reconstructing unknown and missing power measurement data to improve the accuracy and precision of DSSE, while improving system observability. This method can in principle be applied to both primary and secondary distribution systems, to estimate power consumption at secondary transformer or customer levels. However, the primary target in this paper is to perform pseudo-measurement generation and load estimation for primary networks. The proposed machine-learning-based approach employs the concept of Relevance Vector Machine (RVM) to design sparse kernelized nonlinear regression models [19]. Moreover, unlike most regression models, RVM is capable of quantifying the uncertainty of pseudo-measurements by learning the variance of the estimated output. The variance learning process eliminates the need for relying on high-variance empirical distributions and is used to define weights for pseudo-measurements in the DSSE. Moreover, the inherent pruning mechanism of RVM introduces robustness against bad training data samples in the state estimation process. To alleviate the high cost of training, we propose a parallel computational framework using Multiple RVM (MRVM) units, each fitting a probabilistic model to a region of training set. The outcomes of these parallel training units are then recombined using a game-theoretic strategy to obtain final pseudo-measurement power consumption samples (along with their estimated variance). This game-theoretic framework is based on the concept of repeated games with vector payoffs [20], [21]. It is observed that by employing this technique the pseudo-measurement generation accuracy can be significantly improved by exploiting the strong seasonal changes in customer behavior. The power consumption estimation model is then integrated with a Branch Current State Estimator (BCSE) module through a closed-loop information system to iteratively improve the pseudo-measurements using the additional information provided by the BCSE. The idea of using corrective closed-loop information system for DSSE has been employed in [17] and [18], as well. It will be shown that using the proposed machine learning technique, the performance of both pseudo-measurement generation and DSSE can be enhanced considerably. The machine-learning-based estimation technique is tested on real data from a distribution feeder belonging to a utility company in the U.S. with smart meter measurements (power consumption and voltage measurement data).

To summarize, the contributions of this paper are as follows: a novel computationally-efficient machine learning framework is proposed to generate accurate nodal active power pseudo-measurement samples for DSSE. The novelty of the proposed model is to train parallel machine learning units by exploiting the seasonal patterns in load, which improves the performance of pseudo-measurements and computational efficiency of the framework. Seasonal changes in customer behavior are captured via a game-theoretic platform. Also, compared to previous works in the literature, the proposed approach provides a basis for automatic rejection of bad training data samples for enhanced robustness against noise, along with concurrent estimation of the variance of pseudo-measurements. The proposed machine learning module is integrated within a closed-loop DSSE to further improve the accuracy of state estimation, by feeding the DSSE information back into the machine-learning-based power consumption estimation. This paper is an effort towards enhanced monitoring and management of emerging smart distribution grids as cyber-physical systems using AMI [22], [23]. The proposed framework is tested on real utility data for verification.

The rest of this paper is constructed as follows: in Section II, a description of the game-theoretic probabilistic learning framework for power consumption pseudo-measurement generation is presented. In Section III, the overall closed-loop DSSE module is described (and summarized in Section IV). The numerical results are analyzed in Section V. In Section VI, the conclusions of the paper are presented.

II. Proposed Pseudo-Measurement Generator

A. AMI Dataset Description and Pre-Processing

The available AMI data history contains the hourly power consumption (kW) and voltage magnitude measurement of 3000 customers (with more than 40,000 data samples per customer) connected to 10 distribution feeders, which are located in the U.S. The dataset spans a time period of around five consecutive years (2013-2018). While a few industrial and large commercial loads are included in the dataset, the majority of customers are residential and small commercial loads.

The data was initially processed to remove grossly erroneous data samples. The bad data removal process was defined by the deviation of data samples from the seasonal mean of consumption signal for each customer. Hence, the samples that fall outside of ±5 deviation from the seasonal mean are removed, as having grossly erroneous values. The dataset

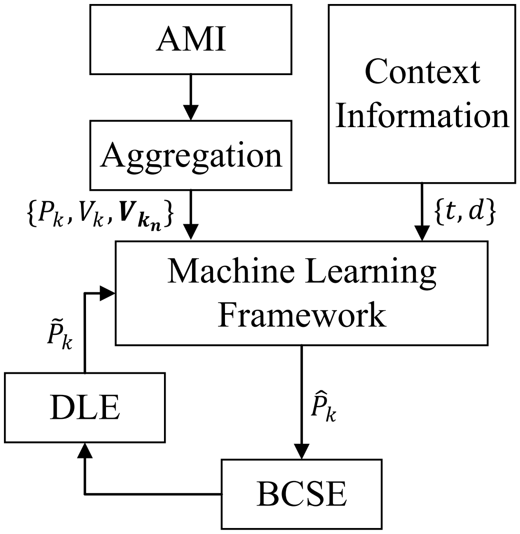

Fig. 1. Machine learning framework functionality.

was divided into two separate subsets for training (80% of the total data) and testing (20% of the total data). K-fold cross-validation was performed (over the training set) to choose certain model parameters (e.g., kernel bandwidth) [24]. A basic statistical analysis was performed on the dataset to identify variables with high correlation levels. As discussed in [25], the power consumption variable has a relatively high correlation level with voltage magnitude at the same bus and the neighboring nodes. This was also observed in the distribution system under study in this paper, specifically for larger loads, for which close to unit correlation values were recorded. Hence, available voltage magnitude measurement samples can be used as inputs in the regression models for estimating the consumption levels at different buses of the feeder. All these variables are normalized based on their maximum/minimum range of change. The objective of the pseudo-measurement generation process is to use the available noisy observations in real-time (i.e., customer voltage measurements, customer power measurements, time of day, etc.) to infer the unknown power consumption levels of unobserved secondary transformers connected to the primary distribution feeder nodes. The unobservability can be caused by the unavailability of meters, missing data, bad data, communication delays, and faulty measurements. To perform this task, regression models are trained using the system data history and employed to develop a mapping between the input and output data samples.

