Mitigation of grid parameter uncertainties for the steady-state operation of a model-based voltage controller in distribution systems

Muhammad Andy Putratama, Rémy Rigo-Mariani, Vincent Debusschere, Yvon Bésanger

Mitigation of Grid Parameter Uncertainties for the Steady-state Operation of a Model-based Voltage Controller in Distribution Systems

Muhammad Andy Putratama[a], Rémy Rigo-Mariani[a,*], Vincent Debusschere[a], Yvon Bésanger[a]

[a] *Univ. Grenoble Alpes, CNRS, Grenoble INP, G2Elab, GreEn-Er, 21 avenue des Martyrs, Grenoble, 38 000, France*

Abstract

This paper deals with the impact of line impedances uncertainties on model-based voltage controllers for distribution networks in the context of secondary to tertiary control levels (i.e., 30 minutes control horizon). The study proposes two methodologies: i) centralized and ii) distributed approaches, to estimate grid impedances by relying on static historical measurement data and adjust the parameters of a model-based voltage controller. Furthermore, an online impedance tuning scheme is proposed to successively fine-tune the impedance estimation over successive control periods (along several days). The simulations results highlight the preciseness of the proposed methodologies, with both centralized and distributed able to estimate the grid impedances within an acceptable accuracy (between 4 % and 7 % of error). Moreover, the proposed tuning algorithm shows to be very effective, where the estimation error can be lowered under 1 %. Finally, robustness studies are performed to test the proposed methodologies in the presence of measurement noises. Through this study, the robustness of the proposed tuning scheme can be validated, in which the algorithm is able to correct massive impedance errors after three months of tuning rounds only.

Keywords:
Impedance estimation, distribution grid, grid parameter, model-based voltage controller, optimal power flow (OPF), distributed control

1. Introduction

Nowadays, the development of worldwide electricity systems are starting to be "decentralized, decarbonized, and democratized" [1]. With this trend in mind, many renewable-based sources are now being deployed in the distribution networks (i.e., medium and low voltage level) and being closer to the customer-side. This type of generation is called as Distributed Energy Resources (DERs) with a geographic dispersion of small scales units. Unlike conventional generating units, DERs can also be deployed and owned by the consumers as "*behind the meter*" assets. This phenomenon, and complement with consumption flexibility, leads to turning traditional consumers into prosumers [2].

Globally, distributed solar photovoltaic systems (PV) account the highest share of DER deployments, and are foreseen to increase vastly in the future. However, due to its intermittent and non-dispatchable nature, massive integrations of PV systems can lead to unpredictable network flows that can affect the quality of the power supply. In particular, voltage problems have been addressed as the dominant effect in the massive integration of PV systems [3]. Unlike transmission systems, distribution systems have a high R/X characteristic. As a consequence, the grid voltage is more sensitive to changes in active power flows, which is a typical case in the grid with high penetration of PV systems. Therefore, appropriate operational management of DERs is essential to prevent possible voltage violations, and especially, to ensure the most economic benefit among the distribution system actors, such as distribution system operator (DSO) and prosumers.

The operational management of DER is typically formulated as an optimal power flow problem (OPF) that rely on grid parameters data (i.e., detailed data of branch resistance and reactance) [4]. In the context of voltage management, it can also be referred to as a model-based voltage controller. In the literature, most of researches on OPF or model-based voltage controller utilize an assumption of perfect knowledge of the grid parameters.

*Corresponding author
Email address:
remy.rigo-mariani@g2elab.grenoble-inp.fr (Rémy Rigo-Mariani)

This assumption may not be realistic, as detailed information of distribution grids is often inaccurate or even not available, especially in low-voltage grids [5]. This is mainly explained due to outdated information, which can happen because of regular grid maintenance and reconfiguration that may not be well synchronized with all stakeholders [6]. In addition, inaccurate grid data can also occur due to the natural degradation (i.e., ageing) of the lines due to exploitation.

Studies in [7, 8] have highlighted that inacurrate grid parameters can significantly affect the operational effeciency of the system, and even worse, cause a system instability. In the context of model-based voltage controllers, grid parameter uncertainties can lead to irrelevant control actions, such as over-curtailments of PV systems that may incur additional economic loss to the prosumers. Furthermore, previous works [9] showed how the grid parameters error in model-based voltage controller can even lead to an increase occurrence of voltage violations. All in all, the study carried out in this paper highlights the importance of having an accurate grid model, especially to ensure that a model-based voltage controller is able to maintain the grid voltage within the required standards. This also points out the need to develop algorithms that can precisely estimate grid parameters, so that the optimality and the accuracy of the control can be guaranteed.

Numerous methodologies are available in the literature for grid impedance estimation. For instance, studies in [5, 6] proposed a methodology to estimate grid parameters using voltage magnitude as well as branch active and reactive power measurements. While in [8], the authors were able to perform the estimation by only relying on voltage magnitude and phase information. A recent study in [10] proposed a methodology to compute lines resistance and reactance using only time-stamped voltage magnitude measurements. However, the main limitation of the proposed method is that it relies on recursive algorithms (not optimization-based) and requires more than ten thousands of measurement samples, resulting to high computational complexity. Our motivation is to address the gap in the studied literature. In particular, our objective is to develop a methodology to estimate the grid impedance using the least least amount of available information and with low computational complexity.

Most of impedance estimation methodologies in the literature employ centralized approach, meaning that the problem is solved by a dedicated centralized entity (e.g., DSO or aggregator) who has access to prosumers' assets and data, including their smart meters. This authority may lead to risk of security and privacy concerns to the prosumers. Moreover, a centralized computation also presents reliability issues, since any loss of controller/communication will jeopardize the overall control system. As a consequence, moving to distributed computation is the key answer to overcome the aforementioned challenges.

