Dynamic Distribution State Estimation Using Synchrophasor Data

Jianhan Song, Emiliano Dall'Anese, *Member, IEEE*, Andrea Simonetto, *Member, IEEE*, and Hao Zhu, *Senior Member, IEEE*

***Abstract*—The increasing deployment of distribution-level phasor measurement units (PMUs) calls for dynamic distribution state estimation (DDSE) approaches that tap into high-rate measurements to maintain a comprehensive view of the distribution-system state in real time. Accordingly, this paper explores the development of a fast algorithmic framework by casting the DDSE task within the *time-varying* optimization realm. The time-varying formulation involves a time-varying robustified least-squares approach, and it naturally models optimal trajectories for the estimated states under streaming of measurements. The formulation is based on a linear surrogate of the AC power-flow equations, and it includes an element of robustness with respect to measurement outliers. The paper then leverages a first-order prediction-correction method to achieve simple online updates that can provably track the state variables from heterogeneous measurements. This online algorithm is computationally efficient as it relies on the Hessian of the cost function without computing matrix-inverse. Convergence and bounds on the estimation errors of proposed algorithm can be analytically established.**

***Index Terms*—Distribution state estimation, synchrophasor data, time-varying optimization, prediction-correction methods.**

I. Introduction

RECENTLY, power distribution networks have witnessed an increasing connection of renewable energy sources, electric vehicles, energy storage systems, among other distributed energy resources. These transformations have propelled the development and deployment of advanced sensing, communications, and control technologies. In particular, distribution-level phasor measurement units (PMUs), termed as μPMUs [1] have equipped distribution system operators with synchronized, high-quality and high-resolution measurements that can be collected on a fast time scale. It is timely to design efficient and effective distribution situational awareness modules that can leverage the availability of PMU data therein.

Distribution dynamic state estimation (DDSE) is a fundamental tool that can enable general distribution operations and control tasks. This is similar to the operational paradigm for wide-area transmission grids [2]. Nonetheless, distribution networks have unique characteristics compared to transmission systems, such as unbalanced loads and higher resistance-to-reactance ratios [3]. Specifically, the unbalanced nature of distribution systems necessitates the multi-phase modeling at increased dimension and coupling. Moreover, nodes with zero injections commonly exist while historic load data are often included to improve redundancy. The high variation of accuracy from both types of data can lead to numerical conditioning issues. Thus, the dynamic SE approaches traditionally developed for transmission grids may not be directly applicable for distribution systems with PMU data.

Traditionally, distribution SE research has been limited to the static setting, constrained by meter availability and data rates. Several efforts have been focused on addressing the numerical conditioning issue using, e.g., branch current formulation [4], conversion to current measurements [5], [6], zero-injection information [7], [8]. More recent work has considered the incorporation of synchrophasor data for static SE, but using the purely linear SE assuming sufficient observability from PMU data only [9]–[11]. This is not yet a reality for distribution networks, and thus legacy meters should still be included for distribution SE. Motivated by the availability of fast PMU measurements, dynamic SE methods has been developed relying on the recursive Kalman filter (KF) updates for distribution systems [12], [13]. Note that dynamic SE in transmission systems also uses variations of KF recursions as the "workhorse" algorithm; see, e.g., [14], [15]. Please also see a recent review on distribution SE in [3].

This paper aims to develop an efficient and effective DDSE solution technique that can address the unique characteristics of distribution networks and fast sampling rates of PMU data. We leverage the linearized multi-phase AC power flow model recently developed in [16]. The model can account for unbalanced operation, as well as for wye and delta connections. Based on this model, Section II formulates a new DDSE problem by advocating a *time-varying* optimization formalism. This formulation involves a time-varying robustified least-squares approach, and it naturally models optimal trajectories for the estimated states under

Manuscript received October 14, 2018; revised March 31, 2019 and July 16, 2019; accepted August 20, 2019. Date of publication September 24, 2019; date of current version December 23, 2019. The work of E. Dall'Anese was supported in part by the National Renewable Energy Laboratory under Grant APUP UGA-0-41026-109. The work of H. Zhu was supported in part by NSF under Award ECCS-1802319. Paper no. TSG-01559-2018. *(Corresponding author: Hao Zhu.)*

J. Song and H. Zhu are with the Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX 78712 USA (e-mail: haozhu@utexas.edu).

E. Dall'Anese is with the Department of Electrical, Computer, and Energy Engineering, University of Colorado Boulder, Boulder, CO 80309 USA.

A. Simonetto is with the Optimization and Control Group, IBM Research Ireland, Dublin 15, D15 HN66 Ireland.

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

Digital Object Identifier 10.1109/TSG.2019.2943540

streaming of measurements. Thus, our proposed DDSE formulation is novel due to its linearized model and time-varying nature to account for dynamic states and data. Based on the DDSE formulation, Section III proposes an algorithmic framework to *track* the state of the distribution system by leveraging running prediction-correction methodologies [17], [18]. Prediction-correction methods involve two phases, sequentially implemented at each time step: i) a *prediction* phase where, based on the measurements collected up the current time instant, the algorithm attempts to predict the optimal solution of the next time period by exploring into intrinsic temporal correlations of the cost function; and, ii) once a new datum/measurement becomes available, the *correction* phase refines the predicted solution. To facilitate the development of computationally affordable algorithms, the paper considers first-order prediction-correction (FOPC) methods that rely on the Hessian of the cost function, instead of requiring the computation of its inverse [18]. FOPC methods are very attractive for DDSE problems where measurements are collected at high frequency by distribution-level PMUs or even distributed energy resources to enhance real-time situational awareness, which is another contribution of the present work. In particular, FOPC has merits in the following two cases:

- The DDSE task has a given computational budget to estimate the state before a new measurement is collected and processed. In this case, in par with the computational budget specified for each interval between the arrival of two measurements, FOPC is shown numerically to outperform iterative algorithms without prediction steps.
- FOPC can perform a prediction of the state while waiting for the measurement to be transmitted from the PMUs to the state estimator; once the measurement is received, the correction step can be performed. The prediction stage is shown to enable substantial improvements in terms of tracking performance.

Relative to traditional KF-based approaches, it is worth pointing out that: (i) FOPC provides an appreciable flexibility to include a variety of performance metrics in the cost function of the problem; for example, the paper will demonstrate how the DDSE can be easily robustified by modifying the cost function. (ii) KF requires covariance matrices; in a DDSE setting where heterogeneous measurements are collected at different rates, it is practically challenging to obtain accurate estimates of noise covariance matrices. Last, (iii) KF typically needs the Hessian inverse computations, which can be too computationally burdensome; on the other hand, FOPC relies on first-order updates at no Hessian inversion step. Overall, FOPC is naturally data-driven, while KF is grounded on models for the dynamics and the noises. Hence, using the preferred time-varying optimization algorithms, our proposed FOPC-based DDSE methods are computationally efficient and flexible to incorporate various types of measurements, as shown by the numerical tests in Section IV.

II. Modeling and Problem Statement

We consider a generic multi-phase unbalanced distribution system with multiphase nodes collected in the set $\mathcal{N} \cup \{0\}$, $\mathcal{N} := \{1, \ldots, N\}$, and distribution line segments collected in the set of edges $\mathcal{E} := \{(m, n)\}$. Node 0 denotes the three-phase slack bus, i.e., the point of connection of the distribution grid with the rest of the electrical system. At each multiphase node, the loads can be either wye- or delta-connected [19], with the number of each type being N^Y, N^Δ respectively.

We briefly introduce the AC power-flow model for multiphase distribution systems (a comprehensive description can be found in [16], [19]). To this end, let N_ϕ denote the total number of single-phase connections; for example, $N_\phi = 3N$ if all the nodes are three-phase. Let $\mathbf{v}$ be a vector collecting the line-to-ground voltages in all phases of the nodes in $\mathcal{N}$; similarly, vector $\mathbf{i}$ collects all the phase net current injections, $\mathbf{i}^\Delta$ the phase-to-phase currents in all the delta connections, and vectors $\mathbf{s}^Y$ and $\mathbf{s}^\Delta$ collect the net complex powers injected for wye- and delta-connected loads, respectively. All of these vectors are of length N_ϕ. With these definitions, the AC power-flow equations can be compactly written as:

$$\mathsf{diag}\Big(\mathbf{H}^\mathsf{T}\big(\mathbf{i}^\Delta\big)^*\Big)\mathbf{v} + \mathbf{s}^Y = \mathsf{diag}(\mathbf{v})\mathbf{i}^*, \tag{1a}$$

$$\mathbf{s}^\Delta = \mathsf{diag}(\mathbf{H}\mathbf{v})(\mathbf{i}^\Delta)^*, \mathbf{i} = \mathbf{Y}_{L0}\mathbf{v}_0 + \mathbf{Y}_{LL}\mathbf{v}, \tag{1b}$$

where $[\mathbf{Y}_{L0}, \mathbf{Y}_{LL}] \in \mathbb{C}^{N_\phi \times (3+N_\phi)}$ is the submatrix of the admittance matrix $\mathbf{Y}$ by eliminating the slack-bus rows, while $\mathbf{H}$ is an $N_\phi \times N_\phi$ block-diagonal matrix mapping $\mathbf{v}$ to line-to-line voltages; see [16] for a detailed description. We leverage a linearized AC power flow model to facilitate the development of computationally affordable algorithms that can be implemented in real time. For any given complex $\mathbf{s}^Y, \mathbf{s}^\Delta$ and its voltage solution $\mathbf{v}$, a fixed-point approximation of (1) has been developed in [16] to obtain a linearized model that exactly touches upon $(\mathbf{s}^Y, \mathbf{s}^\Delta, \mathbf{v})$ and $(\mathbf{0}, \mathbf{0}, \mathbf{w})$, where $\mathbf{w}$ is the zero-load voltage solution to (1). Notice that, relative to alternative linearization techniques (e.g., [20]–[23]), the approach in [16] accounts for both wye and delta connections.

