53rd IEEE Conference on Decision and Control
December 15-17, 2014. Los Angeles, California, USA

State estimation in power distribution networks with poorly synchronized measurements

Saverio Bolognani, Ruggero Carli and Marco Todescato

Abstract— **We consider the problem of designing a state estimation architecture for the power distribution grid, capable of filtering raw measurements from low-end phasor measurement units (PMUs) and presenting the state estimate to the control and monitoring applications that need it. Specifically, two algorithms to solve the estimation problem, which perfectly fit with the designed architecture, are proposed. Then, the contribution of this paper is threefold: i) the proposed approach is leaderless, scalable, and consists in the distributed solution of a least square problem which takes into explicit consideration the inexact time synchronization of inexpensive PMUs. ii) The first algorithm is a provably convergent scalable specialization of the alternating directions method of multipliers (ADMM). iii) The second algorithm is an extremely lightweight Jacobi-like algorithm which can be powerfully exploited in setup where the computational burden represent a major issue. Both algorithms can be implemented locally, even in a peer-to-peer fashion. The proposed approach is validated via simulations on the IEEE 123 test feeder, considering the IEEE standard C37.118-2005 for PMUs.**

I. Introduction

The electric grid is currently undergoing a deep renovation process towards the so-called *smart grid*. One of the major aspects of this modernization is the widespread deployment of measurement, monitoring, and actuation devices.

In this paper, we consider one specific thrust, which is the deployment of phasor measurement units (PMUs) in the medium and low voltage power distribution grid. The presence of PMUs is quite uncommon in today's power distribution networks. However, this scenario has become the subject of recent research efforts in the power systems community [1], [2], including a 3-year research project involving University of California together with the Power Standards Lab and Lawrence Berkeley National Lab [3].

The cost of PMUs depends on their sensing accuracy. In particular, time synchronization between them is a major technological issue, generally tackled via expensive GPS modules. So noisy and low-end device will necessarily be preferred in order to keep the cost of large scale deployment acceptable.

In this paper we propose a leaderless and distributed state estimation architecture, in which PMUs communicate with their neighbor peers in order to improve the quality of their estimates. In particular, we show how a proper distributed state estimation scheme can possibly enable the use of PMUs that are not even provided with a GPS module, and that are synchronized over general purpose communication networks.

The paper is organized as follows. In the remainder of this section, we review the main related technical literature. In Section II we introduce a model for the power grid and for the measurements obtained from the PMUs. Based on this model, we define the problem to solve as least squares problem in Section III. In Section IV we illustrate a suitable problem decomposition and we propose two different distributed and scalable solutions. The first based on the well known Alternate Direction Multiplier Method (ADMM) while the second based on the Jacobi iterative algorithm to solve liner systems. In Section V we validate our solutions via simulations. Finally, Section VI concludes the paper.

A. Related work

Several solutions alternative to the centralized computation of the state of a electric grid have been proposed in the literature. The most common approach is based on a two level architecture requiring a global coordinator [4], [5]. In this multi-area approach the grid is split into macro-areas, each of them equipped with a data processor. After a first local estimation step the global coordinator combines all these estimates making them compatible to the boundary bus measurements.

Leaderless solutions include approximate algorithms developed from the optimality conditions involved [6], [7]. Beyond requiring local observability, these algorithm are not always guaranteed to converge. Also the distributed algorithms proposed in [8] adopts the multi-area approach. The proposed algorithm is shown to be provably convergent in finite time to the optimal solution of a classical weighted least squares problem. However each area is envisioned to maintain a copy of the entire high-dimensional state vector.

To address the conventional least-squares power system state estimation problem, the authors in [9] propose a novel algorithm which is a local version of the classical ADMM. Even though the ADMM-based algorithm introduced in [9] exhibits good performance in simulations, a complete proof of its convergence is not provided. We stress the fact that one of the two approaches that we propose in this paper is an extension of the local ADMM algorithm of [9] to the minimization of quadratic functions. These quadratic functions are obtained by casting the state estimation problem as a classical least-squares problem and by adopting a specific linearized model which also allows to deal with the synchronization errors in the measurements provided by the PMUs. The ADMM algorithm we propose in this paper

S. Bolognani is with the Laboratory on Information and Decision Systems, Massachusetts Institute of Technology, Cambridge (MA), USA email: saverio@mit.edu.

R. Carli and M. Todescato are with Department of Information Engineering, University of Padova, Padova (PD), Italy email {carlirug | todescat}@dei.unipd.it

978-1-4673-6090-6/14/$31.00 ©2014 IEEE

reduces to linear iterations and it is shown to be provably convergent. Due to space limitations, we do not report here the proof of its convergence and we refer the interested reader to the document [10], where we provide all the technical details.

II. Power distribution grid model

In this section, we introduce some useful notation and we provide a description and a mathematical model for the power distribution grid and for the PMUs.