B. Machine Learning Framework

The machine learning framework is a supervised approach, which maps the input data to the output target space (power consumption), as shown in Fig. 1. The input space consists of three types of variables: 1) nodal voltage measurement data input with high correlation with power consumption (measured at bus k (V_k) or neighboring buses ($V_{k_n}, k_n \in N_k$)). A "node" or a "bus" in this paper refers to the primary network nodes. Notice that customers are connected to these primary network nodes via secondary transformers. Hence, available AMI measurements are collected from secondary distribution systems. To transfer these measurements to the primary feeder, the aggregator module performs two operations: (a) the available customer power measurements are aggregated at secondary transformer level through summation at different times, ignoring system losses. (b) The available secondary voltage measurements are averaged at the secondary transformer level for each time point, and then transferred to the primary side of each secondary transformer. Note that our assumption here is that the voltage drops on secondary networks are small. 2) context variables (time of day (t), and day of week d), and 3) the "feedback" power consumption signal generated by DSSE-based Load Estimation module (DLE), which also is highly correlated with the target nodal power consumption (more details in Section III). Note that in this paper two distinct variables are defined to approximate the target power consumption space: $\hat{P}_k$, which defines the k^{th} node's power pseudo-measurement variable (i.e., output of the machine learning framework), and $\tilde{P}_k$, which denotes the estimated nodal power using the DLE module (i.e., DSSE feedback signal). Basically, after solving BCSE over the primary network the estimated nodal voltages (or branch currents) are used to determine nodal power consumption levels. These estimations are used in a closed-loop mechanism to re-train the machine learning consumption estimation models. Hence, the role of the DLE module is to provide a link between the BCSE and machine learning framework.

The RVM algorithm is premised on a kernelized regression model, which can be formulated as follows [19]:

$$\hat{P}_k = \sum_{i=1}^{N} \omega_i K(\boldsymbol{x}(k), \boldsymbol{x_i}(k)) + \omega_0 \tag{1}$$

where, $\hat{P}_k$ represents the nodal power consumption pseudo-measurement for the k^{th} node, N denotes the total number of samples in the training set (i.e., number of previous observations), ω_i is the weight assigned to the i^{th} input sample in the training set ($\boldsymbol{x_i}$), and K denotes the kernel function over the samples in the training set and the new input sample $\boldsymbol{x}$ in the test set ($\boldsymbol{x}(k) = \{V_k, \boldsymbol{V_{k_n}}, t, d, \tilde{P}_k\}$). In this paper, radial basis function kernel, which is a measure of similarity between the training samples and the new observations, is used to quantify $K(.,.)$:

$$K\big(\boldsymbol{x_i}(k), \boldsymbol{x_j}(k)\big) = \exp\left\{-\frac{\left\|\boldsymbol{x_i}(k) - \boldsymbol{x_j}(k)\right\|^2}{r^2}\right\} \tag{2}$$

where, r is a tunable parameter which defines the kernel bandwidth. The objective of the machine learning framework is twofold: 1) learn the parameters of the kernelized regression model (ω_i's), 2) quantify the uncertainty of estimation. This uncertainty is defined by the variance (σ^2) of the estimation error $\epsilon = P_k - \hat{P}_k$, where P_k is the power consumption of the k^{th} node. RVM provides a computationally robust approach to achieve these goals. The learning mechanism employs a probabilistic view of the regression equation (1), in which parameters $\boldsymbol{\omega} = \{\omega_0, \ldots, \omega_N\}$ are assumed to be normally-distributed independent random variables, with hyperparameters α_i defining their variance, as follows:

$$p\{\boldsymbol{\omega}|(\alpha_0, \ldots, \alpha_N)\} = \prod_{i=0}^{N} \mathcal{N}\Big(0, \alpha_i^{-1}\Big) \tag{3}$$

where, $\mathcal{N}(a, b)$ denotes a normal distribution with mean a and variance b. Note that using (3), the α values can be used for eliminating irrelevant samples and pruning the training set. Accordingly, data samples for which the α levels converge to very large values can be removed safely from the training set, as their assigned weights get more concentrated around zero. The learning process is based on finding the most probable values for the set of hyperparameters $\{\alpha_0, \ldots, \alpha_N\}$ and parameter σ of the kernelized model to maximize the marginal likelihood function, which is formulated as follows:

$$\Big(\boldsymbol{\alpha^*}, \sigma^*\Big) = \arg\max_{\boldsymbol{\alpha},\sigma} p\{P_k|(\boldsymbol{\alpha}, \sigma)\} \tag{4}$$

To achieve this, different recursive update rules have been obtained for these variables based on expectation-maximization process. The overall algorithm has the following steps for each bus, as discussed in [19]:

- **Step 1:** Initialize hyperparameters $\boldsymbol{\alpha}$, and parameter σ
- **Step 2:** Formulate the "design matrix", Φ, and auxiliary matrix A over the existing data samples in the training set $X = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_N}\}$:

$$\Phi = \begin{bmatrix} 1 & K(\boldsymbol{x_1}, \boldsymbol{x_1}) & \cdots & K(\boldsymbol{x_1}, \boldsymbol{x_N}) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K(\boldsymbol{x_N}, \boldsymbol{x_1}) & \cdots & K(\boldsymbol{x_N}, \boldsymbol{x_N}) \end{bmatrix} \tag{5}$$

$$A = \begin{bmatrix} \alpha_0 & & \\ & \ddots & \\ & & \alpha_N \end{bmatrix} \tag{6}$$

- **Step 3:** Given the current values of $\boldsymbol{\alpha}$ and σ, the parameters $\boldsymbol{\omega}$ are estimated using a joint Gaussian distribution with covariance matrix $\boldsymbol{\Sigma}$ and mean vector $\boldsymbol{\mu}$, obtained as follows:

$$\boldsymbol{\Sigma} = \Big(\sigma^{-2}\Phi^\top\Phi + A\Big)^{-1} \tag{7}$$

$$\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}\Phi^\top\boldsymbol{P}_k \tag{8}$$

- **Step 4:** Update hyperparameters $\boldsymbol{\alpha}$ and parameter σ by equating the derivative of the objective function in (4) to zero, as follows:

$$\alpha_i^{new} = \frac{1 - \alpha_i\Sigma_{i,i}}{\mu_i^2} \tag{9}$$

$$\Big(\sigma^2\Big)^{new} = \frac{||\boldsymbol{P}_k - \Phi\mu||^2}{N - \sum_i\big(1 - \alpha_i\Sigma_{i,i}\big)} \tag{10}$$

  where, $\Sigma_{i,i}$ and μ_i denote the $(i,i)^{th}$ and i^{th} elements of $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$, respectively.
- **Step 5:** Prune the training data set by removing samples that correspond to $\alpha_i \geq \alpha_{max}$, with α_{max} denoting a user-defined threshold. The columns and rows of Φ corresponding to the pruned data samples will also be removed.
- **Step 6:** Go to Step 2, until convergence is achieved (i.e., changes in hyperparameters fall below a threshold).