For real-valued notations, define the $2N_\phi \times 1$ rectangular-form voltage vector

$$\mathbf{z} := \Big[\Re\{\mathbf{v}\}^\mathsf{T}, \Im\{\mathbf{v}\}^\mathsf{T}\Big]^\mathsf{T}. \tag{2}$$

For the power variables, similarly define the real-valued vectors $\mathbf{u}^Y$ and $\mathbf{u}^\Delta$ to convert only the non-zero entries in $\mathbf{s}^Y$ and $\mathbf{s}^\Delta$, respectively. Accordingly, consider the following linearized model of (1):

$$\widetilde{\mathbf{z}} = \mathbf{M}^Y\mathbf{u}^Y + \mathbf{M}^\Delta\mathbf{u}^\Delta + \mathbf{m} = \mathbf{M}\mathbf{u} + \mathbf{m} \tag{3}$$

where the model parameters $\mathbf{M}^Y \in \mathbb{R}^{2N_\phi \times 2N_\phi^Y}$, $\mathbf{M}^\Delta \in \mathbb{R}^{2N_\phi \times 2N_\phi^\Delta}$, and $\mathbf{m} \in \mathbb{R}^{2N_\phi \times 1}$, with $\mathbf{M} := [\mathbf{M}^Y, \mathbf{M}^\Delta]$ and $\mathbf{u} := [(\mathbf{u}^Y)^\mathsf{T}, (\mathbf{u}^\Delta)^\mathsf{T}]^\mathsf{T}$. The vector $\mathbf{m} := [\Re\{\mathbf{w}\}^\mathsf{T}, \Im\{\mathbf{w}\}^\mathsf{T}]^\mathsf{T}$ ensures that $(\mathbf{0}, \mathbf{0}, \mathbf{w})$ always satisfies the model (3) for $\mathbf{u} = \mathbf{0}$. In addition, matrices $\mathbf{M}^\Delta$ and $\mathbf{M}^Y$ are computed based on the network parameters and a given voltage profile.

Remark 1 (Power State Variables): The linear model (3) asserts that the power vector $\mathbf{u}$ can uniquely determine the system-wide voltage phasor. Accordingly, $\mathbf{u}$ is used as the unknown states in the ensuring DDSE formulation. This setting is different from conventional transmission SE with voltage phasor as state variables, and the reason is two-fold.

First, the model (3) with $\mathbf{u}$ as unknowns can conveniently include the zero-injection constraints, which are popular in distribution systems. Note that $\mathbf{u}$ only contains buses with non-zero loads, and thus no additional equality constraints are needed to account for zero-injection buses. Second, a voltage-phasor based model would require inverting the linear system in (3), which may incur numerical instability issue. Therefore, the unique characteristics of distribution systems make it more convenient to adopt the power variables as the unknown states.

A. Problem Statement

We introduce the measurement model for our μPMU assisted DDSE problem. Assume that the temporal axis is discretized as $t_k = hk$, where $k = 0, 1, \ldots,$ and the sampling period is $h = t_{k+1} - t_k$. Assume that μPMUs are located at a subset of nodes $\mathcal{M}_{\mathbf{v}} \subset \mathcal{N}$; let $M_{\mathbf{v}}$ denote the number of line-to-ground voltage measurements collected from the multi-phase nodes in $\mathcal{M}_{\mathbf{v}}$. These μPMUs can obtain accurate measurements of the voltages in rectangular coordinates, and they can produce measurements in real time. Furthermore, net injected powers from wye and delta connections are measured at the multi-phase nodes in $\mathcal{M}_{\mathbf{u}}^{Y}$ and $\mathcal{M}_{\mathbf{u}}^{\Delta}$, respectively, with $\mathcal{M}_{\mathbf{u}} := \mathcal{M}_{\mathbf{u}}^{Y} \bigcup \mathcal{M}_{\mathbf{u}}^{\Delta} \subset \mathcal{N}$. Accordingly, let $M_{\mathbf{u}}$ denote the number of nodes in $\mathcal{M}_{\mathbf{u}}$.

In this setting, at each time instant t_k a set of new measurements are collected and processed for DSSE; the interval h can be small – even on the order of seconds – if the fast-acting measurement capabilities of μPMUs and distributed energy resources are leveraged. In particular, (a subset of) the following quantities are *measured* at every time t_k, $k \in \mathbb{N}$:

- $\mathbf{y}_v^{(k)} \in \mathbb{R}^{2M_{\mathbf{v}}\times 1}$: measurements of the line-to-ground voltages at all the phases of the nodes $\mathcal{M}_{\mathbf{v}}$. The measurement model for the μPMUs is $\mathbf{y}_v^{(k)} = \mathbf{z}_{\mathcal{M}_{\mathbf{v}}}^{(k)} + \mathbf{n}_v^{(k)}$ with the measurement noise $\mathbf{n}_v^{(k)}$.
- $\mathbf{y}_u^{(k)} \in \mathbb{R}^{2M_{\mathbf{u}}\times 1}$: measurements of the net active and reactive powers from wye and/or delta connections at nodes $\mathcal{M}_{\mathbf{u}}$. The measurement model is $\mathbf{y}_u^{(k)} = \mathbf{u}_{\mathcal{M}_{\mathbf{u}}}^{(k)} + \mathbf{n}_u^{(k)}$ with the measurement noise $\mathbf{n}_u^{(k)}$.

Using (3), the measurement equation per time t_k is:

$$\begin{bmatrix} \mathbf{y}_v^{(k)} \\ \mathbf{y}_u^{Y(k)} \\ \mathbf{y}_u^{\Delta(k)} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_v\mathbf{M}^{\mathbf{Y}(k)} & \mathbf{J}_v\mathbf{M}^{\Delta(k)} \\ \mathbf{J}^{Y} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}^{\Delta} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{Y(k)} \\ \mathbf{u}^{\Delta(k)} \end{bmatrix} + \begin{bmatrix} \mathbf{m}_v \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{n}_v^{(k)} \\ \mathbf{n}_u^{Y(k)} \\ \mathbf{n}_u^{\Delta(k)} \end{bmatrix} \tag{4}$$

where: $\mathbf{J}_v$ is a suitable permutation matrix which selects rows of $\mathbf{z}$ to form $\mathbf{z}_{\mathcal{M}_{\mathbf{v}}}$; and, similarly for $\mathbf{J}^{Y}$ and $\mathbf{J}^{\Delta}$, selecting the measured loads. Note that $\mathbf{M}^{\mathbf{Y}(k)}$ and $\mathbf{M}^{\Delta(k)}$ are time-variant and the last iterate $\mathbf{z}^{(k)}$ is utilized as the voltage profile for the fixed-point linearization. In our algorithmic development later on, $\mathbf{z}^{(k)}$ is approximated by its estimated value $\widehat{\mathbf{z}}^{(k)}$ at each time step.

Remark 2 (Heterogeneous Measurements): Although we model the DDSE problem for only voltage phasor data from PMUs and power data from pseudo-measurements or smart meters, it can be generalized to encompass a variety of measurements available in distribution systems, as reviewed in [3]. First, the zero-injection constraint is always satisfied by model (4) as only load buses are included by the power variables. Second, as detailed in [16], [24], the linearized model (3) is generalizable to voltage magnitude, line currents, and power flows. Accordingly, real-time measurements of these variables can be included in (4). To fit various sensing frequency of different measurements, one can set h as the fastest sampling time (typically from PMUs) and maintain the values of other slower measurements until new datum arrives.

Next, consider rewriting the measurement model (4) in the following compact form:

$$\mathbf{y}^{(k)} = \mathbf{G}^{(k)}\mathbf{u}^{(k)} + \bar{\mathbf{m}} + \mathbf{n}^{(k)} \tag{5}$$

where $\mathbf{y}^{(k)} := [(\mathbf{y}_v^{(k)})^{\mathsf{T}}, (\mathbf{y}_u^{Y(k)})^{\mathsf{T}}, (\mathbf{y}_u^{\Delta(k)})^{\mathsf{T}}]^{\mathsf{T}}$, $\bar{\mathbf{m}} := [\mathbf{m}_v^{\mathsf{T}}, \mathbf{0}^{\mathsf{T}}, \mathbf{0}^{\mathsf{T}}]^{\mathsf{T}}$, $\mathbf{n}^{(k)} := [(\mathbf{n}_v^{(k)})^{\mathsf{T}}, (\mathbf{n}_u^{Y(k)})^{\mathsf{T}}, (\mathbf{n}_u^{\Delta(k)})^{\mathsf{T}}]^{\mathsf{T}}$, and

$$\mathbf{G}^{(k)} := \begin{bmatrix} \mathbf{J}_v\mathbf{M}^{\mathbf{Y}(k)} & \mathbf{J}_v\mathbf{M}^{\Delta(k)} \\ \mathbf{J}^{Y} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}^{\Delta} \end{bmatrix}. \tag{6}$$

The data $\mathbf{y}_u$ is usually collected or generated at much lower quality compared to the high-resolution μPMU data. In fact, $\mathbf{y}_u$ is either collected from meters or generated from historic load information. Thus, to account for different granularities and precisions, we define the instantaneous error mismatch loss function $\ell^{(k)}(\mathbf{z})$ at time t_k as:

$$\ell^{(k)}(\mathbf{u}) := \frac{1}{2}\left\|\mathbf{y}_v^{(k)} - \mathbf{G}_v^{(k)}\mathbf{u} - \mathbf{m}_v\right\|_2^2 + \mathsf{L}\left(\mathbf{y}_u^{Y(k)} - \mathbf{J}^{Y}\mathbf{u}^{Y}\right) + \mathsf{L}\left(\mathbf{y}_u^{\Delta(k)} - \mathbf{J}^{\Delta\mathbf{u}^{\Delta}}\right) \tag{7}$$

with $\mathbf{G}_v^{(k)} := [\mathbf{J}_v\mathbf{M}^{\mathbf{Y}(k)}, \mathbf{J}_v\mathbf{M}^{\Delta(k)}]$, and the Huber loss function $\mathsf{L}(\boldsymbol{\epsilon}) := \sum_i \mathsf{L}_i(\epsilon_i)$ written as

$$\mathsf{L}_i(\epsilon_i) := \begin{cases} -\delta\epsilon_i - \delta^2/2, & \text{if } \epsilon_i < -\delta \\ |\epsilon_i|^2/2, & \text{if } |\epsilon_i| \le \delta \\ \delta\epsilon_i - \delta^2/2, & \text{if } \epsilon_i > \delta \end{cases} \tag{8}$$

where $\delta > 0$ is a positive parameter determined by the load data quality. The Huber loss function is utilized to reject possible outliers, or down-weight data with substantial measurement errors [25].