A. Power grid model

We model a power distribution grid as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, in which edges represent the power lines, and the n nodes ($n = |\mathcal{V}|$) represent buses and also the point of common coupling (PCC), i.e., the point of connection of the power distribution grid to the trasmission grid. The grid topology in represented via the incidence matrix A, defined component-wise for $\ell \in \mathcal{E}$ and $v \in \mathcal{V}$ as $[A]_{\ell v} = -1$ if ℓ starts from node v, $[A]_{\ell v} = 1$ if ℓ ends in node v, $[A]_{\ell v} = 0$ otherwise.

We consider a balanced three-phase grid where every steady state current and voltage can be represented as a complex number $y = Ye^{j\theta_y}$ of magnitude Y and phase θ_y (measured with respect to an arbitrary common time reference). The steady state of the network is described by

- $u \in \mathbb{C}^n$, where $u_v = U_v e^{j\theta_{u_v}}$ is the voltage at bus v;
- $i \in \mathbb{C}^n$, where $i_v = I_v e^{j\theta_{i_v}}$is the current injected at v.

Let $Z = \operatorname{diag}(z_\ell, \ell \in \mathcal{E})$ be the diagonal matrix of line impedances and $L = A^T Z^{-1} A$ be the complex-valued weighted Laplacian matrix. From Kirchhoff's laws we have

$$i = Lu. \tag{1}$$

Later on, we refer to the power grid state as the vector u which describe the system working point uniquely.

B. PMU model

Every bus of the power distribution grid is provided with a PMU, which measures the bus voltage u_v and the current i_v. Conversely to other measurements setups presented in literature, in the setup presented here only nodal measurements are available, while power lines are unmonitored.

We assume that the different PMUs sense the grid at synchronous, evenly distributed, times $t \in \{kT, k \in \mathbb{N}\}$, where $T > 0$ is a predetermined sample time, as dictated also by the IEEE standard C37.118-2005 [11].

At every measurement time t, the PMUs at every bus v obtain the following noisy data

$$\begin{aligned}
U_v^{(m)} &= U_v + e_{U_v}; & e_{U_v} &\sim \mathcal{N}(0, \sigma_U^2)\\
\theta_{u_v}^{(m)} &= \theta_{u_v} + e_{\theta_{u_v}} + e_{\text{sync},v}; & e_{\theta_{u_v}} &\sim \mathcal{N}(0, \sigma_\theta^2)\\
I_v^{(m)} &= I_v + e_{I_v}; & e_{I_v} &\sim \mathcal{N}(0, \sigma_I^2)\\
\theta_{i_v}^{(m)} &= \theta_{i_v} + e_{\theta_{i_v}} + e_{\text{sync},v}; & e_{\theta_{i_v}} &\sim \mathcal{N}(0, \sigma_\theta^2)\\
& & e_{\text{sync},v} &\sim \mathcal{N}(0, \sigma_{\text{sync}}^2)
\end{aligned} \tag{2}$$

where we adopt a Gaussian distribution for all these error terms. The different measurement error terms are caused by independent physical causes and can therefore be assumed uncorrelated:

- The terms e_{U_v} and e_{I_v} are caused by quantization of the acquisition devices and by harmonic distortion.
- The phase error terms $e_{\theta_{u_v}}$ and $e_{\theta_{i_v}}$ depend on the sampling time of the acquisition devices and on the algorithm employed for the estimation of the voltage-current phase difference, traditionally denoted by ϕ.
- The synchronization error $e_{\text{sync},v}$ represents the phase error due to inexact time synchronization between different PMUs, and is the same for both the voltage and the current measurements at the same PMU.

We assume that PMUs are homogeneous (even if the proposed formulation could easily account for heterogeneity of the measurement devices) and thus their errors exhibit the same probability distribution. A discussion about the variance of these error terms is postponed to Section V, where typical values given by industrial standards, are reported.

Let us stack all the measurement errors in the vectors $e_U, e_{\theta_u}, e_I, e_{\theta_i}, e_{\text{sync}}$, and let

$$e = \begin{bmatrix} e_U \\ e_{\theta_u} + e_{\text{sync}} \\ e_I \\ e_{\theta_i} + e_{\text{sync}} \end{bmatrix}$$

be the cumulative vector of all noise terms. Then the correlation matrix $R_e = \mathbb{E}[ee^T]$ for the noise is

$$R_e = \begin{bmatrix} \sigma_U^2 \mathbf{I}_n & & & \\ & (\sigma_\theta^2 + \sigma_{\text{sync}}^2)\mathbf{I}_n & & \sigma_{\text{sync}}^2 \mathbf{I}_n \\ & & \sigma_I^2 \mathbf{I}_n & \\ & \sigma_{\text{sync}}^2 \mathbf{I}_n & & (\sigma_\theta^2 + \sigma_{\text{sync}}^2)\mathbf{I}_n \end{bmatrix}.$$

where $\mathbf{I}_n$ denotes the n-dimensional identity matrix.

It is worth remarking that the variance σ_{sync}^2 of the synchronization error is typically larger than the variance of the angle measurement error σ_θ^2. Therefore, the correlation matrix R_e is not diagonal (i.e., the different cumulative error terms are not independent). Proper modeling of this fact (as we did here) is the crucial point that allows to tackle the problem of poorly synchronized PMUs in an effective way.