The objective of RVM is to learn a "sparse" model using the basic regression framework (1) (with ω_i's and σ as model parameters to be learned). The sparsity of the learning process is based on convergence of most of the model parameters (ω_i) to near-zero values, which is also an automatic mechanism to avoid overfitting. To implement this mechanism, a pruning operation is performed at each iteration of the algorithm (Step 5) to eliminate the irrelevant data-points within the training set (only "relevant" samples are used for model-fitting).

Following convergence, the estimated power consumption target variable at bus k ($\hat{P}_k$) can be written as a conditional normal distribution (which is highly nonlinear in the input variables):

$$p\Big(\hat{P}_k|X\Big) \sim \mathcal{N}\Big(\boldsymbol{\mu}^\top\boldsymbol{\phi}(\boldsymbol{x}(k)), \sigma^2 + \boldsymbol{\phi}(\boldsymbol{x}(k))^\top\boldsymbol{\Sigma}\boldsymbol{\phi}(\boldsymbol{x}(k))\Big) \tag{11}$$

where, $\boldsymbol{x}(k)$ denotes the input variable from the test set. Also, $\boldsymbol{\phi}$ is the basis function designed over the remaining training samples $\boldsymbol{x_r} = \{\boldsymbol{x_{r_1}}, \ldots, \boldsymbol{x_{r_M}}\}$ where $\boldsymbol{x_r} \subset X$, and is defined as follows:

$$\boldsymbol{\phi}(\boldsymbol{x}(k)) = \begin{bmatrix} 1 & K\big(\boldsymbol{x}(k), \boldsymbol{x_{r_1}}\big) & \cdots & K\big(\boldsymbol{x}(k), \boldsymbol{x_{r_M}}\big) \end{bmatrix}^\top \tag{12}$$

As can be seen from (11), RVM is able to estimate both the target variable and its uncertainty (i.e., variance parameter, which represent factors such as noisy data and modeling errors).

C. Game-Theoretic Extension

The computational complexity of RVM is normally proportional to N^3 (with N denoting the number of training samples), which poses a considerable burden for large datasets. In this paper, to reduce the high computational cost of learning, the training dataset is decomposed into multiple subsets and distributed among a population of RVMs that train models in parallel with each other. Hence, each RVM unit is trained based on a specific time interval of the input space. In this way, the computational load becomes proportional to $\frac{N^3}{M^2}$, with M denoting the number of parallel RVM units. Hence, the computational complexity can be reduced by a factor of $1/M^2$ due to parallelization compared to the case where the whole dataset is used for training one RVM unit. The generated pseudo-measurement samples from the parallel RVM units are then recombined through weighted averaging (with weight value $w_{j,t}$ for the j^{th} RVM unit at time t) to reach a final power consumption pseudo-measurement value. The objective is to find the optimal values of the weight values to maximize the pseudo-measurement accuracy. It was observed that to reach the best pseudo-measurement accuracy, the training set should be decomposed based on seasons of the year, which implies existence of strong seasonal changes in customers' behavior. Thus, four parallel RVM units (each corresponding to a season) are selected and trained over the training set. The recombination process has to be performed in a manner to preserve the precision of the estimation process. To perform this recombination task, the pseudo-measurement generation process is modeled as a repeated game with vector payoff [21]. Based on this model, the game has two elements: 1) the "nature", which generates target time-series according to an unknown process (in our case, these time-series are the estimated nodal power consumption data generated by DLE), and 2) the "estimator" (referred to as the "player"), which has the objective of inferring the behavior of nature and tries to maximize its long-term payoff by predicting the time-series generated by the nature. The estimator has access to multiple sources of "advice" (generated by RVM units) and needs to combine the received advice in a way to optimize its behavior in the game. Mathematically, the goal of the estimator is to minimize the *Cumulative Regret*, which is defined with respect to the j^{th} advisor ($j \in \{1, \dots, M\}$), k^{th} node, at time m, as follows:

$$R_{j,k}(m) = \sum_{t=1}^{m} \Big\{ \ell\Big(\hat{P}_k(t), P_k(t)\Big) - \ell\big(f_{j,k}(t), P_k(t)\big) \Big\} \tag{13}$$

where, $f_{j,k}(k)$ is the j^{th} advisor (i.e., RVM unit) estimation of the target variable ($P_k(t)$). The function $\ell(.,.)$ defines the loss level due to mis-estimation, and is defined as $\ell(x, y) = |x - y|$ (which is convex in its first variable). Hence, the cumulative regret at a certain time point represents the player's loss for not following a specific advisor's estimations up to that point. For ease of reference, the player's instantaneous regret level with respect to the j^{th} advisor at time t is defined as follows:

$$r_{j,k}(t) = \ell\Big(\hat{P}_k(t), P_k(t)\Big) - \ell\big(f_{j,k}(t), P_k(t)\big) \tag{14}$$

Hence, the instantaneous regret vector and the regret vector are defined as, $\boldsymbol{r_k(t)} = (r_{1,k}(t), \dots, r_{M,k}(t))^\top$ and $\boldsymbol{R_k(m)} = \sum_{t=1}^{m} \boldsymbol{r_k(t)}$, respectively. While $\boldsymbol{r_k(t)}$ represents a vectorized representation of instantaneous regret in the advisor space, $\boldsymbol{R_k(m)}$ quantifies the summation of these instantaneous vectors up to a point in time.

The objective of the player is to assign optimal weight values to the advisors. Thus, the combination process for obtaining pseudo-measurements relies on weighted averaging of the received estimations from the RVM units, as follows:

$$\hat{P}_k(t) = \frac{\sum_{j=1}^{M} w_{j,k}(t-1) f_{j,k}(t)}{\sum_{j=1}^{M} w_{j,k}(t-1)} \tag{15}$$

The weight selection process is based on the choice of scalar non-negative, and twice-differentiable convex *potential functions* over the regret vector, denoted by $U(\boldsymbol{R_k(m)})$ [21]. The goal of weight selection is to reduce the potential function value to limit the long term accumulated estimation regret. Basically, the potential function penalizes higher levels of regret. Hence, one choice of weight for adaptive correction of importance levels (weights) of RVM units is $w_{j,k}(t) = \nabla U(\boldsymbol{R_k(t)})_j$ to improve the weights based on local gradient information of potential function. In this paper, an exponential potential function is chosen as follows:

$$U(\boldsymbol{R_k(t)}) = \frac{1}{\eta_k(t)} \ln\left(\sum_{j=1}^{M} e^{\eta_k(t) R_{j,k}(t)} \right) \tag{16}$$

where, $\eta_k(t)$ is a tunable parameter (at time t). The choice of an exponential potential function leads to the following weight update mechanism:

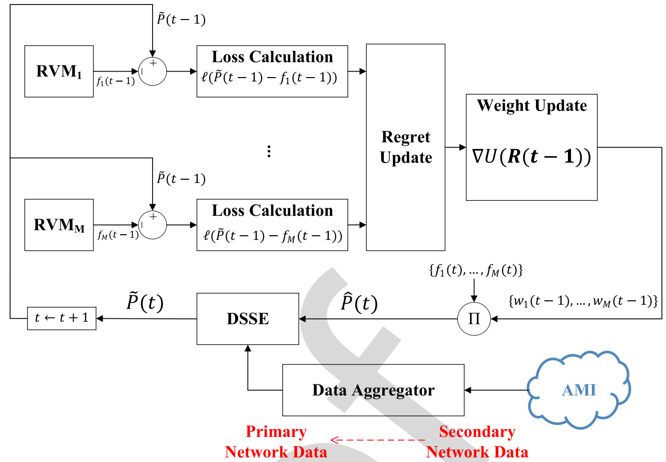

Fig. 2. Proposed structure of the game-theoretic learning process.

$$w_{j,k}(t-1) = \nabla U(\boldsymbol{R_k(t-1)})_j = \frac{e^{\eta_k(t) R_{j,k}(t-1)}}{\sum_{j=1}^{M} e^{\eta_k(t) R_{j,k}(t-1)}} \tag{17}$$

It can be proved that with the choice of $\eta_k(t) = \sqrt{\frac{8 ln M}{t}}$ (and a normalized convex loss function) the following upper-bound on the maximum regret level is achieved [21]:

$$\max_{j=1,\dots,M} R_{j,k} \leq 2\sqrt{\frac{k \cdot \ln M}{2}} + \sqrt{\frac{\ln M}{8}} \tag{18}$$

The overall game-theoretic platform is shown in Fig. 2. As can be seen in this figure, the game-theoretic machine learning framework updates the importance weight factors online (in case the nodal data samples or DLE outputs become available) or offline (using cross-validation). Also, the combined estimated nodal power pseudo-measurement variance for the k^{th} node ($\hat{\sigma}_k^2$) is calculated at time t as follows:

$$\hat{\sigma}_k^2(t) = \frac{\sum_{j=1}^{M} w_{j,k}(t-1)^2 \hat{\sigma}_{j,k}(t)^2}{\left(\sum_{j=1}^{M} w_{j,k}(t-1) \right)^2} \tag{19}$$

where, $\hat{\sigma}_{j,k}(t)^2$ is the estimated variance for the j^{th} RVM unit at time t obtained using (11).

III. Closed-Loop DSSE Module

The structure of the DSSE module is shown in Fig. 3. The module consists of two subsystems: BCSE and DLE.

A. BCSE

A BCSE algorithm is used for implementing the DSSE module over the primary distribution system [26], [27]. This algorithm is based on minimization of summation of weighted measurement residuals:

$$\hat{\boldsymbol{s}} = \arg\min_{\boldsymbol{s}} \sum_{i=1}^{N_z} \frac{1}{\sigma_i^2} (z_i - h_i(\boldsymbol{s}))^2 \tag{20}$$

where, z_i's represent the measurement and pseudo-measurements (with standard deviations σ_i representing

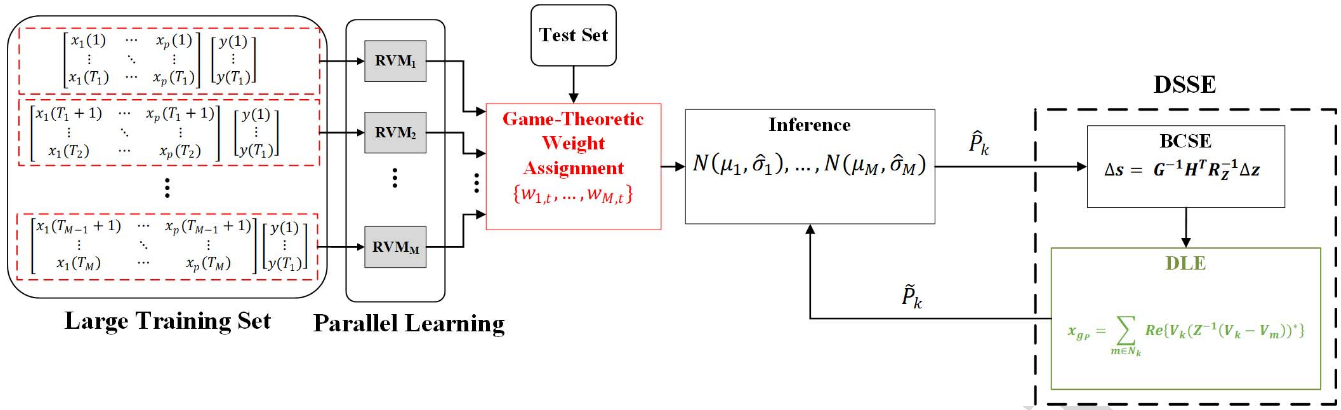

Fig. 3. Overall structure of the DSSE.

user's confidence, and total number of N_z), $\boldsymbol{s}$ denotes the state vector, h_i is the measurement function (which maps the state vector to the i^{th} measurement/pseudo-measurement.) In this paper, the measurement samples are the active/reactive customer power consumption and voltage magnitude data which are aggregated at secondary transformer level to obtain equivalent measurements for the primary network, branch flow measurements (primary feeder), and voltage measurement at the main substation. The state variables are the real/imaginary branch current values for each phase of the primary feeder. Gauss-Newton method is used to iteratively update the state vector and achieve convergence [28]. The update mechanism at step q is as follows:

$$\boldsymbol{s}_{q+1} = \boldsymbol{s}_q + G^{-1}(\boldsymbol{s}_q) H^\top(\boldsymbol{s}_q) R_Z^{-1} (z - \boldsymbol{h}(\boldsymbol{s}_q)) \tag{21}$$

where, G is the "gain matrix" defined as $G(\boldsymbol{s}_q) = H^\top(\boldsymbol{s}_q) R_Z^{-1}(\boldsymbol{s}_q) H(\boldsymbol{s}_q)$, H is the Jacobian matrix corresponding to the measurement function vector $\boldsymbol{h}(\boldsymbol{s}_q)$, and $R_Z = diag(\sigma_1^2, \dots, \sigma_{N_z}^2)$ is the measurement/pseudo-measurement uncertainty matrix.