Remark 3 (Weighted Error Objective): To accommodate varying data quality, *non-uniform weights* can be assigned to different types of measurements. With the high accuracy of voltage data from PMUs, one can use a large positive weight for the voltage error term in (4). Note that this weighted error objective does not affect the problem structure and thus is not included specifically by the algorithmic developments.

Using the error mismatch $\ell^{(k)}$, we formulate the following state estimation problem at time t_k:

$$\mathsf{P}^{(k)}(\mathbf{u}): \min_{\mathbf{u}\in\mathbb{R}^{N_\phi}} f^{(k)}(\mathbf{u}) := \ell^{(k)}(\mathbf{u}) + r^{(k)}(\mathbf{u}), k \in \mathbb{N} \tag{9}$$

where $r^{(k)}(\mathbf{u})$ is a (possibly time-varying) regularization function that renders the overall cost function $f^{(k)}(\mathbf{u})$ globally

strongly convex. Notice further that the problem (9) is unconstrained; however, possible prior information on the minimum and maximum values of the vector $\mathbf{u}$ can be naturally incorporated in the proposed approach.

Problem (9) models a *time-varying* state estimation task under streaming of measurements, and it implicitly defines an *optimal trajectory* $\{\mathbf{u}^{(k,\star)}\}_{k\in\mathbb{N}}$ for the estimation task. Accordingly, the objective here is to track $\{\mathbf{u}^{(k,\star)}\}_{k\in\mathbb{N}}$ by processing the incoming measurements in real time. One way to obtain $\{\mathbf{u}^{(k,\star)}\}_{k\in\mathbb{N}}$ is to solve the problem $\mathsf{P}^{(k)}(\mathbf{u})$ *to convergence* (i.e., a batch solution) at each time step t_k. However, in a real-time setting with an asynchronous streaming of measurements, a batch solution of (9) might not be achievable within an interval h due to underlying communication and computational complexity requirements. Thus, the objective of the paper is to develop an algorithmic solution to generate a sequence $\{\mathbf{u}^{(k)}\}$ of approximate optimizers for the time-varying problem $\{\mathsf{P}^{(k)}(\mathbf{u})\}$, which eventually converges to the optimization trajectory $\{\mathbf{u}^{(k,\star)}\}$. Accordingly, the next section presents a running prediction-correction method to solve (9) in real time. Before doing so, a remark on the strong convexity of $\{\mathsf{P}^{(k)}(\mathbf{u})\}$ is in order.

Strong convexity can ensure that the optimizer trajectory for the sequence of problems $\{\mathsf{P}^{(k)}(\mathbf{u})\}$ is unique. In addition, it allows us to establish convergence for the proposed online algorithms. Regarding the regularization function, possible choices are exemplified next:

i) $r^{(k)}(\mathbf{u}) = \frac{a}{2}\|\mathbf{u} - \mathbf{u}_{\text{pr}}^{(k)}\|_2^2$, $a > 0$ and where $\mathbf{u}_{\text{pr}}^{(k)}$ is a priori guess on the load profile;

ii) $r^{(k)}(\mathbf{u}) = \frac{a}{2}\|\mathbf{u}\|_2^2$, and it is time invariant;

iii) if the Hessian of the cost function is available, or, a subspace tracking method is in place, the regularization function can be set to $r^{(k)}(\mathbf{u}) = \frac{a}{2}\mathbf{u}^{\mathsf{T}}\mathbf{Q}^{(k)}\mathbf{u}$, where the positive eigenvalues of the matrix $\mathbf{Q}^{(k)}$ are in the null space of the Hessian of $\ell^{(k)}(\mathbf{u})$. The options i) and ii) would involve a deviation from optimal solutions that one would have obtained by minimizing $\ell^{(k)}(\mathbf{u})$, with the magnitude of a possible deviation dependent on the parameter a; see [26]. The option iii) would not perturb the optimal solution, but the overall solution would incur a higher computational complexity.

III. Dynamic State Estimation

We consider *First-Order Prediction-Correction (FOPC)* method [18] to solve the DDSE problem at hand. Inspired by Kalman filtering approaches, prediction-correction approaches allow one to solve a broad class of time-varying convex optimization objectives in a dynamic setting by involving two stages: a prediction phase and a correction phase. In the prediction phase, the algorithm attempts to approach the optimal solution of the next time period (without new observations) by tapping into intrinsic temporal correlations of the cost function; on the other hand, in the correction phase the predicted vector is corrected using the latest measurement. This mechanism improves the response time to external dynamics and shows a good convergence result when the objective function changes smoothly over time, which is the case in the DDSE setting.

A. FOPC Algorithm

Consider a continuously time-varying unconstrained optimization with objective $f(\boldsymbol{x}; t)$ to model external dynamics, along with the sampled counterpart $\{f^{(k)} := f(\boldsymbol{x}; t_k)\}$ for $t_k = kh$, $k \in \mathbb{N}$. The goal is to produce a trajectory $\{\boldsymbol{x}^{(k)}\}$ such that $\boldsymbol{x}^{(k)} \approx \boldsymbol{x}^*(t_k)$, where $\boldsymbol{x}^*(t_k)$ denotes the optimal solution at time t_k. In order to predict the solution at time t_{k+1}, a strategy is to find $\boldsymbol{x}^{(k+1|k)}$ that satisfies the condition

$$\nabla_x f^{(k+1)}\Big(\boldsymbol{x}^{(k+1|k)}\Big) = (1-\gamma)\nabla_x f^{(k)}\Big(\boldsymbol{x}^{(k)}\Big), \tag{10}$$

with $\gamma \in [0, 1]$. Varying γ, this condition imposes optimality ($\gamma = 1$), or the fact that the estimate $\boldsymbol{x}^{(k+1|k)}$ is no worse than $\boldsymbol{x}^{(k)}$ in terms of suboptimality ($\gamma = 0$) even when the function changes. The choice $\gamma = 1$ combines moving towards the optimizer while moving with the varying objective function; the choice $\gamma = 0$ represents a *rigid* motion with the objective.

Notwithstanding the choice of γ, condition (10) cannot be computed at time t_k without information about $f^{(k+1)}$. Instead, consider the following Taylor approximation:

$$\begin{aligned}(1-\gamma)\nabla_x f^{(k)}(\boldsymbol{x}^{(k)}) = \nabla_x f^{(k+1)}\Big(\boldsymbol{x}^{(k+1|k)}\Big) \approx \nabla_x f^{(k)}\Big(\boldsymbol{x}^{(k)}\Big) \\ + \nabla_{xx} f^{(k)}\Big(\boldsymbol{x}^{(k)}\Big)\Big(\boldsymbol{x}^{(k+1|k)} - \boldsymbol{x}^{(k)}\Big) \\ + h\nabla_{tx} f^{(k)}\Big(\boldsymbol{x}^{(k)}\Big).\end{aligned} \tag{11}$$

By solving this equation, we have a recursion of the form:

$$\begin{aligned}\boldsymbol{x}^{(k+1|k)} = \boldsymbol{x}^{(k)} - \Big[\nabla_{xx} f^{(k)}(\boldsymbol{x}^{(k)})\Big]^{-1} \\ \times \Big(\gamma\nabla_x f^{(k)}\Big(\boldsymbol{x}^{(k)}\Big) + h\nabla_{tx} f^{(k)}\Big(\boldsymbol{x}^{(k)}\Big)\Big),\end{aligned} \tag{12}$$

which (as anticipated) combines a Newton's step with a rigid motion with the objective function.

Concerning over the cost of computing the inverse of the Hessian in a possibly small time interval h, the FOPC method further involves a first-order update to solve (11). Specifically, the prediction solution of (11) is sought by constructing an equivalent quadratic optimization as follows

$$\boldsymbol{x}^{(k+1|k)} = \text{argmin}_{\mathbf{x}} \quad \widehat{f}^{(k)}(\mathbf{x}) \tag{13}$$

where

$$\begin{aligned}\widehat{f}^{(k)}(\mathbf{x}) := \frac{1}{2}\mathbf{x}^{\mathsf{T}}\nabla_{xx} f^{(k)}\Big(\boldsymbol{x}^{(k)}\Big)\mathbf{x} + \Big(\gamma\nabla_x f^{(k)}(\boldsymbol{x}^{(k)}) \\ - \nabla_{xx} f^{(k)}(\boldsymbol{x}^{(k)})\boldsymbol{x}^{(k)} + h\nabla_{tx} f^{(k)}\Big(\boldsymbol{x}^{(k)}\Big)\Big)^{\mathsf{T}}\mathbf{x}.\end{aligned} \tag{14}$$

Thus, one can replace the update (12) with the gradient descent solution for (14); i.e., each iteration p is given by:

$$\begin{aligned}\widehat{\boldsymbol{x}}_{p+1} = \widehat{\boldsymbol{x}}_p - \alpha\Big[\nabla_{xx} f^{(k)}(\boldsymbol{x}_k)(\widehat{\boldsymbol{x}}_p - \boldsymbol{x}^{(k)}) + \gamma\nabla_x f^{(k)}(\boldsymbol{x}_k) \\ + h\nabla_{tx} f^{(k)}(\boldsymbol{x}_k)\Big], \quad p = 0, \ldots, P-1\end{aligned} \tag{15}$$

where integer P is the number of *prediction steps* that one can afford within an interval h, and $\alpha > 0$ the stepsize to be designed later on. Hence, the predicted solution is set to $x^{(k+1|k)} = \widehat{\boldsymbol{x}}_P$.