III. Weighted least square state estimation

We pose the power state estimation problem as a *weighted least square problem* [4]. We therefore consider the optimization problem

$$\left(\hat{U}, \hat{\theta}_u\right) = \arg\min_{U,\theta_u} J(U, \theta_u) = \arg\min_{U,\theta_u} z^T R_e^{-1} z \tag{3}$$

with z defined as

$$z = \begin{bmatrix} U^{(m)} - U \\ \theta_u^{(m)} - \theta_u \\ I^{(m)} - I(U, \theta_u) \\ \theta_i^{(m)} - \theta_i(U, \theta_u) \end{bmatrix}. \tag{4}$$

The functions $I(U, \theta_u)$ and $\theta_i(U, \theta_u)$ in (4) derive from the grid model presented in Section II-A and are nonlinear functions of the decision variables U and θ_u. Therefore the optimization problem (3) is in general non-convex, and can be difficult to tackle via standard iterative algorithms.

As proposed in other state estimation approaches based on PMUs, we introduce a rectangular representation of the state, so that we will be able to introduce a convenient linearized measurement model and, ultimately, a convenient closed form solution to (3). Let us define the complex valued measurements $u_v^{(m)} = U_v^{(m)} \exp\left(j\theta_{u_v}^{(m)}\right)$ and $i_v^{(m)} = I_v^{(m)} \exp\left(j\theta_{i_v}^{(m)}\right)$. The measurement model can be rewritten as a linear measurement model in the form

$$m = H\,x \,+\, \eta \tag{5}$$

in which[1]$m = [\Re(u^{(m)}) \;\; \Im(u^{(m)}) \;\; \Re(i^{(m)}) \;\; \Im(i^{(m)})]^T$, $x = [\Re(u) \;\; \Im(u)]^T$ and $\eta = [\Re(e_u) \;\; \Im(e_u) \;\; \Re(e_i) \;\; \Im(e_i)]^T$. The vectors e_u and e_i are defined element-wise as

$$e_{u_v} = u_v^{(m)} - u_v, \qquad e_{i_v} = i_v^{(m)} - i_v,$$

and

$$H = \begin{bmatrix} \mathbf{I}_n & 0 \\ 0 & \mathbf{I}_n \\ \Re(L) & -\Im(L) \\ \Im(L) & \Re(L) \end{bmatrix}. \tag{6}$$

Defining $R_\eta(x) = E[\eta\eta^T]$, problem (3) is equivalent to

$$\min_x \;\; (m - Hx)^T [R_\eta(x)]^{-1} (m - Hx) \tag{7}$$

The nonlinearity of the original model is now entirely contained in the error covariance matrix $R_\eta(x)$, which depends on x and makes the problem (7) hard to solve in general. This matrix is not diagonal, because both magnitude and phase errors will be reprojected into both real and imaginary components in the error vector η of the linear model (5).

A suitable approximation for R_η (that maintains the correct modeling of the time synchronization errors between PMUs) can be easily obtain but due to lack of space we do not report it here.Eventually, the matrix R_η takes the form

$$R_\eta \approx \begin{bmatrix} \Sigma_{\Re(u)} & \Sigma_{\Re(u)\Im(u)} & \Sigma_{\Re(u)\Re(i)} & \Sigma_{\Re(u)\Im(i)} \\ \Sigma_{\Im(u)\Re(u)} & \Sigma_{\Im(u)} & \Sigma_{\Im(u)\Re(i)} & \Sigma_{\Im(u)\Im(i)} \\ \Sigma_{\Re(i)\Re(u)} & \Sigma_{\Re(i)\Im(u)} & \Sigma_{\Re(i)} & \Sigma_{\Re(i)\Im(i)} \\ \Sigma_{\Im(i)\Re(u)} & \Sigma_{\Im(i)\Im(u)} & \Sigma_{\Im(i)\Re(i)} & \Sigma_{\Im(i)} \end{bmatrix} \tag{8}$$

where the sub-block matrices are all diagonal whose v-th diagonal elements depend on the measurements. We refer the interested reader to [12] where all the explicit expressions of these matrices can be found.

Notice that the value of these elements in the proposed approximation depends on the value of the measurements, and does not depend on the state x. Therefore, an approximation of the optimal solution x^* of problem (7) can be written in closed form as

$$x^* = \left(H^T R_\eta^{-1} H\right)^{-1} H^T R_\eta^{-1}\, m. \tag{9}$$

IV. Problem decomposition and Distributed optimization algorithms

Before describing the two algorithm proposed, we recall here the multi-area decomposition approach presented in [7] and also advocated in the IEEE standard C37.118-2005 [11]. Moreover, in order to reduce the complexity of the notation to a minimum, we review in general form, a common setup in distributed optimization.

To compute the solution x^* directly as in (9), one needs to know the entire measurements vector and the matrices H and R_η. To reduce the computational complexity it is possible to introduce a convenient multi-area decomposition. We assume that the grid is partitioned into non-overlapping sub-areas each of them provided with a local data processor with computational and communication capabilities. In particular, it is assumed each processor having only knowledge of the local grid topology and of the measurements which pertain to the monitored area.