B. DLE

After the convergence of the BCSE, the active power consumption is estimated at each node of the feeder using the estimated nodal voltage variables for each phase, employing power flow equations:

$$\tilde{P}_k = \sum_{m \in N_k} \mathrm{Re}\left(\hat{V}_k \left(Z_{km}^{-1}\left(\hat{V}_k - \hat{V}_m\right)^*\right)\right) \tag{22}$$

where, $\hat{V}_k$ and $\hat{V}_m$ denote the BCSE-based three phase voltage phasor at bus k and its neighboring nodes (included in the set N_k), and Z_{km} defines the phase-based impedance matrix of the line connecting nodes k and m. The estimated active power usage of each node ($\tilde{P}_k$) is used to train and test the machine learning framework. The basic idea is that even under initial erroneous pseudo-measurement assignment, $\tilde{P}_k$ is highly correlated with the actual power usage information. The maximum correlation levels between the input/outputs of the machine learning framework at different nodes (for the primary distribution feeder) are shown in Fig. 4. As can be seen, the DLE output (obtained under open-loop state) has close-to-unity correlation with the actual power consumption. Hence, these artificially-constructed DLE signals can be exploited for training the machine learning framework to improve the accuracy of power consumption pseudo-measurements and state estimation algorithm in a closed-loop information system.

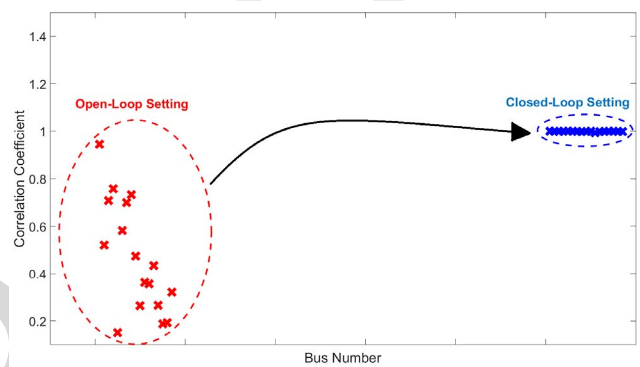

Fig. 4. Correlation between inputs/outputs of the machine learning framework, with respect to the state of the inner-loop.

IV. Overall Estimation Functionality

In this section a summary of the different stages of the proposed state estimation framework is presented:

- **Stage I - Offline BCSE:** Perform BCSE on the primary feeder using nodal measurement data history (consisting of real measurements and open-loop pseudo-measurement samples) to obtain estimated power consumption data. The primary feeder nodal measurements are obtained from aggregating the AMI measurements of secondary networks.
- **Stage II - Offline Training:** Augment the training set, using the DLE outcome of Stage I. Decompose the training set along seasonal time frames and train parallel RVM units (Section II-B).
- **Stage III - Weight Initialization:** Choose uniformly-random initial weights for the RVM units.
- **Stage IV - Online Inference (time T):** Based on the available measurements and the DLE output (not available in the first iteration), and the weights assigned to

RVM units update the value and weights of pseudo-measurements (Section II-C).

- **Stage V - Online BCSE (time T):** Run the BCSE algorithm for T based on the input measurements and pseudo-measurements until convergence is achieved (Section III-A).
- **Stage VI - Online DLE (time T):** Update the power consumption information using the outcomes of DLE (Section III-B).
- **Stage VII - Loop Cycling (time T):** Go to Stage IV (with updates from DLE), until changes in pseudo-measurements for time T fall below a threshold.
- **Stage VIII - Weight Update:** $T \leftarrow T+1$, Update the weights assigned to the RVMs based on the latest available observations at time T (Section II-C). Go to Stage IV.

The overall complexity of the proposed system monitoring can be approximated by $O(\frac{N^3}{M^2} + M + N_b^3 f \epsilon^{-2})$, with N_b denoting the number of distribution system nodes, f is the number of iterations in the designed feedback loop, and ϵ is the threshold over gradient norm below which the BCSE is terminated. This complexity approximation is based on the computational complexities of three modules: multiple RVM learning ($O(\frac{N^3}{M^2})$) [19], game-theoretic extension ($O(M)$) [21], and BCSE ($O(N_b^3 f \epsilon^{-2})$).

The designed framework consists of numerical routines that need to have access to: 1) online AMI/SCADA/PMU data stream, and 2) AMI data history. In our case, the customer data history is available to utility partners directly or through hired third-party companies. The online data stream will be fed to the machine learning framework after resolving data formatting and structuring issues. Hence, protocols need to be designed to ensure the interoperability of interfaces. Other than that the proposed framework can be easily (and independently) implemented and integrated within the distribution automation systems with minimum modifications in the hardware (except maybe addition of parallel computational resources). The outcome of the framework is the state variables for the system operator.

V. Numerical Results

The proposed method is tested on a sample feeder from the available utility dataset (described in Section II) with 220 customers. The test feeder and symbolic secondary to primary data aggregation process are shown in Fig. 5. The test feeder has three primary power flow measurement units and has around 35% smart meter penetration. The accuracy of measurement units is assumed to be $\pm 1\%$. The performance of the monitoring system is analyzed in both open- and closed-loop states. Also, the machine learning framework's robustness against bad data has been compared to conventional methods, such as ANN, linear regression, and Gaussian Maximum Likelihood Estimation (MLE).

A. Pseudo-Measurement Generation Performance

The machine learning framework was tested using the AMI data history. The histogram for nodal power consumption pseudo-measurement error is shown in Fig. 6 for both open- and closed-loop situations. As can be seen in this figure, by closing the inner-loop (i.e., using DLE data) the pseudo-measurement *precision* (defined as the inverse of the error distribution variance) has been improved by a considerable margin of 347.6%. The Mean Absolute Percentage Error (MAPE) has also been reduced from 31.74% to 1.94% by employing model training using the signals generated by the DLE module in the inner-loop. The actual nodal consumption is used as the ground truth for performance evaluation.

The behavior of the outer-loop is captured by studying the changes in the game-theoretic weight assignment module. The weights assigned to the parallel RVM units (averaged over all nodes), corresponding to different seasons of the year in the training set, are shown in Fig. 7. Given that the test set is selected to be the summer of 2017, higher weights are assigned to the regions of training set with similar

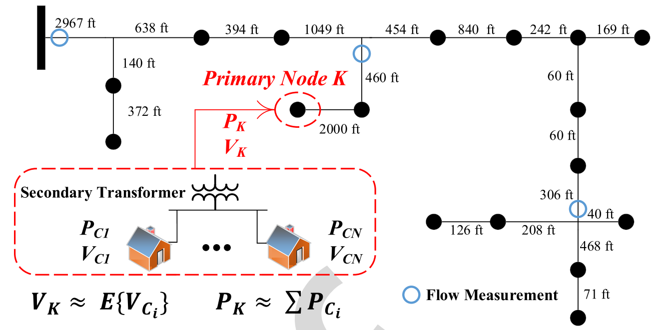

Fig. 5. The test system under study (220 customers).