Initializing at $x^{(k+1|k)}$, the correction phase further involves C first-order gradient steps; that is, for each $c = 0, \ldots, C-1$

$$\widehat{\boldsymbol{x}}_{c+1} = \widehat{\boldsymbol{x}}_c - \beta\nabla_x f^{(k+1)}(\widehat{\boldsymbol{x}}_c)\Big(\widehat{\boldsymbol{x}}_c - \boldsymbol{x}^{(k)}\Big) \tag{16}$$

Algorithm 1 FOPC for DDSE

Notation: $\widehat{\mathbf{u}}^{(k)}$ and $\widehat{\mathbf{u}}^{(k|k-1)}$ are the estimate/prediction of $\mathbf{u}^{(k)}$ at time t_k and t_{k-1} respectively, $k \in \mathbb{N}$.

Initialization: Choose the number of prediction and correction steps P, C, the stepsizes α, β, the parameter $\gamma \in [0, 1]$, the parameter δ of the Huber loss, the regularizer r. Set $\mathbf{m} = [\Re\{\mathbf{w}\}^\mathsf{T}, \Im\{\mathbf{w}\}^\mathsf{T}]^\mathsf{T}$ and $\mathbf{u}^{(0)} = \mathbf{0}$.

Algorithm: for $k = 0, 1, 2, \cdots$, perform:

At time t_{k-1}:

Prediction step:

[S1-0] When $k = 0$, set $\widehat{\mathbf{u}}^{(0|-1)} = \mathbf{u}_0, \widehat{\mathbf{z}}^{(0|-1)} = \mathbf{m}$ and skip the following prediction.

[S1-1] Set $\bar{\mathbf{u}}_0 = \widehat{\mathbf{u}}^{(k-1)}$.

[S1-2] For $p = 0, \ldots, P-1$, do:

$$\bar{\mathbf{u}}_{p+1} := \bar{\mathbf{u}}_p - \alpha\Big[\Big(\nabla_{\mathbf{uu}} f^{(k-1)}\Big(\widehat{\mathbf{u}}^{(k-1)}\Big)\Big)(\bar{\mathbf{u}}_p - \widehat{\mathbf{u}}^{(k-1)}) + h\nabla_{t\mathbf{u}} f^{(k-1)}\Big(\widehat{\mathbf{u}}^{(k-1)}\Big) + \gamma\Big(\nabla_{\mathbf{u}} f^{(k-1)}(\widehat{\mathbf{u}}^{(k-1)})\Big)\Big] \tag{17}$$

[S1-3] Set $\widehat{\mathbf{u}}^{(k|k-1)} = \bar{\mathbf{u}}_P$ and compute

$$\widehat{\mathbf{z}}^{(k|k-1)} = \Big[\mathbf{M}^{\mathbf{Y}(k-1)}\ \mathbf{M}^{\Delta(k-1)}\Big]\widehat{\mathbf{u}}^{(k|k-1)} + \mathbf{m}. \tag{18}$$

At time t_k:

Function update:

[S2-1] Compute the updated AC linear model $\mathbf{M}^{\Delta(k)}$ and $\mathbf{M}^{\mathbf{Y}(k)}$ using the current estimate of voltage, *i.e.*, $\widehat{\mathbf{z}}^{(k|k-1)}$. This is used to update the linear model $\mathbf{G}^{(k)}$ in (5).

[S2-2] Update $f^{(k)}$ from $f^{(k-1)}$ using $\mathbf{G}^{(k)}$ and the new observations $\mathbf{y}^{(k)}$.

Corrections step:

[S3-1] Set $\bar{\mathbf{u}}_0 = \widehat{\mathbf{u}}^{(k|k-1)}$.

[S3-2] For $c = 0, \ldots, C-1$, do:

$$\bar{\mathbf{u}}_{c+1} := \bar{\mathbf{u}}_c - \beta\nabla_{\mathbf{u}} f^{(k)}(\bar{\mathbf{u}}_c) \tag{19}$$

[S3-3] Set $\widehat{\mathbf{u}}^{(k)} = \bar{\mathbf{u}}_C$ and compute

$$\widehat{\mathbf{z}}^{(k)} = \Big[\mathbf{M}^{\mathbf{Y}(k)} \quad \mathbf{M}^{\Delta(k)}\Big]\widehat{\mathbf{u}}^{(k)} + \mathbf{m}. \tag{20}$$

with the stepsize $\beta > 0$. The final corrected estimate is set as $\boldsymbol{x}^{(k+1)} = \widehat{\boldsymbol{x}}_C$. The number of steps P and C for the prediction and correction stages, respectively, are selected based on the duration of the interval h.

The complete running FOPC algorithm for the DDSE problem (9) is tabulated as Algorithm 1. Selection strategies for P, C as well as the stepsizes α and β will be elaborated in the next subsection. Notice that, once the prediction $\widehat{\mathbf{u}}^{(k|k-1)}$ and the corrected estimate $\widehat{\mathbf{u}}^{(k)}$ are obtained, the voltage vectors can be readily calculated using the linearized model (3). Alternatively, given the estimate of the power injections, the voltages can be calculated by solving the AC power flow equations (see, e.g., the fixed-point power flow method in [16]).

B. Online Tracking Results

This subsection describes in which sense the sequence of approximate optimizers $\{\widehat{\mathbf{u}}^{(k)}\}_{k\in\mathbb{N}}$ generated by Algorithm 1 *tracks* the sampled solution trajectory $\mathbf{u}^{(*,k)}$. To this end, we will adapt some of the results of [18] to the DDSE problem.

The following assumptions on the cost function of (9) and its time variations are presupposed.

Assumption 1: The cost function $f^{(k)}(\mathbf{u})$ is ν-strongly convex for all $k \in \mathbb{N}$. The regularized function $r^{(k)}(\mathbf{u})$ is L_r-strongly smooth for all $k \in \mathbb{N}$.

Assumption 2: There exists a constant $C_0 < +\infty$ so that the time variation of the gradient of the cost $f^{(k)}(\mathbf{u})$ can be bounded as

$$\Big\|\nabla_{t\mathbf{u}} f^{(k)}(\mathbf{u})\Big\| \le C_0, \quad \text{for all } k \in \mathbb{N}. \tag{21}$$

Assumption 1 guarantees that the solution trajectory is unique. This is why one may need a regularizer $r^{(k)}(\mathbf{u})$. As explained in Section II, this assumption can be verified by utilizing a suitable regularization function. Assumption 2 makes sure that the functional changes are bounded and thus it is possible to bound the errors arising from a time-varying problem. Recalling that the cost function $f^{(k)}$ may change over time because of the processing of a different datum $\mathbf{y}^{(k)}$, possible changes in the linearized model $\mathbf{G}^{(k)}$, or changes in the cost function (for example, tuning of the Huber function), the assumption simply states that the temporal variation of the gradient of the function is finite. The temporal variation is not finite only when the function $f^{(k)}(\mathbf{u})$ has a jump discontinuity for a given k; this is not the case for the cost function proposed in the paper, as well as traditional state estimation problems in power systems.

Note that, in the sequel, the exact value C_0 will not be needed for determining the parameters such as stepsizes, but we will only need the bounded condition.

First of all, we show that the Hessian of the cost function is lower and upper bounded uniformly in time, as follows.

Proposition 1: Under Assumption 1, the Hessian $\nabla_{\mathbf{uu}} f^{(k)}(\mathbf{u})$ of the cost function (9) is lower and upper bounded uniformly in time as

$$\nu \le \Big\|\nabla_{\mathbf{uu}} f^{(k)}(\mathbf{u})\Big\| \le L, \quad \text{for all } k \in \mathbb{N}. \tag{22}$$

Proof: The lower bound follows directly from Assumption 1. The upper bound follows by direct computation. The Hessian of $f^{(k)}(\mathbf{u})$ is bounded as

$$\begin{aligned}\Big\|\nabla_{\mathbf{uu}} f^{(k)}(\mathbf{u})\Big\| &= \Big\|\nabla_{\mathbf{uu}}\ell^{(k)}(\mathbf{u}) + \nabla_{\mathbf{uu}} r^{(k)}(\mathbf{u})\Big\| \\ &\le \Big\|\mathbf{G}_v^{(k)\mathsf{T}}\mathbf{G}_v^{(k)}\Big\| + \Big\|\mathbf{J}^{Y\mathsf{T}}\mathbf{J}^{Y}\Big\| + \Big\|\mathbf{J}^{\Delta\mathsf{T}}\mathbf{J}^{\Delta}\Big\| + L_r.\end{aligned}$$

By properly defining L as the upper bound of the right-hand term, the Hessian is upper bounded. ∎

We are now ready for the online tracking result.