According to this decomposition, assume the set of buses $\mathcal{V}$ be partitioned into s sub-areas $\mathcal{A}_1, \ldots, \mathcal{A}_s$, where $|\mathcal{A}_i| = n_i$ with $\sum_{i=1}^s n_i = n$. Moreover $\forall i \in \{1, \ldots, s\}$ let $x_{\mathcal{A}_i}$ be the portion of the state, i.e. the relevant information, belonging to area i and denote with $\mathcal{N}_i$ the set of areas which are linked to area i, that is its neighbors set. Then we recall that Problem (7) is equivalent to the problem

$$\min_{x_{\mathcal{A}_1}, \ldots, x_{\mathcal{A}_s}} \sum_{i=1}^{s} J_i\left(x_{\mathcal{A}_i}, \{x_{\mathcal{A}_j}\}_{j \in \mathcal{N}_i}\right) \tag{10}$$

Because of the specific least square estimation problem at hand, the functions J_i has a specific quadratic form, and therefore we further restrict to the case where

$$J_i(x_i, \{x_j\}_{j \in \mathcal{N}_i}) = \tag{11}$$
$$\left(q_i - A_{ii}x_i - \sum_{j \in \mathcal{N}_i} A_{ij}x_j\right)^T Q_i \left(q_i - A_{ii}x_i - \sum_{j \in \mathcal{N}_i} A_{ij}x_j\right)$$

To solve (10), and so to ultimately solve our state estimation problem, we now propose two iterative algorithms.

A. ADMM-based algorithm

The method we propose in this subsection is a partition-based version of the classical ADMM method which exploits the equivalence between problem in (10) and the following problem

$$\begin{aligned} \min_x \quad & \sum_{i=1}^{s} J_i(x_i^{(i)}, \{x_j^{(i)}\}_{j \in \mathcal{N}_i}) \\ \text{subject to} \quad & x_i^{(i)} = z_i^{(i,j)};\; x_j^{(i)} = z_j^{(i,j)} \\ & x_i^{(i)} = z_i^{(j,i)};\; x_j^{(i)} = z_j^{(j,i)}, \qquad \forall\, j \in \mathcal{N}_i, \end{aligned} \tag{12}$$

where with the notation $x_j^{(i)}$ we denote a copy of state x_j stored in memory by node i, and where the z's are auxiliary variables that are introduced by the ADMM algorithm. Observe that the connectedness of the graph $\mathcal{G}$ and the presence of the bridge variables $z's$ ensures that the optimal solution of (12) is given by $x_i^{(i)} = x_i^*$ and $x_j^{(i)} = x_j^*$.

[1] The notations $\Re(\cdot)$ and $\Im(\cdot)$ represent the real and imaginary part of a vector, respectively.

The redundant constraints added in problem (12) with respect to problem (10), allow to find the optimal solution through a distributed, iterative, partition-based implementation which optimizes the standard augmented Lagrangian defined, for $\rho > 0$, as

$$\mathcal{L} = \sum_{i=1}^{s} \Bigg\{ J_i(x_i^{(i)}, \{x_j^{(i)}\}_{j\in\mathcal{N}_i}) + \sum_{j\in\mathcal{N}_i} \Big[\lambda_i^{(i,j)} \left(x_i^{(i)} - z_i^{(i,j)}\right) + \lambda_j^{(i,j)} \left(x_j^{(i)} - z_j^{(i,j)}\right)\Big] + \sum_{j\in\mathcal{N}_i} \Big[\mu_i^{(i,j)} \left(x_i^{(i)} - z_i^{(j,i)}\right) + \mu_j^{(i,j)} \left(x_j^{(i)} - z_j^{(j,i)}\right)\Big] + \frac{\rho}{2} \sum_{j\in\mathcal{N}_i} \Big[\|x_i^{(i)} - z_i^{(i,j)}\|^2 + \|x_j^{(i)} - z_j^{(i,j)}\|^2 + \|x_i^{(i)} - z_i^{(j,i)}\|^2 + \|x_j^{(i)} - z_j^{(j,i)}\|^2\Big] \Bigg\}$$

At each iteration of the algorithm, node i, $i \in \{1, \dots, s\}$, alternates dual ascent step on the Lagrange multipliers $\lambda_i^{(i,j)}$, $\{\lambda_j^{(i,j)}\}_{j\in\mathcal{N}_i}$ and $\mu_i^{(i,j)}$, $\{\mu_j^{(i,j)}\}_{j\in\mathcal{N}_i}$, with minimization steps on the variables $x_i^{(i)}$, $\{x_j^{(i)}\}_{j\in\mathcal{N}_i}$ and $z_i^{(i,j)}$, $\{z_j^{(i,j)}\}_{j\in\mathcal{N}_i}$.