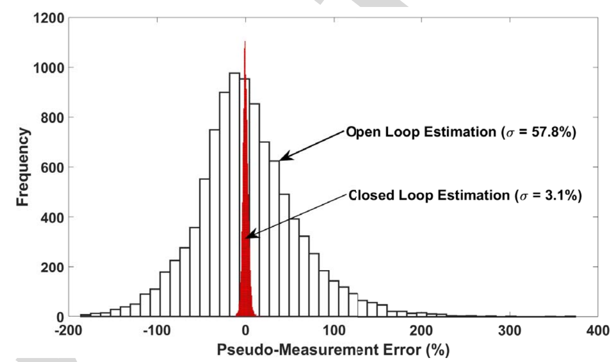

Fig. 6. The pseudo-measurement generation error histograms.

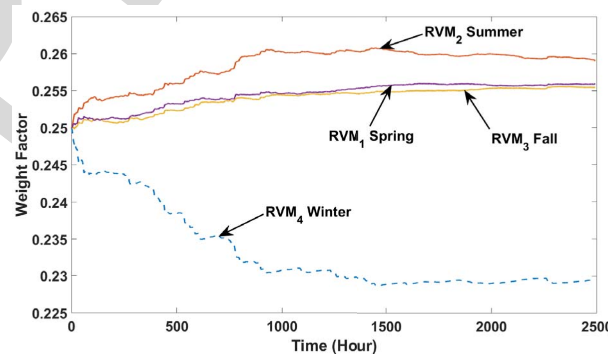

Fig. 7. Game-theoretic weight assignment (outer-loop).

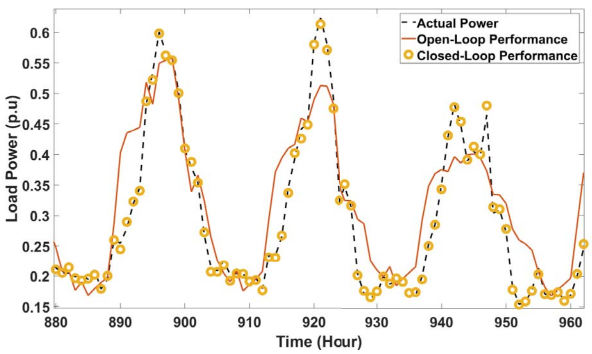

Fig. 8. Pseudo-measurement accuracy demonstration.

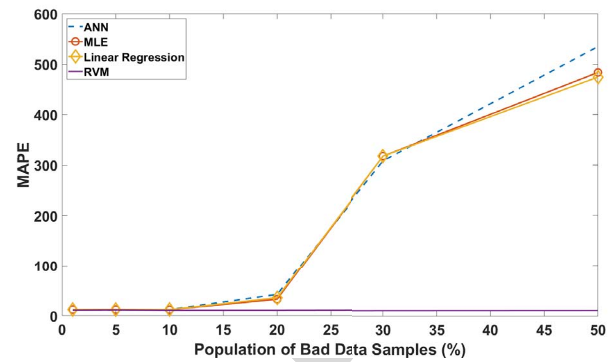

Fig. 9. Performance of the machine learning frameworks against bad data.

patterns (summer and spring of 2014-2016). A critical aspect of the estimation process is that the game-theoretic aggregation of the RVM units outperforms each of the individual units in the long run *on average*. The long run average MAPE for the aggregate estimator is 1.94%, while this index increases to 2.18%, 2.97%, 3.15%, and 5.37% for the available individual RVM units, implying the advantage of the proposed signal combination method in terms of accuracy. Hence, parallelization not only reduces training computational complexity but also leads to more accurate pseudo-measurement samples.

The performance of the pseudo-measurement generation module for the two cases of open and closed inner-loop states are shown in Fig. 8. As can be seen in this figure, after closing the inner-loop near-perfect fit to the underlying data can be achieved, which demonstrates the effectiveness of the proposed machine learning framework in closed-loop setting.

Providing robustness and detecting bad data is a critical step of DSSE [29]–[31]. The robustness of the proposed machine learning model is tested by injecting artificially generated bad data to the training set. The pseudo-measurement generation MAPE is shown as a function of the bad data sample population for different methods in Fig. 9. To add the error to the training data two steps were taken: 1) N data points were randomly selected from the training set. 2) Noise values generated by Gaussian distributions were added to each selected data point. The Gaussian distributions have zero means and standard deviations equal to 50% of the magnitude of the corresponding selected data sample. After distorting the N training data samples the machine learning models are trained and tested. This process is repeated several times for each N value. Then N is modified (decreased or increase). As is seen in this figure, an increase in the population of bad data samples leads to a drastic decline in the performance of ANN, MLE, and linear regression. However, the performance of the proposed MRVM method remains highly stable for a wide range of bad data sample population size. The reason for this stability is the ability of the RVM algorithm to prune the training dataset and eliminate "irrelevant" data samples that do not contribute positively to the marginal likelihood function. In other words, RVM has a natural mechanism for bad data detection and elimination, which is highly beneficial when dealing with real data.

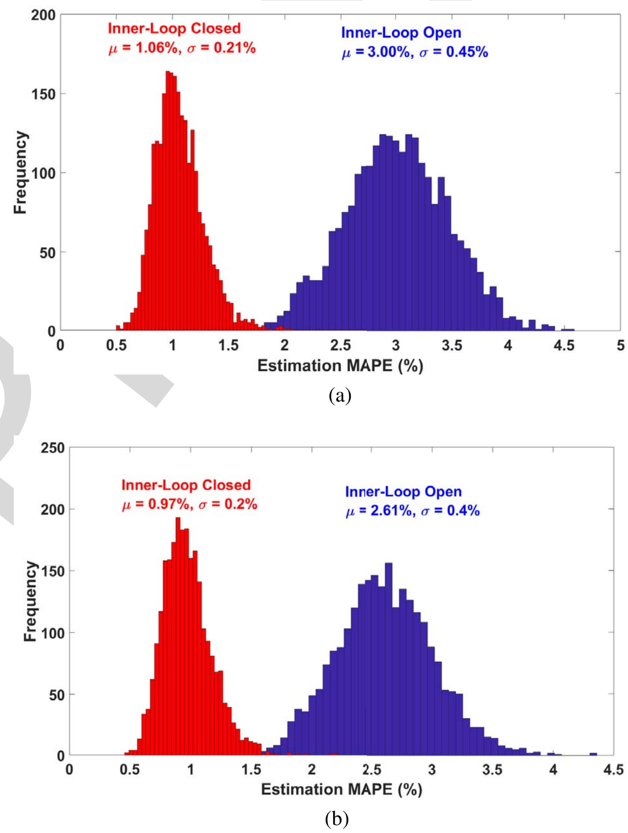

Fig. 10. BCSE performance in estimating state variables in open- and closed-inner-loop conditions. (a) Branch current real component error. (b) Branch current imaginary component error.