Theorem 1 (Adapted from [18, Th. 3]): Consider the sequence $\{\widehat{\mathbf{u}}^{(k)}\}_{k\in\mathbb{N}}$ generated by Algorithm 1, and let Assumptions 1-2 hold true. Let $\mathbf{u}^{(*,k)}$ be the optimizer of (9) at time t_k. Choose stepsizes α and β as

$$\alpha < 2/L, \quad \beta < 2/L, \tag{23}$$

and define the following non-negative quantities

$$\varrho_{\mathrm{P}} = \max\{|1-\alpha\nu|, |1-\alpha L|\},$$
$$\varrho_{\mathrm{C}} = \max\{|1-\beta\nu|, |1-\beta L|\}. \tag{24}$$

Further, select the number of correction steps C in a way that

$$\tau_0 \coloneqq \varrho_{\mathrm{C}}^{C}\left[\varrho_{\mathrm{P}}^{P} + \left(\varrho_{\mathrm{P}}^{P}+1\right)\left(1-\gamma+\gamma\frac{2L}{\nu}\right)\right] < 1. \tag{25}$$

Then, the sequence $\{\|\mathbf{u}^{(k)}-\mathbf{u}^{(*,k)}\|\}_{k\in\mathbb{N}}$ converges linearly with rate τ_0 to an asymptotic error bound, and

$$\limsup_{k\to\infty}\left\|\mathbf{u}^{(k)}-\mathbf{u}^{(*,k)}\right\| = O\left(\varrho_{\mathrm{C}}^{C}h\right). \tag{26}$$

Theorem 1 asserts that the sequence $\{\widehat{\mathbf{u}}^{(k)}\}_{k\in\mathbb{N}}$ generated by Algorithm 1 *converges to* (and tracks) the sampled solution trajectory $\mathbf{u}^{(*,k)}$ up to an asymptotic bound. This bound is linearly related to the sampling period h and exponentially decreasing with C. Theorem 1 requires standard bounds on the stepsizes (23) and a condition on the number of prediction and correction steps (25).

If one chooses the parameter $\gamma = 0$ (i.e., rigid motion only), then the second condition boils down to

$$\varrho_{\mathrm{C}}^{C}\left[2\varrho_{\mathrm{P}}^{P}+1\right] < 1. \tag{27}$$

Since both ϱ_{P} and ϱ_{C} are less than 1 by construction, the condition (27) is not very restrictive. In fact, fixing a level of prediction $\bar{P}$, then the number of correction steps one has to perform is

$$C \geq \left\lceil -\frac{\log\left(2\varrho_{\mathrm{P}}^{\bar{P}}+1\right)}{\log\varrho_{\mathrm{C}}}\right\rceil. \tag{28}$$

For reasonable values such as $\varrho_{\mathrm{C}} = \varrho_{\mathrm{P}} = 0.8$ and $\bar{P} = 4$, it implies that $C \geq 3$ is sufficient. When $\gamma > 0$, Condition (25) gets more restrictive, while the tracking error accuracy may benefit from the Newton's step in (12).

The choice of γ is important to trade-off convergence region, requirements for prediction and correction steps, and conditioning on the measurement matrix. On one hand, if $\gamma = 0$, then the convergence region is bigger, Condition (25) is less restrictive, which is good when one can afford only a small number of prediction and correction steps (in a fast sampling scenario); however $\gamma = 0$ could be more affected by a badly conditioned measurement matrix. On the other hand, if $\gamma = 1$, you get a Newton step in the prediction that helps in case the measurement matrix is badly conditioned, the convergence region gets smaller, and the number of required prediction and correction steps gets higher.

More insights could be provided by the exact form of the asymptotic error $O(\varrho_{\mathrm{C}}^{C}h)$ in Theorem 1, as given by [18]

$$4\frac{h\varrho_{\mathrm{C}}^{C}}{1-\tau_0}\frac{C_0}{\nu}(1+L/\nu)\left(2\varrho_{\mathrm{P}}^{P}+1\right). \tag{29}$$

Naturally, when $C\to\infty$, the correction phase is equivalent to the batch static estimation mode and thus the error would go to 0. The worse conditioned the problem gets (i.e., small ν and big L/ν), the higher the error is. Meanwhile, the faster the problem varies (that is, for large values of C_0), the greater the error gets. In the DDSE context, the error scales with the external dynamics (C_0), and the measurement matrix conditioning (which yields L).

Remark 4 (First-Order Backward Finite Difference): Algorithm 1 requires one to compute $\nabla_{\mathbf{u}}f^{(k)}(\widehat{\mathbf{u}}^{(k)})$. Indeed, a model for time derivative of the gradient $\nabla_{\mathbf{u}}f^{(k)}(\widehat{\mathbf{u}}^{(k)})$ might not be available. Therefore, one can obtain an approximation $\widehat{\nabla}_{t\mathbf{u}}f^{(k)}(\widehat{\mathbf{u}}^{(k)})$ of $\nabla_{t\mathbf{u}}f^{(k)}(\widehat{\mathbf{u}}^{(k)})$ via first-order backward finite difference, which is given by:

$$\widehat{\nabla}_{t\mathbf{u}}f^{(k)}(\mathbf{u}) = \frac{1}{h}\left(\nabla_{\mathbf{u}}f^{(k)}(\mathbf{u}) - \nabla_{\mathbf{u}}f^{(k-1)}(\mathbf{u})\right). \tag{30}$$

The first-order backward finite difference requires information of the previous step; the approximation error is bounded on the order of $O(h)$, as shown in [27].

Remark 5 (Projections to Convex Sets): In a prior information in the set of admissible $\mathbf{u}^{(k)}$ is available, the correction Algorithm 1 can be modified to accommodate a projection into convex sets. Letting $\mathcal{X}$ be a convex set, then the prediction step is modified as:

$$\begin{aligned}\bar{\mathbf{u}}_{p+1} \coloneqq \mathrm{proj}_{\mathcal{X}}\Big\{\bar{\mathbf{u}}_p - \alpha\Big[&\left(\nabla_{\mathbf{u}\mathbf{u}}f^{(k-1)}\left(\widehat{\mathbf{u}}^{(k-1)}\right)\right)\left(\bar{\mathbf{u}}_p-\widehat{\mathbf{u}}^{(k-1)}\right)\\ &+ h\nabla_{t\mathbf{u}}f^{(k-1)}\left(\widehat{\mathbf{u}}^{(k-1)}\right)\\ &+ \gamma\left(\nabla_{\mathbf{u}}f^{(k-1)}\left(\widehat{\mathbf{u}}^{(k-1)}\right)\right)\Big]\Big\}\end{aligned} \tag{31}$$

where $\mathrm{proj}_{\mathcal{X}}(\mathbf{y}) = \arg\min_{\mathbf{u}\in\mathcal{X}}\|\mathbf{u}-\mathbf{y}\|_2$ is the projection operator. On the other hand, the correction step is modified as:

$$\bar{\mathbf{u}}_{c+1} \coloneqq \mathrm{proj}_{\mathcal{X}}\left\{\bar{\mathbf{u}}_c - \beta\nabla_{\mathbf{u}}f^{(k)}(\bar{\mathbf{u}}_c)\right\}. \tag{32}$$

The results of the Theorem 1 are applicable to the case with projections.

IV. Numerical Simulations

Numerical tests have been performed using the IEEE 37-bus and 123-bus test feeders (see, e.g., [28] for a description of the feeders) on a standard laptop with Intel Core i7-7500 CPU @2.70Hz. In the 37-bus test feeder, only 32 nodes (phases) are connected to non-zero loads, all delta-connected. The 123-bus test feeder is a popular case with 72 non-zero load nodes and various single-, two-, and three-phase lines, with a mix of delta and wye connections. The load profiles were generated from a real dataset that the National Renewable Energy Laboratory produced from real consumption data received from a utility company in California; the data includes 55 load consumption trajectories over the course of 24-hour, at a time-resolution of 6 seconds. The trajectories for the active power are described in more details in [29]; 5 representative trajectories are shown in Fig. 1. A constant power factor of 0.95 has been postulated to create the trajectories for the reactive power. The load profiles were randomly chosen for each load node.

As for the measurement settings, the DDSE algorithm used both the load power data and PMU voltage data. We assume the load active/reactive power injections are observed everywhere in the system. Nonetheless, since they are typically collected by smart meters, the load profiles in Fig. 1 were downsampled to generate the measurement data at a slower

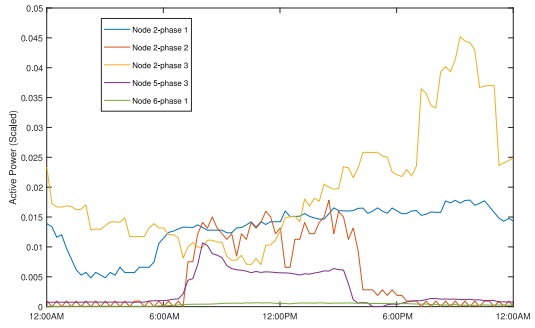

Fig. 1. Five sample active-power trajectories from the 24-hour load profile dataset.

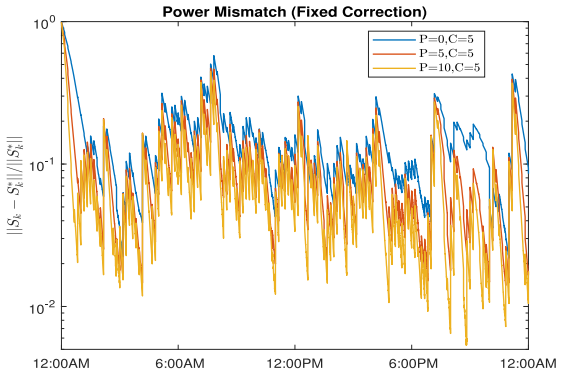

Fig. 2. Relative error of tracking the instantaneous optimal system-wide power state under fixed C steps.