However, for the case where the functions $J_i's$ have the particular quadratic structure illustrated in (11), these optimization steps can be greatly simplified. Indeed in this case the partition-based ADMM algorithm reduces to a linear algorithm requiring, during each iteration of its implementation, only one communication round involving the $x_i^{(i)}$, $\{x_j^{(i)}\}_{j\in\mathcal{N}_i}$, $i \in \{1, \dots, s\}$, variables. To show that, let us it is convenient to introduce the following compact notation. Consider node i and, without loss of generality, assume $\mathcal{N}_i = \{j_1, \dots, j_{|\mathcal{N}_i|}\}$. Then let

$$X^{(i)} = \begin{bmatrix} x_i^{(i)} \\ \left\{x_j^{(i)}\right\}_{j\in\mathcal{N}_i} \end{bmatrix},$$

$$A_i = \begin{bmatrix} A_{ii} \; A_{ij_1} \; \dots \; A_{ij_{|\mathcal{N}_i|}} \end{bmatrix},$$

$$M_i = \text{diag}\left\{ |\mathcal{N}_i| \, I_{m_i}, I_{m_{j_1}}, \dots, I_{m_{j_{|\mathcal{N}_i|}}} \right\}.$$

Additionally we introduce the following auxiliary variables,

$$G^{(i)} = \begin{bmatrix} G_i^{(i)} \\ G_{j_1}^{(i)} \\ \vdots \\ G_{j_{|\mathcal{N}_i|}}^{(i)} \end{bmatrix}, \; F^{(i)} = \begin{bmatrix} F_i^{(i)} \\ F_{j_1}^{(i)} \\ \vdots \\ F_{j_{|\mathcal{N}_i|}}^{(i)} \end{bmatrix}, \; B^{(i)} = \begin{bmatrix} B_i^{(i)} \\ B_{j_1}^{(i)} \\ \vdots \\ B_{j_{|\mathcal{N}_i|}}^{(i)} \end{bmatrix}$$

where $G_i^{(i)}, F_i^{(i)}, B_i^{(i)} \in \mathbb{R}^{m_i}$ and $G_{j_h}^{(i)}, F_{j_h}^{(i)}, B_{j_h}^{(i)} \in \mathbb{R}^{m_{j_h}}$. It turns out that $A_i \in \mathbb{R}^{r_i \times \gamma_i}$, $M_i \in \mathbb{R}^{\gamma_i \times \gamma_i}$ and $G^{(i)}, F^{(i)}, B^{(i)} \in \mathbb{R}^{\gamma_i}$, where $\gamma_i = m_i + \sum_{h=1}^{|\mathcal{N}_i|} m_{j_h}$.

The *partition-based ADMM algorithm* for quadratic functions is formally described as follows. The standing assumption is that all the matrices $A_i^T Q_i \, A_i + M_i$, $i \in \{1, \dots, n\}$ are invertible.

Processor states: For $i \in \{1, \dots, s\}$, node i stores a copy of the variables $X^{(i)}, G^{(i)}, F^{(i)}, B^{(i)}$.

Initialization: Every node initializes the variables it stores in memory to 0.

Transmission iteration: For $t \in \mathbb{N}$, at the start of the t-th iteration of the algorithm, node i transmits to node j, $j \in \mathcal{N}_i$, its estimates $x_i^{(i)}(t)$, $x_j^{(i)}(t)$. It also gathers the t-th estimates of its neighbors, $x_j^{(j)}(t)$, $x_i^{(j)}(t)$, $j \in \mathcal{N}_i$.

Update iteration: For $t \in \mathbb{N}$, node i, $i \in \{1, \dots, s\}$, based on the information received from its neighbors, perform the following computations in order:

$$G_i^{(i)}(t) = \frac{\rho}{2} \sum_{j\in\mathcal{N}_i} \left(x_i^{(i)}(t) - x_i^{(j)}(t)\right)$$

$$G_{j_h}^{(i)}(t) = \frac{\rho}{2} \left(x_{j_h}^{(i)} - x_{j_h}^{(j_h)}\right), \quad 1 \leq h \leq |\mathcal{N}_i|$$

$$F^{(i)}(t+1) = F^{(i)}(t) + G^{(i)}(t)$$

$$B^{(i)}(t+1) = 2\rho M_i X^{(i)}(t) - G^{(i)}(t+1) - 2F^{(i)}(t+1)$$

Finally, it updates $X^{(i)}$ as follows

$$X^{(i)}(t+1) = \left[A_i^T Q_i \, A_i + M_i\right]^{-1} \left[A_i^T Q_i z_i + \frac{1}{2} B^{(i)}(t+1)\right]$$

Proposition IV.1 characterizes the performance of the algorithm.

Proposition IV.1. *Consider the partition-based ADMM algorithm described above. Let ρ be any real number. Assume that the matrices $A_i^T Q_i \, A_i + M_i$, $i \in \{1, \dots, s\}$, are invertible. Then the trajectory $t \to \left\{X^{(i)}(t)\right\}$ converge exponentially to the optimal solution, namely, for $i \in \{1, \dots, n\}$, $x_j^{(i)}(t) \to x_j^*$ for all $j \in \mathcal{N}_i$ and, in particular,*

$$x_i^{(i)}(t) \to x_i^*.$$

Proof. The proof is omitted and is available in [10]. □

Remark. *We point out that it is be possible to provide, as done in [9], a simplified a version of the partition-based ADMM algorithm, where two monitors communicate with each other only the information related to those nodes which are on the boundary of their sub-areas of interest. Then the vectors $X^{(i)}$, $G^{(i)}, F^{(i)}, B^{(i)}$ and the matrices M_i, A_i result to be smaller in size.*

For the sake of the notation simplicity we have preferred not to provide the details of this implementation aspect.