B. State Estimation Performance

The state estimation performance (in terms of MAPE) is shown in Fig. 10 for both open- and close-loop conditions for real and imaginary branch current components. As is demonstrated in these figures, using the closed-loop DSSE module improves both the accuracy and precision (i.e., mean and variance) of the BCSE.

The distribution of current magnitude and phase estimation error is shown under open- and closed-inner loop conditions in Fig. 11 using scatter plots. In this figure, the improvements in DSSE can be observed, where a shift in the regions with high concentration of error data is observed (from (1.57%, 2.61%) to (0.54%, 0.87%)). We have also observed

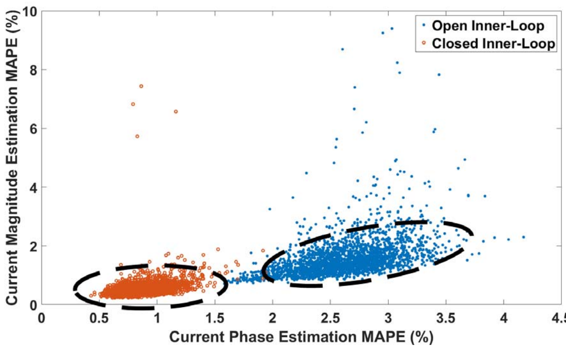

Fig. 11. State estimation accuracy demonstration (open-loop and closed-loop comparison).

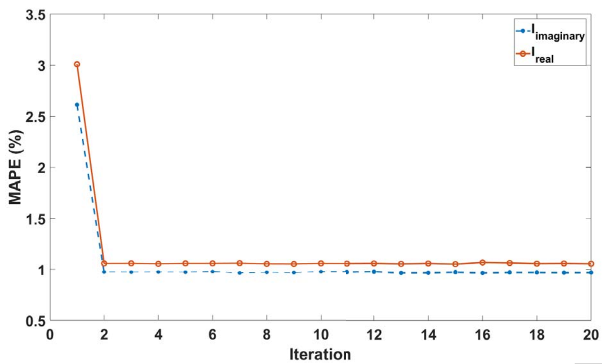

Fig. 12. Convergence of the proposed DSSE module.

that the performance of state estimation depends on the location and number of measurement units distributed across the system. However, in all cases the proposed closed-loop machine learning framework leads to improvements compared to the open-loop setting for any number of measurement units.

The convergence of the proposed DSSE model is shown in Fig. 12, where the estimation MAPE is demonstrated as function of iterations, with each iteration representing a cycle in the inner-loop. Note that the estimation error is calculated as an average over all branches in the feeder. As is seen in the figure, the proposed method reaches steady-state after a single iteration, which implies fast convergence and suitability for real-time applications.

The proposed framework requires an average 10.1 seconds per transformer per year of training data to generate solutions for each hour, as tested on a Intel Xeon CPU E3-1240 V6 @ 3.7 GHz hardware. Hence, given that the processing time is almost 357 times faster than the actual system time flow, the proposed method is well capable of real-time monitoring of distribution system states. The total training time using the data collected over 3 years, is 484.2 seconds.

VI. Conclusion

In this paper, we have presented a computationally-efficient machine learning method for accurate pseudo-measurement generation to improve the quality of DSSE against unknown, missing, and bad data. The proposed approach is based on parallel training of multiple machine learning units and is shown to be highly robust against bad data samples in the training set. Employing the proposed technique we are able to exploit the seasonal patterns in customers' behavior to improve the accuracy of pseudo-measurement generation. A nested closed-loop DSSE module is developed to improve the accuracy and precision of the state estimation process by enabling interaction between the learning framework and the DSSE. The proposed method is successfully tested on a utility feeder with real smart meter data.

References

[1] R. Hidalgo, C. Abbey, and G. Joós, "A review of active distribution networks enabling technologies," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Jul. 2010, pp. 1–9.

[2] R. R. Mohassel, A. Fung, F. Mohammadi, and K. Raahemifar, "A survey on advanced metering infrastructure," *Int. J. Elect. Power Energy Syst.*, vol. 63, pp. 473–484, Dec. 2014.

[3] S. Bhela, V. Kekatos, and S. Veeramachaneni, "Enhancing observability in distribution grids using smart meter data," *IEEE Trans. Smart Grid*, vol. 9, no. 6, pp. 5953–5961, Nov. 2018.

[4] V. Miranda, J. Krstulovic, H. Keko, C. Moreira, and J. Pereira, "Reconstructing missing data in state estimation with autoencoders," *IEEE Trans. Power Syst.*, vol. 27, no. 2, pp. 604–611, May 2012.

[5] A. Primadianto and C.-N. Lu, "A review on distribution system state estimation," *IEEE Trans. Power Syst.*, vol. 32, no. 5, pp. 3875–3883, Sep. 2017.

[6] A. Alimardani, F. Therrien, D. Atanackovic, J. Jatskevich, and E. Vaahedi, "Distribution system state estimation based on nonsynchronized smart meters," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2919–2928, Nov. 2015.

[7] A. Arefi, G. Ledwich, and B. Behi, "An efficient DSE using conditional multivariate complex Gaussian distribution," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 2147–2156, Jul. 2015.

[8] A. K. Ghosh, D. L. Lubkeman, M. J. Downey, and R. H. Jones, "Distribution circuit state estimation using a probabilistic approach," *IEEE Trans. Power Syst.*, vol. 12, no. 1, pp. 45–51, Feb. 1997.

[9] R. Singh, B. C. Pal, and R. Jabr, "Distribution system state estimation through Gaussian mixture model of the load as pseudo-measurement," *IET Gener. Transm. Distrib.*, vol. 4, no. 1, pp. 50–59, Jan. 2009.