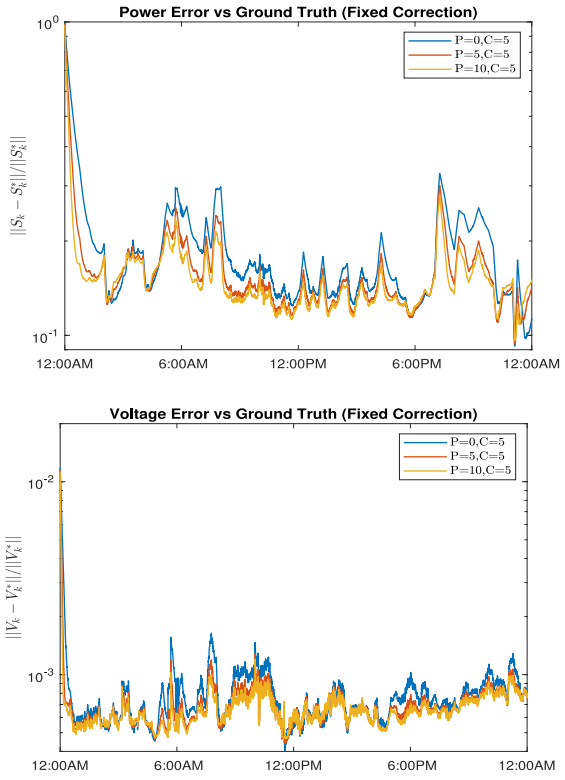

Fig. 3. Relative error of estimating the ground-truth system-wide power state (top) and voltage output (bottom) under fixed C steps.

time resolution of 10 minutes. Specifically, the power measurements were kept as the average value within every 10-minute window. Accordingly, the parameter for the Huber's loss function in (8) was chosen to be $\delta = 8e{-}3$, coinciding with the maximum deviation of the down-sampled power measurement from the actual one. Since PMUs are not installed everywhere, only selected buses are equipped with voltage phasor data at the fast resolution of 6 seconds, same to the original load data. For the 37-bus feeder, 5 PMUs are used except for the last test with varying number of PMUs from 0 to 5. The 123-bus feeder has 10 PMUs. For both feeders, the PMU locations are selected as the connection points on the feeder to major laterals, same to the typical placement for distribution-level monitoring/protection devices. Random Gaussian noises with a small standard deviation of $2e{-}5$ were added to the actual voltage profiles. This noise setting coincides with a maximum total vector error of 0.01% for the voltage phasor, according to the data sheet for μPMUs [30].

A. 37-Bus Test Feeder

This test feeder was used to compare the FOPC-based DDSE algorithm under different computational settings, with the performance in terms of both the tracking error with the instantaneous optimum and estimation error with the actual voltage state. For simplicity of implementation, the regularization term in (9) was chosen to be $r(\mathbf{u}) = 0.1\|\mathbf{u}\|_2$. This sufficiently small term guarantees convergence and yet does not degrade significantly the performance of the estimation problem. To enhance the effectiveness of the prediction step under potentially ill-conditioning issue (see Section III), we set the parameter $\gamma = 0.9$ to be close to 1. Based on these parameter settings, the norm of the Hessian matrix as in Prop. 1 can be bounded within the interval $[5,\ 10]$. Thus, the step-size parameters α and β were chosen to be very small around $5e{-}2$. according to Theorem 1. Accordingly, the two parameters therein, ϱ_{P} and ϱ_{C}, are around 0.75, both less than 1. To satisfy the convergence condition in (25), It turns out $C = 5$ steps of correction is sufficient even if no prediction is performed ($P = 0$), as $\tau_0 \approxeq 0.5$ in this case.

1) Fixed Number of Correction Steps: We first show the advantage to add prediction phase before new measurements are processed. As mentioned in Section I, FOPC can perform a prediction of the state while waiting for the measurement to be transmitted from the PMUs; once the measurement is received, a fixed number of correction steps can be performed. Thus, this test uses a fixed $C = 5$ steps of correction and compares the results from $P = 0, 5, 10$ steps of prediction.

Fig. 2 plots the relative tracking error with the instantaneous optimal state $\mathbf{u}^{(*,k)}$. Clearly, a larger number of P does help the tracking of the optimal solution, with slightly more noticeable change in mismatch error from $P = 0$ to $P = 5$ steps of prediction. It also makes the relative error trajectory more quickly to reach the steady-state level of below 0.1. Hence, the prediction phase has been shown to improve the tracking error performance for the time-varying DDSE problem.

Furthermore, we compare the relative estimation errors of both the system-wide power states and corresponding voltage outputs with ground-truth values, as plotted in Fig. 3. As

TABLE I
SUMMARY OF AVERAGE RELATIVE ESTIMATION ERROR WITH THE GROUND-TRUTH VALUES FOR DIFFERENT PAIRS OF (P, C) VALUES

	$(0,5)$	$(5,5)$	$(10,5)$	$(0,6)$	$(8,3)$
Avg V Err	1.02e-3	0.95e-3	0.93e-3	0.99e-3	0.95e-3
Avg S Err	1.78e-1	1.51e-1	1.42e-1	1.69e-1	1.50e-1

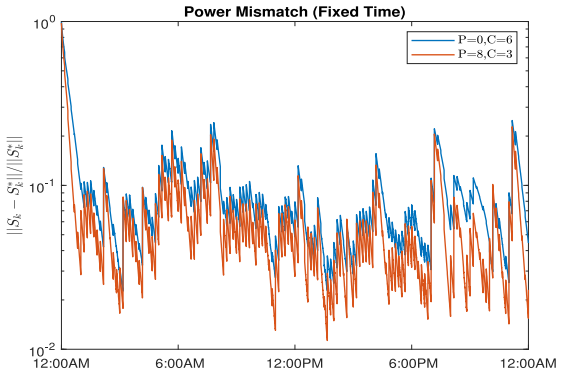

Fig. 4. Relative error of tracking the instantaneous optimal system-wide power state under a fixed computational time.

for the power states, the relative estimation error increases slightly from the tracking error level. This could be due to the approximation error of the linearized model and the additional measurement error. Meanwhile, the effectiveness of prediction phase is still evident in improving the estimation error and convergence rate. More interestingly, the voltage estimation error is very minimal at the level of $1e-3$ for all scenarios, corroborating that the PMU voltage data is instrumental for recovering the feeder voltage profile even under highly uncertain power data. The average relative power/voltage estimation error for all scenarios in Fig. 3 is listed in Table I, showing the consistent improvement with more prediction steps.

2) Fixed Computational Time: This test compares the FOPC performance under a total computational time constraint. The computational time in the prediction phase is mainly spent on a one-shot computation of Hessian $\nabla_{\mathbf{uu}} f^{(k)}(\widehat{\mathbf{u}}^{(k)})$ in (17), while that of the correction phase grows linearly with C as it needs to compute the gradient in every step. We choose two sets of (P, C) values: $(8, 3)$ and $(0, 6)$, both taking roughly a total of 0.3ms per iteration.

Similar to the last test, Fig. 4 and Fig. 5 plot the relative tracking and estimation error trajectories for both scenarios, respectively. Interestingly, the improvement of including the prediction phase can be demonstrated even if C has to be very small to compensate for the one-shot Hessian computation. Both the tracking and estimation errors for the power states are lower for the case of $(P, C) = (8, 3)$. The voltage error is again very small thanks to the high-quality PMU data. Again, the average relative power/voltage estimation error for both scenarios in Fig. 5 is listed in Table I.

3) Varying Number of PMUs: We further compare the performance when different number of PMUs are installed in the system, by reducing it from 5 PMUs (as in previous tests) to 0, for fixed $(P, C) = (5, 5)$. Fig. 6 plots the relative error of estimating both the ground-truth power and

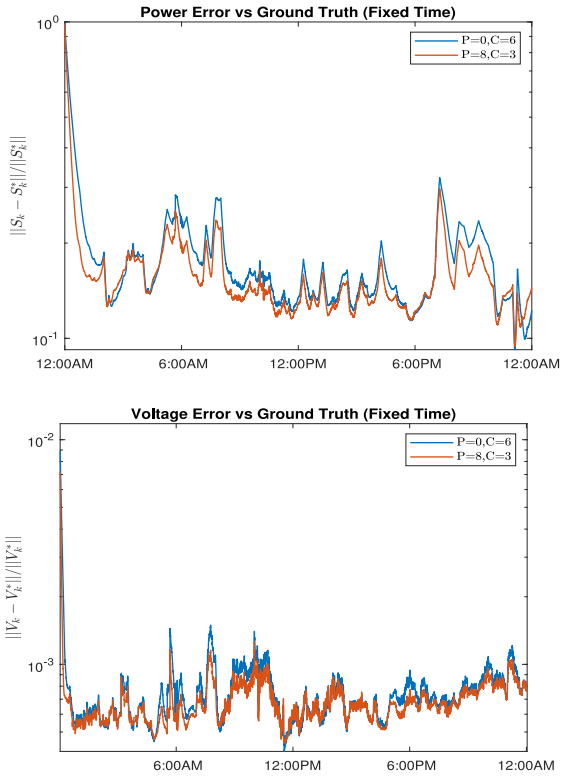

Fig. 5. Relative error of estimating the ground-truth system-wide power state (top) and voltage output (bottom) under a fixed computational time.