B. Jacobi-like algorithm

The method we propose in this subsection, hereafter denoted as *Jacobi-like algorithm*, is inspired by the Jabobi technique used to iteratively solve systems of linear equations [13]. The algorithm is formally described as follows. The standing assumption is that the matrices $A_{ii}^T Q_i A_{ii}$, $i \in \{1, \dots, s\}$ are all invertible.

Processor states: For $i \in \{1, \dots, s\}$, node i stores an estimate $x_i(0) \in \mathbb{R}^{m_i}$ of its own state.

Initialization: Every node initializes its estimate to an arbitrary value.

Transmission iteration: For $t \in \mathbb{N}$, at the start of the t-th iteration of the algorithm, node i transmits its estimate $x_i(t)$ to all its neighbors. It also gathers the t-th estimates of its neighbors, $x_j(t)$, $j \in \mathcal{N}_i$.

Update iteration: For $t \in \mathbb{N}$, node i, $i \in \{1, \dots, s\}$ updates its estimate as follows

$$x_i(t+1) = \underset{x_i}{\operatorname{argmin}} \; J_i\left(x_i; \; \{x_j(t)\}_{j \in \mathcal{N}_i}\right) = \left(A_{ii}^T Q_i A_{ii}\right)^{-1} A_{ii}^T Q_i \left(z_i - \sum_{j \in \mathcal{N}_i} A_{ij} x_j(t)\right)$$

To establish the convergence properties of the Jacobi-like algorithm it is convenient to introduce the following block matrix $K = [K_{ij}]$, $i, j = 1, \dots, s$, where $K_{ij} \in \mathbb{R}^{m_i \times m_j}$ is defined as

$$K_{ij} = \begin{cases} A_{ii}^T Q_i A_{ii} & \text{if } j = i \\ A_{ii}^T Q_i A_{ij} & \text{if } j \in \mathcal{N}_i, \; j \neq i \\ 0 & \text{if } j \notin \mathcal{N}_i \end{cases}$$

In the following Proposition, by diag$\{K\}$ we denote the block diagonal matrix having in the diagonal the blocks $K_{11}, \dots, K_{ss}$. We have the following result.

Proposition IV.2. *Assume the spectral radius of the matrix* $(diag\{K\})^{-1}(K - diag\{K\})$ *is strictly less than one, i.e., all its eigenvalues are strictly inside the unitary circle. Then, there exists* $\bar{x} \in \mathbb{R}^N$, *such that the trajectory* $t \to x(t)$ *generated by the Jacobi-like algorithm converges exponentially to* $\bar{x}$. *Let be* $\bar{z} = \left[(A_{11}^T Q_1 z_1)^T, \dots, (A_{ss}^T Q_s z_s)^T\right]^T$. *If the matrix* K *is invertible then*

$$\bar{x} = K^{-1} \bar{z}$$

Proof. Observe that the updating step can be written as

$$x(t+1) = (\text{diag}\{K\})^{-1} (\bar{z} - (K - \text{diag}\{K\}) x(t)) \quad (13)$$

where (13) represents the standard iteration of the Jacobi method. The result established in the Proposition follows from the classical results on the Jacobi method [13]. □

Remark. *In general the state* $\bar{x}$ *is different from the optimal state* x^**.However, it can be seen via numerical simulation, as shown in Section V that, when applied to Problem* (10), *the Jacobi-like algorithm converges to an estimate* $\bar{x}$ *which is very close to* x^*.

Remark. *It is worth stressing that the computation of the update iteration of the Jacobi-like algorithm is simpler, and, in turn, less time-consuming, than the update iteration of the partition-based ADMM algorithm.*

V. Simulation results

In this section we validate and compare the two proposed algorithms via simulations on standard medium voltage IEEE testbeds [14]. Specifically, we provide a comparison between the performance of the *ADMM-based algorithm* presented in Section IV-A and the *Jacobi-like algorithm* presented in IV-B on the IEEE 123 test feeder [14].

To show the effectiveness of the proposed algorithms, for both of them we consider only a peer-to-peer architecture, in which all PMUs independently process their available data.

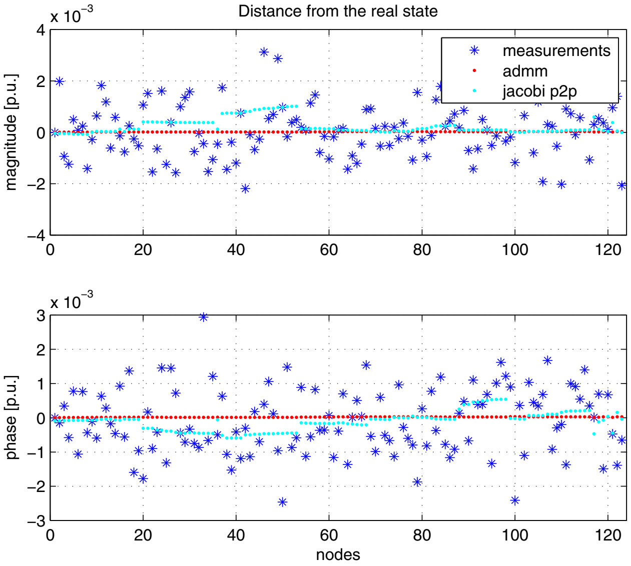

Fig. 1. Residual error at the steady state of the Admm algorithm (red) and of the Jacobi-like algorithm (blue), in the case of a peer-to-peer architecture, compared to the raw measurements.