[10] R. Singh, B. C. Pal, and R. A. Jabr, "Statistical representation of distribution system loads using Gaussian mixture model," *IEEE Trans. Power Syst.*, vol. 25, no. 1, pp. 29–37, Feb. 2010.

[11] Y. R. Gahrooei, A. Khodabakhshian, and R. A. Hooshmand, "A new pseudo load profile determination approach in low voltage distribution networks," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 463–472, Jan. 2018.

[12] A. Angioni, T. Schlösser, F. Ponci, and A. Monti, "Impact of pseudo-measurements from new power profiles on state estimation in low-voltage grids," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 1, pp. 70–77, Jan. 2016.

[13] D. T. Nguyen, "Modeling load uncertainty in distribution network monitoring," *IEEE Trans. Power Syst.*, vol. 30, no. 5, pp. 2321–2328, Sep. 2015.

[14] Q. Chen, D. Kaleshi, Z. Fan, and S. Armour, "Impact of smart metering data aggregation on distribution system state estimation," *IEEE Trans. Ind. Informat.*, vol. 12, no. 4, pp. 1426–1437, Aug. 2016.

[15] D. Gerbec, S. Gasperic, I. Smon, and F. Gubina, "Allocation of the load profiles to consumers using probabilistic neural networks," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 548–555, May 2005.

[16] E. Manitsas, R. Singh, B. C. Pal, and G. Strbac, "Distribution system state estimation using an artificial neural network approach for pseudo measurement modeling," *IEEE Trans. Power Syst.*, vol. 27, no. 4, pp. 1888–1896, Nov. 2012.

[17] J. Wu, Y. He, and N. Jenkins, "A robust state estimator for medium voltage distribution networks," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 1008–1016, May 2013.

[18] B. P. Hayes, J. K. Gruber, and M. Prodanovic, "A closed-loop state estimation tool for MV network monitoring and operation," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 2116–2125, Jul. 2015.

[19] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, Jun. 2001.

[20] M. Maschler, E. Solan, and S. Zamir, *Game Theory*. New York, NY, USA: Cambridge Univ. Press, 2013.

[21] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning, and Games*. New York, NY, USA: Cambridge Univ. Press, 2006.

[22] A. S. Bretas, N. G. Bretas, B. Carvalho, E. Baeyens, and P. P. Khargonekar, "Smart grids cyber-physical security as a malicious data attack: An innovation approach," *Electric Power Syst. Res.*, vol. 149, pp. 210–219, Aug. 2017.

[23] A. S. Bretas, N. G. Bretas, and B. Carvalho, "Further contributions to smart grids cyber-physical security as a malicious data attack: Proof and properties of the parameter error spreading out to the measurements and a relaxed correction model," *Electric Power Energy Syst.*, vol. 104, pp. 43–51, Jan. 2019.

[24] J. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning* (Springer Series in Statistics). New York, NY, USA: Springer, 2001.

[25] W. Luan, J. Peng, M. Maras, J. Lo, and B. Harapnuk, "Smart meter data analytics for distribution network connectivity verification," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1964–1971, Jul. 2015.

[26] M. E. Baran and A. M. Kelley, "A branch-current-based state estimation method for distribution systems," *IEEE Trans. Power Syst.*, vol. 10, no. 1, pp. 483–491, Feb. 1995.

[27] H. Wang and N. N. Schulz, "A revised branch current-based distribution system state estimation algorithm and meter placement impact," *IEEE Trans. Power Syst.*, vol. 19, no. 1, pp. 207–213, Feb. 2004.

[28] N. G. Bretas and A. S. Bretas, "The extension of the Gauss approach for the solution of an overdetermined set of algebraic non linear equations," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 65, no. 9, pp. 1269–1273, Sep. 2018.

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

[30] N. G. Bretas, A. S. Bretas, and A. C. P. Martins, "Convergence property of the measurement gross error correction in power system state estimation, using geometrical background," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3729–3736, Nov. 2013.

[31] N. G. Bretas, S. A. Piereti, A. S. Bretas, and A. C. P. Martins, "A geometrical view for multiple gross errors detection, identification, and correction in power system state estimation," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 2128–2135, Aug. 2013.

Kaveh Dehghanpour (S'14–M'17) received the B.Sc. and M.S. degrees in electrical and computer engineering from the University of Tehran, Tehran, Iran, in 2011 and 2013, respectively, and the Ph.D. degree in electrical engineering from Montana State University, Bozeman, MT, USA, in 2017. He is currently a Post-Doctoral Research Associate with Iowa State University, Ames, IA, USA. His research interests include application of machine learning and data-driven techniques in power system monitoring and control.

Yuxuan Yuan (S'18) received the B.S. degree in electrical and computer engineering from Iowa State University, Ames, IA, in 2017, where he is currently pursuing the Ph.D. degree. His research interests include distribution system state estimation, synthetic networks, data analytics, and machine learning.

Zhaoyu Wang (S'13–M'15) received the B.S. and M.S. degrees in electrical engineering from Shanghai Jiaotong University in 2009 and 2012, respectively, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Georgia Institute of Technology, GA, USA, in 2012 and 2015, respectively. He is the Harpole-Pentair Assistant Professor with Iowa State University. He was a Research Aid with Argonne National Laboratory in 2013 and an Electrical Engineer Intern with Corning Inc. in 2014. His research interests include power distribution systems, microgrids, renewable integration, power system resilience, and power system modeling. He is the Principal Investigator for a multitude of projects focused on the above areas and funded by the National Science Foundation, the Department of Energy, National Laboratories, PSERC, and Iowa Energy Center. He was a recipient of the IEEE PES General Meeting Best Paper Award in 2017 and the IEEE Industrial Application Society Prize Paper Award in 2016. He is the Secretary of IEEE Power and Energy Society Award Subcommittee. He is an Editor of the IEEE TRANSACTIONS ON POWER SYSTEMS, the IEEE TRANSACTIONS ON SMART GRID, and IEEE PES LETTERS, and an Associate Editor of *IET Smart Grid*.

Fankun Bu (S'18) received the B.S. and M.S. degrees from North China Electric Power University, Baoding, China, in 2008 and 2013, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA. From 2008 to 2010, he was a Commissioning Engineer with NARI Technology Company, Ltd., Nanjing, China. From 2013 to 2017, he was an Electrical Engineer with the State Grid Corporation of China at Jiangsu, Nanjing, China. His research interests include load modeling, load forecasting, distribution system estimation, machine learning and power system relaying.