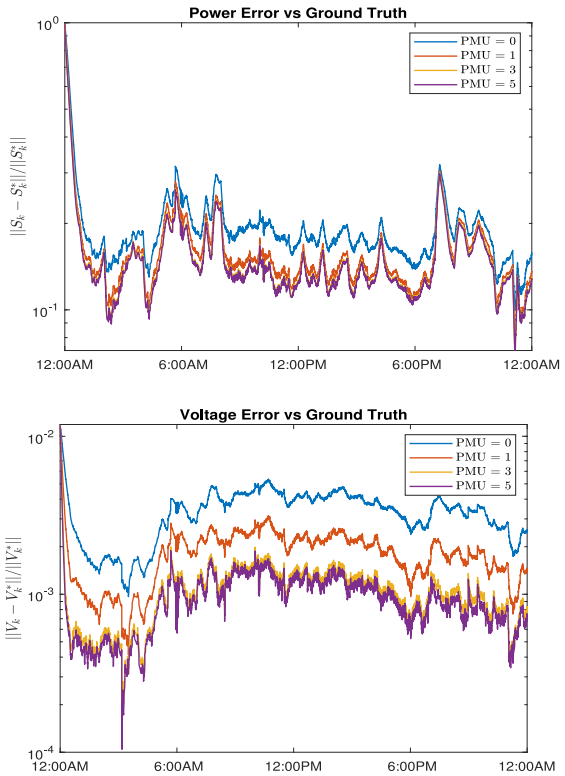

Fig. 6. Relative error of estimating the ground-truth system-wide power state (top) and voltage output (bottom) under varying number of PMUs.

voltage variables. Clearly, more high-resolution voltage data can improve the estimation error performance. For the 37-bus case, it seems that 3 PMUs are sufficiently good for estimating

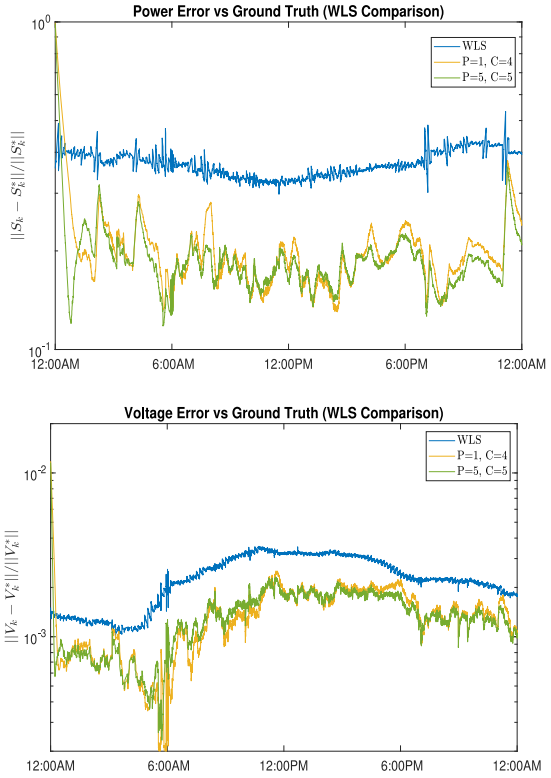

Fig. 7. Relative error of estimating the ground-truth system-wide power state (top) and voltage output (bottom) compared to the WLS solution under large power measurement noises.

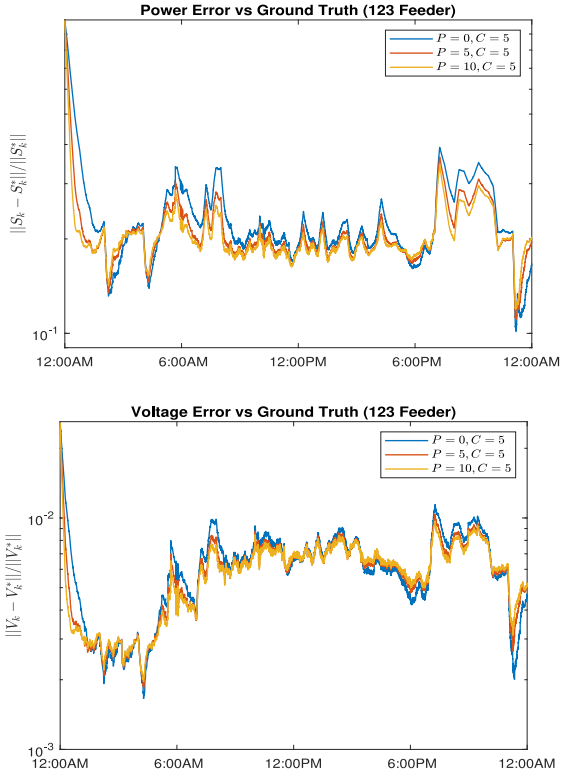

Fig. 8. Relative error of estimating the ground-truth system-wide power state (top) and voltage output (bottom) of the 123-bus case for a fixed C.

TABLE II
SUMMARY OF AVERAGE COMPUTATION TIME AND AVERAGE RELATIVE ESTIMATION ERROR WITH THE GROUND-TRUTH VALUES FOR FOPC AND WLS METHODS

	WLS	FOPC $(1, 4)$	FOPC $(5, 5)$
Avg Comp Time	0.25ms	0.25ms	0.40ms
Avg V Err	2.33e-3	1.38e-3	1.34e-3
Avg S Err	3.71e-1	2.06e-1	1.92e-1

the system-wide voltage, as there is very limited improvement when increasing to 5 PMUs.

4) WLS Comparison: Last, we compare the classical weighted least-squares (WLS) solution, using various choices of (P, C). The WLS solution takes around 0.25ms per iteration, at the same computational time as the case of $(P, C) = (1, 4)$. We also include the case of $(P, C) = (5, 5)$ for demonstrating the effectiveness of prediction steps. Moreover, this case has used a pseudo-measurement scenario for generating power measurements, with each datum randomly varying from 50% to 150% of its actual value. Hence, the power measurements are very inaccurate with large outliers.

Fig. 7 plots the relative error of estimating both the ground-truth power and voltage variables. Clearly, the FOPC method outperforms the WLS solution. Due to the bad power data, the WLS method leads to an almost flat error floor in the plots, unable to track the system changes. Between the two FOPC updates, the trajectory for $(P, C) = (5, 5)$ shows slightly improved accuracy thanks to the additional prediction steps. As compared to the smart-meter data based results in Fig. 3, the error degrades slightly in Fig. 7 due to the outliers. The average computation time and relative power/voltage estimation error is listed in Table II. This test confirms the advantages of our proposed FOPC

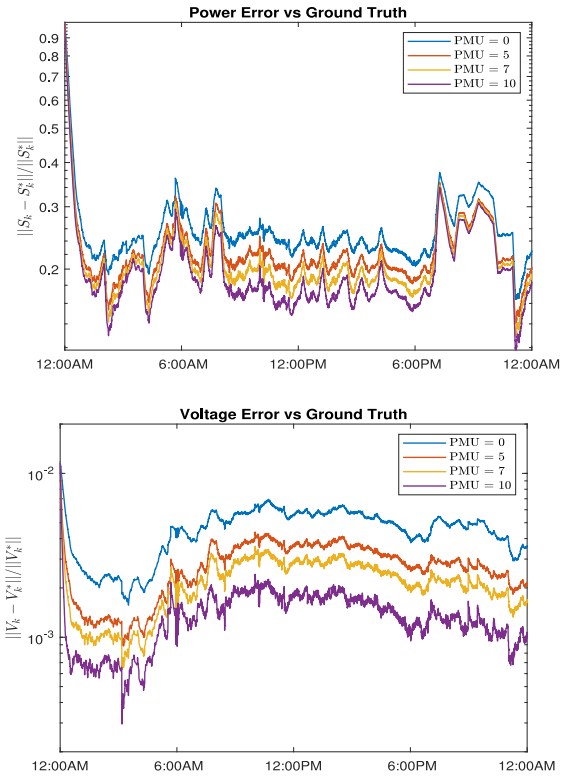

Fig. 9. Relative error of estimating the ground-truth system-wide power state (top) and voltage output (bottom) of the 123-bus case under varying number of PMUs.

updates in terms of robustness to outlier data and computation efficiency.

B. 123-Bus Test Feeder

Last, we tested the FOPC method on the 123-bus case to demonstrate its scalability. The parameter settings follow from those in the 37-bus tests, with the estimation error comparisons for fixed C given in Fig. 8 under 10 PMUs. In addition, we also compare the estimation error for varying number of PMUs for 0, 5, 7, or 10 PMUs for fixed $(P, C) = (5, 5)$, as plotted in Fig. 9. It is observed that the voltage estimation improves noticeably from 7 PMUs to 10 PMUs. Other observations are similar to those of the 37-bus tests, corroborating the improvement of including the prediction phase and high-resolution voltage data from PMUs.

V. Conclusion

This paper presented a distribution state estimation algorithm that can dynamically incorporate fast and accurate PMU voltage data. The first-order prediction-correction (FOPC) algorithm is proposed to solve the time-varying optimization problem of DDSE using a linearized power flow model. Compared to existing recursive updates, the FOPC iterations are computationally simple and require no specific modeling of system transition, suitable for the time-critical DDSE problem where the load dynamics is difficult to model. Numerical tests have shown that the data-driven prediction phase of FOPC is effective in reducing the mismatch error in tracking the power state variable. With the availability of high-resolution voltage data, the voltage estimation error is significantly improved. Future work includes exploring more diverse types of distribution system measurements and large-scale system validations using real data.