We considered the following standard deviations for the measurement errors:

$$\begin{aligned} \text{voltage amplitude: } \sigma_U &= 10^{-3} U_N \text{ [Volt]} \\ \text{current amplitude: } \sigma_I &= 10^{-3} I_{\max} \text{ [A]} \\ \text{angle: } \sigma_\theta &= 10^{-3} \text{ [rad]} \\ \text{sync: } \sigma_{\text{sync}} &= 3 \cdot 10^{-3} \text{ [rad]} \end{aligned}$$

where U_N is the nominal voltage of the grid and where $I_{\max}$ is the largest current magnitude in the network.These parameters are in accordance with the maximum measurement errors allowed by the IEEE standard C37.118-2005 [11], which specifies only an aggregate constraint on the measurement errors, without differentiating between magnitude and phase constraints:

$$\frac{|u_v^{(m)} - u_v|}{|u_v|} \leq 1\%, \qquad \frac{|i_v^{(m)} - i_v|}{|i_v|} \leq 1\%.$$

In the upper panel of Figure 1, we plotted, for each node v of the distribution test feeder, a comparison between the magnitude errors $\hat{U}_v - U_v$ and $U_v^{(m)} - U_v$, while, in the lower panel, we plotted a comparison between the phase errors $\hat{\theta}_{u_v} - \theta_{u_v}$ and $\theta_{u_v}^{(m)} - \theta_{u_v}$, where by $\hat{U}_v$ and $\hat{\theta}_{u_v}$ we denote the steady state estimates computed by estimation algorithms. We want to stress the fact that the state estimates obtained via the ADMM-based algorithm (red dots) do not depend on the adopted decomposition of the grid, as the steady state is provably the unique optimal estimate. Notice, on the other hand that the Jacobi-like algorithm does not converge to the true optimal estimate, as a small steady state error is noticeable (blue dots). It is however stable, as predicted by Proposition IV.2, whose assumptions are all verified in this scenario. In both cases, the measurement errors are drastically reduced by the algorithms proposed, which provide a consistent and accurate estimate of the node voltages.

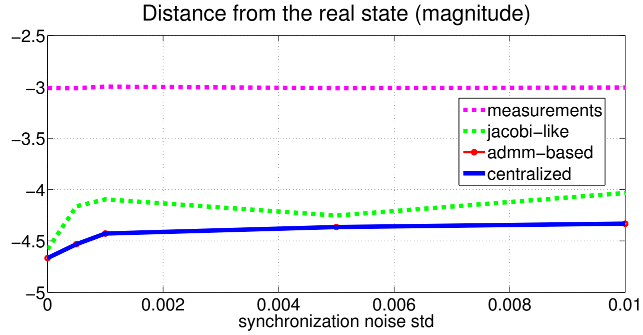

Fig. 2. Comparison of the norm of the magnitude estimation error for different values of the standard deviation σ_{sync}. Both algorithms have been reported, together with the raw measurements and the optimal centralized estimate (which corresponds to the estimate generated by the ADMM algorithm). The plot represents the average of 1000 trials, and the norm of the error has been plotted in logarithmic scale.

We also consider the case in which the synchronization error is larger. Specifically, up to sync: $\sigma_{\text{sync}} = 10^{-2}$ [rad], which is comparable to the synchronization accuracy achievable without GPS, if the PMUs are synchronized via data network synchronization protocols (see for example [15] and references therein). In Figures 2 we plotted the norm of the magnitude as a function of σ_{synch}. It can be seen that the ADMM algorithm outperforms the Jacobi-like algorithm, even if they both reduce the measurement errors present in the raw data. As expected, the curves generated by the ADMM algorithm correspond to the optimal centralized estimate. Similar results can be observed for the phase errors.

In order to analyze the computational complexity of the two proposed algorithm, we studied the evolution of the error norm $\|\hat{u}(t) - u^*\|$, where $\hat{u}(t) = \hat{U}(t)e^{j\hat{\theta}_u(t)}$ denotes the estimate computed after t iterations of the algorithms, and u^* is the optimal state estimate. We plotted the error norm as a function of the computational time, in Figure 3. Even if the estimate $\hat{u}(t)$ generated by ADMM-based algorithm converges exponentially to u^*, it can be observed how the updating step of the Jacobi-like algorithm is computationally much simpler than the corresponding iteration in the ADMM-based algorithm.Iindeed, the Jacobi-like algorithm exhibit a faster transient, and may be preferred if the system sampling time is short, compared to the available computational capabilities at the agents.