Nomenclature

$N, N^{Y/\Delta}$	Number of (wye/delta-connected) buses.
N_ϕ	Number of single-phase nodes.
$\mathbf{v}, \mathbf{z}$	Voltage phasor vector of all nodes and the real-valued counterpart.
$\mathbf{w}, \mathbf{m}$	Zero-loading voltage phasor vector of all nodes and the real-valued counterpart.
$\mathbf{i}(\mathbf{i}^\Delta)$	Current phaosr vector of phase net (phase-to-phase) injections of all nodes.
$\mathbf{s}^Y(\mathbf{s}^\Delta)$	Complex power vector of wye- (delta-) connected loads of all nodes.
$\mathbf{H}$	Matrix mapping $\mathbf{v}$ to line-to-line voltages.
$\mathbf{u}, \mathbf{u}^{Y/\Delta}$	Real-valued power vector of (wye-/delta-connected) nodes with non-zero loads.
$\mathbf{M}, \mathbf{M}^{Y/\Delta}$	Linear transformation matrix for all (wye/delta-connected) loads.
h	Fast sampling period.
t_k	Time index for discrete samples.
$\mathbf{y}_v^{(k)}$	Voltage measurement of $\mathbf{z}$ at time k.
$\mathbf{y}_u^{(k)}$	Power measurement of $\mathbf{u}$ at time k.
$\mathbf{J}_v, \mathbf{J}^{Y/\Delta}$	Measurement selection matrix.
$\mathbf{y}^{(k)}$	Full measurement vector at time k.
$\mathbf{G}^{(k)}$	Full measurement matrix at time k.
$\mathbf{n}^{(k)}$	Full measurement noise at time k.
$l^{(k)}$	Error loss function at time k.
$r^{(k)}, L_r$	Regularization function at time k that is L_r-strongly smooth.
$f^{(k)}, \nu$	Total objective function at time k that is ν-strongly convex.
L, δ	Huber loss function parameterized by δ.
$\boldsymbol{x}^*(t_k)$	Optimal solution to $f^{(k)}$ at t_k.
$\boldsymbol{x}^{(k)}$	Output solution by FOPC at time k.
$\boldsymbol{x}^{(k+1\|k)}$	Output prediction by FOPC at time k.
γ	Parameter for FOPC prediction step.
$\widehat{f}^{(k)}(\mathbf{x})$	Approximation of $f^{(k)}$ for FOPC prediction with input $\mathbf{x}$.
P, C	Number of FOPC prediction/correction steps.
α, β	Stepsize for FOPC prediction/correction steps.
$\widehat{\boldsymbol{x}}_p, \widehat{\boldsymbol{x}}_c$	Iterate for FOPC prediction/correction step p/c.
$\widehat{\mathbf{u}}^{(k)}$	FOPC output solution for DDSE at k.
$\widehat{\mathbf{u}}^{(k\|k-1)}$	FOPC prediction solution for DDSE at k.
C_0	Upper bound on the gradient $\nabla_{t\mathbf{u}} f^{(k)}(\mathbf{u})$.
ν, L	Lower/Upper bound on the Hessian norm $\|\nabla_{\mathbf{uu}} f^{(k)}(\mathbf{u})\|$.
$\varrho_{\mathrm{P}}, \varrho_{\mathrm{C}}$	Constant parameters for stepsize analysis.

References

[1] A. von Meier, D. Culler, A. McEachern, and R. Arghandeh, "Micro-synchrophasors for distribution systems," in *Proc. IEEE PES Innov. Smart Grid Technol. Conf.*, Feb. 2014, pp. 1–5.

[2] A. Gómez-Expósito, A. Abur, P. Rousseaux, A. de la Villa Jaén, and C. Gómez-Quiles, "On the use of PMUs in power system state estimation," in *Proc. 17th Power Syst. Comput. Conf.*, vol. 22, 2011, pp. 1–13.

[3] 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.

[4] M. E. Baran and A. W. Kelley, "State estimation for real-time monitoring of distribution systems," *IEEE Trans. Power Syst.*, vol. 9, no. 3, pp. 1601–1609, Aug. 1994.

[5] C. N. Lu, J. H. Teng, and W.-H. E. Liu, "Distribution system state estimation," *IEEE Trans. Power Syst.*, vol. 10, no. 1, pp. 229–240, Feb. 1995.

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

[7] W.-M. Lin and J.-H. Teng, "State estimation for distribution systems with zero-injection constraints," *IEEE Trans. Power Syst.*, vol. 11, no. 1, pp. 518–524, Feb. 1996.

[8] G. N. Korres, "A robust algorithm for power system state estimation with equality constraints," *IEEE Trans. Power Syst.*, vol. 25, no. 3, pp. 1531–1541, Aug. 2010.

[9] D. A. Haughton and G. T. Heydt, "A linear state estimation formulation for smart distribution systems," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 1187–1195, May 2013.

[10] H. Ahmadi, J. R. Martí, and A. von Meier, "A linear power flow formulation for three-phase distribution systems," *IEEE Trans. Power Syst.*, vol. 31, no. 6, pp. 5012–5021, Nov. 2016.

[11] C. Muscas, M. Pau, P. A. Pegoraro, and S. Sulis, "Uncertainty of voltage profile in PMU-based distribution system state estimation," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 5, pp. 988–998, May 2016.

[12] S. Sarri, M. Paolone, R. Cherkaoui, A. Borghetti, F. Napolitano, and C. A. Nucci, "State estimation of active distribution networks: Comparison between WLS and iterated Kalman-filter algorithm integrating PMUS," in *Proc. IEEE PES ISGT Europe*, 2012, pp. 1–8.

[13] C. Carquex, C. Rosenberg, and K. Bhattacharya, "State estimation in power distribution systems based on ensemble Kalman filtering," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 6600–6610, Nov. 2018.

[14] J. Zhao and L. Mili, "Robust unscented Kalman filter for power system dynamic state estimation with unknown noise statistics," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 1215–1224, Mar. 2019.

[15] G. Valverde and V. Terzija, "Unscented Kalman filter for power system dynamic state estimation," *IET Gener. Transm. Distrib.*, vol. 5, no. 1, pp. 29–37, Jan. 2011.
[16] A. Bernstein, C. Wang, E. Dall'Anese, J.-Y. Le Boudec, and C. Zhao, "Load-flow in multiphase distribution networks: Existence, uniqueness, non-singularity, and linear models," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 5832–5843, Nov. 2018.
[17] A. Simonetto, A. Mokhtari, A. Koppel, G. Leus, and A. Ribeiro, "A class of prediction-correction methods for time-varying convex optimization," *IEEE Trans. Signal Process.*, vol. 64, no. 17, pp. 4576–4591, Sep. 2016.
[18] A. Simonetto and E. Dall'Anese, "Prediction-correction algorithms for time-varying constrained optimization," *IEEE Trans. Signal Process.*, vol. 65, no. 20, pp. 5481–5494, Oct. 2017.
[19] W. H. Kersting, *Distribution System Modeling and Analysis*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2007.
[20] S. Bolognani and F. Dörfler, "Fast power system analysis via implicit linearization of the power flow manifold," in *Proc. Allerton Conf. Commun. Control Comput.*, 2015, pp. 402–409.
[21] S. V. Dhople, S. S. Guggilam, and Y. C. Chen, "Linear approximations to AC power flow in rectangular coordinates," in *Proc. Allerton Conf. Commun. Control Comput.*, 2015, pp. 211–217.
[22] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Trans. Power Del.*, vol. 4, no. 2, pp. 1401–1407, Apr. 1989.
[23] P. Šulc, S. Backhaus, and M. Chertkov, "Optimal distributed control of reactive power via the alternating direction method of multipliers," *IEEE Trans. Energy Convers.*, vol. 29, no. 4, pp. 968–977, Dec. 2014.
[24] A. Bernstein and E. Dall'Anese, "Linear power-flow models in multiphase distribution networks," in *Proc. IEEE PES ISGT-Europe*, 2017, pp. 1–6.
[25] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY, USA: Springer, 2009.
[26] J. Koshal, A. Nedić, and U. V. Shanbhag, "Multiuser optimization: Distributed algorithms and error analysis," *SIAM J. Optim.*, vol. 21, no. 3, pp. 1046–1081, 2011.
[27] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical Mathematics*. New York, NY, USA: Springer, 2001.
[28] K. P. Schneider *et al.*, "Analytic considerations and design basis for the IEEE distribution test feeders," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3181–3188, May 2018.
[29] A. Bernstein and E. Dall'Anese. (2017). *Real-Time Feedback-Based Optimization of Distribution Grids: A Unified Approach*. [Online]. Available: https://arxiv.org/abs/1711.01627
[30] Power Standards Lab. (2018). *MicroPMU Data Sheet*. [Online]. Available: https://www.powerstandards.com/download-center/micropmu/

Jianhan Song received the B.S. degree in information engineering from the Chinese University of Hong Kong in 2017. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Texas at Austin. From May to August 2019, he was with the Intel Lab, Hillsboro, OR, USA. His research interests include optimization, control, queuing theory, wireless networking, and reinforcement learning.

Emiliano Dall'Anese (S'08–M'11) received the Ph.D. degree from the University of Padova, Italy, in 2011. From January 2011 to November 2014, he was a Post-Doctoral Associate with the Department of Electrical and Computer Engineering, Digital Technology Center, University of Minnesota, Minneapolis, USA. From December 2014 to August 2018, he was a Senior Researcher with the National Renewable Energy Laboratory, Golden, USA. He is currently an Assistant Professor with the Department of Electrical, Computer, and Energy Engineering, University of Colorado Boulder, Boulder, USA. His research interests lie in the areas of optimization, control and signal processing, and current application domains include power systems and transportation systems.

Andrea Simonetto (M'12) received the Ph.D. degree in systems and control from the Delft University of Technology, Delft, The Netherlands, in 2012. He is currently a Research Staff Member with the Group of Optimization and Control, IBM Research Ireland, Dublin, Ireland. His research interests include large-scale centralized and distributed optimization with applications in smart energy and intelligent transportation systems.

Hao Zhu (M'12–SM'19) received the B.E. degree from Tsinghua University in 2006, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Minnesota in 2009 and 2012, respectively. She was a Post-Doctoral Research Associate and an Assistant Professor with the University of Illinois at Urbana–Champaign. She has been an Assistant Professor of electrical and computer engineering with the University of Texas at Austin since 2017. Her current research interests include power grid monitoring, power system operations and control, and energy data analytics. She was a recipient of the NSF CAREER Award and the Siebel Energy Institute Seed Grant Award in 2017 and the 2nd Best Paper Award at the 2016 North American Power Symposium. She is currently a member of the Big Data Task Force of the IEEE Power & Energy Society, the Signal Processing Theory and Methods Technical Committee and Big Data Special Interest Group of the IEEE Signal Processing Society, and the Smart Grid SIG of the IEEE Communications Society.