VI. Conclusions

We considered the motivating scenario of a smart power distribution grid, where a number of control applications require precise measurements of the grid state. We modeled a practical case in which noisy measurements are provided by PMUs located at the grid buses. In particular, we model the measurement error that is caused by inexact synchronization of the PMUs, which is a major issue in the deployment of low-end measurement solutions. We derived two different distributed and scalable algorithms in order to compute the optimal state estimate, and we proved their convergence. Via simulations, we showed the effectiveness of both presented solutions, showing that time synchronization between PMUs in the power distribution network becomes a tractable issue and suggest that GPS-less PMUs could be used.

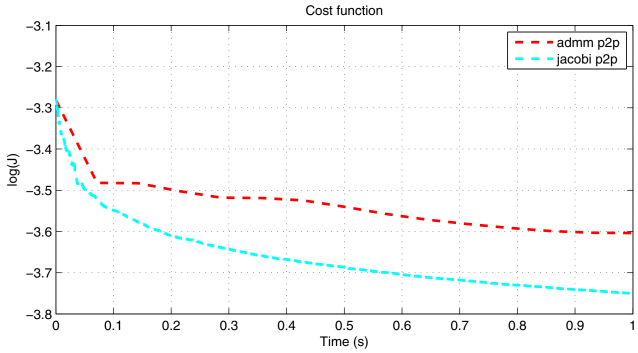

Fig. 3. Behavior of the error norm $\|\hat{u}(t)-u^*\|$ for both the ADMM and the Jacob-like estimation algorithm, in the case of a peer-to-peer architecture, as function of time. The norm of the error has been plotted in logarithmic scale.

References

[1] M. M. Albu, R. Neurohr, D. Apetrei, I. Silvas, and D. Federenciuc, “Monitoring voltage and frequency in smart distribution grids. a case study on data compression and accessibility.” in *IEEE Power and Energy Society General Meeting*, 2010.

[2] J. D. L. Ree, V. Centeno, J. S. Thorp, and A. G. Phadke, “Synchronized phasor measurement applications in power systems,” *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 20–27, Jun. 2010.

[3] A. von Meier, D. Culler, and A. McEachern, “Micro-synchrophasors for distribution systems,” in *IEEE Innovative Smart Grid Technologies (ISGT 2013)*, 2013.

[4] F. C. Schweppe, J. Wildes, and D. Rom, “Power system static state estimation: Parts I, II, and III,” *IEEE Transactions Power App. Syst.*, vol. PAS-89, pp. 120–135, Jan. 1970.

[5] A. Gómez Expósito, A. Abur, A. de la Villa Jean, and C. Goméz Quiles, “A multilevel state estimation paradigm for smart grids,” *Proc. IEEE*, vol. 99, no. 6, pp. 952–976, Jun. 2011.

[6] D. M. Falcao, F. F. Wu, and L. Murphy, “Parallel and distributed state estimation,” *IEEE Trans. Power Syst.*, vol. 10, no. 2, pp. 724–730, May 1995.

[7] A. J. Conejo, S. de la Torre, and M. Cañas, “An optimization approach to multiarea state estimation,” *IEEE Transactions on Power Systems*, vol. 22, no. 1, pp. 213–231, Feb. 2007.

[8] F. Pasqualetti, R. Carli, and F. Bullo, “Distributed estimation via iterative projections with application to power network monitoring,” *Automatica*, vol. 48, no. 5, pp. 747–758, 2012.

[9] V. Kekatos and G. Giannakis, “Distributed robust power system state estimation,” *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1617–1626, May 2013.

[10] S. Bolognani, R. Carli, and M. Todescato, “Convergence of the partition-based ADMM for a separable quadratic cost function,” Dec. 2013, available on arXiv. [Online]. Available: http://www.dei.unipd.it/∼sbologna/papers/proof-admm.pdf

[11] K. E. Martin, D. Hamai, M. Adamiak, S. Anderson, M. Begovic, G. Benmouyal, G. Brunello, J. Burger, J. Cai, B. Dickerson, V. Gharpure, B. Kennedy, D. Karlsson, A. G. Phadke, J. Salj, V. Skendzic, J. Sperr, Y. Song, C. Huntley, B. Kasztenny, and E. Price, “Exploring the IEEE standard C37.118-2005 synchrophasors for power systems,” *IEEE Transactions on Power Delivery*, vol. 23, no. 4, pp. 1805–1811, Oct. 2008.

[12] S. Bolognani, R. Carli, and M. Todescato, “State estimation in power distribution networks with poorly synchronized measurements,” Mar. 2014, available on arXiv. [Online]. Available: http://www.dei.unipd.it/∼sbologna/papers/state-estimation.pdf

[13] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[14] W. H. Kersting, “Radial distribution test feeders,” in *IEEE Power Engineering Society Winter Meeting*, vol. 2, Jan. 2001, pp. 908–912.

[15] S. Bolognani, R. Carli, E. Lovisari, and S. Zampieri, “A randomized linear algorithm for clock synchronization in multi-agent systems,” in *Proceedings of the 51st IEEE Conference on Decision and Control (CDC 2012)*, 2012.