Therefore, this paper proposes methodologies to estimate grid impedance parameters (i.e., branch resistance and reactance) of a radial distribution grid using historical measurement data for voltage management purposes. We focus particularly on steady-state control within the scope of operational management of DERs – i.e., secondary to tertiary levels in the conventional hierarchical control [11] with a control horizon of 30 minutes, such as in [12, 13]. The tuning of the controller follows the different approach proposed and based on RMS measurement of the voltage magnitude. That considered control/measurements horizon is also typically considered in the management of energy markets [14, 15]. All in all, the proposed methodologies in this paper can provide precise values for grid parameters (up to 99% of accuracy) for all the aforementioned control contexts.

The main contributions of this paper are summarized as follows:

- Convex optimization-based strategies to estimate grid impedance parameters by only utilizing voltage and bus power static (i.e., steady-state) historical measurements data of 30 minutes sample-time.
- Distributed impedance estimation schemes using consensus alternating direction method of multipliers (C-ADMM). As far as our knowledge, there is no study in the literature that proposes a distributed algorithm for impedance estimation.
- An impedance tuning algorithm to adaptively refine branch resistance and reactance parameters with successive runs of a model-based voltage controller with 30 minutes time steps, simulated over a year.

The remainder of this paper is organized as follows. Section 2 firstly presents the model-based voltage controller considered in this paper. Then, the proposed methodologies to estimate grid impedance parameters are introduced in the same section. The final part of the section details the proposed strategic impedance tuning algorithm. All the methodologies are simulated and compared in Section 3, and Section 4 concludes the paper.

2. Model Formulation and Methodology

2.1. Model-Based Voltage Controllers

This paper specifically focuses on a radial distribution grid, which can be represented as a graph $\mathcal{G}(\mathcal{B}, \mathcal{E})$, where $\mathcal{B} = \{0, 1, \ldots, |\mathcal{B}| - 1\}$ and $E = \{(i, j)\} \subset \mathcal{B} \times \mathcal{B}$ denote the sets of buses and lines/branches in the grid respectively. Let $\mathcal{T} = \{0, \Delta t, \ldots, (|\mathcal{T}| - 1) \cdot \Delta t\}$ be a generic set of time horizon, with Δt denoting the time step in minutes. Table. 1 presents the nomenclature of the main fundamental symbols used throughout this paper.

Table 1: Nomenclature of the fundamental symbols

Symbol	Description
$v_{j,t}, \nu_{j,t}$	Voltage and squared voltage (i.e., v_j^2) at bus j at time t
$p^{\text{net}}_{j,t}, q^{\text{net}}_{j,t}$	Net active and reactive power at bus j at time t
$p_{ij,t}, q_{ij,t}$	Active and reactive power flowing from bus i to bus j at time t
$i_{ij,t}, \ell_{ij,t}$	Current and squared current (i.e., i_{ij}^2) flowing from bus i to bus j at time t
r_{ij}, x_{ij}	Resistance and reactance of the line between bus i and bus j at time t

In this paper, two different voltage management architectures (i.e., centralized and distributed) are considered, adapted from our previous works [16, 12]. In the centralized control architecture , there exists a dedicated centralized entity (e.g., DSO or aggregator) who is responsible for voltage management. In this case, the centralized entity has direct access to prosumers' assets and data. The centralized voltage control is formulized as a multi-objective optimal power flow (OPF) problem with second-order conic programming (SOCP) relaxation [17]. The objective of the OPF problem is to minimize PV curtailments over a predefined time horizon $\mathcal{T}$, subjected to DistFlow (i.e., grid) model, PV and voltage constraints that are not represented here for the sake of clarity, but described in [16, 12]. The objective function is illustrated in (1).

$$\min_{\boldsymbol{y}} \sum_{j \in \mathcal{B}} \sum_{t \in \mathcal{T}} \underbrace{\Delta p^{\text{net}}_{j,t}}_{\text{PV curtailments}} + \underbrace{\sum_{(i,j) \in \mathcal{E}} \sum_{t \in \mathcal{T}} \ell_{ij,t} \cdot r_{ij}}_{\text{Grid losses}} \tag{1}$$

where $\Delta p^{\text{net}}_{j,t}$ represents the change of bus j net power due to active power curtailments. The vector $\boldsymbol{y}$ aggregates the decision variables that consist of PV set-points and grid state variables over the time horizon. The centralized control architecture can be fully transformed into a distributed architecture using similar distributed algorithm that will be presented in Section 2.2.2. The complete formulation of the distributed OPF problem is also available in previous work [12].

2.2. Impedance Estimation Algorithm

In this subsection, the proposed methodologies to estimate line impedances (i.e., resistance and reactance) are presented. The general flow diagram of the methodologies is shown in Fig. 1. We utilize historical data as inputs of the algorithms. Based on that, the line impedance, i.e., resistance (r_{ij}) and reactance (x_{ij}), of each branch can be estimated. In the proposed methodologies, we assume that we have access to voltage and bus net powers ($p^{\text{net}}_{j,t}, q^{\text{net}}_{j,t}$) measurements of all the buses. This assumption is reasonable, due to the fact that a prosumer (who represents a single bus here) shall be equipped with smart meters.

It should be noted that the following additional assumptions are also considered:

1. Synchronized and accurate measurements. However, further studies on algorithm robustness are also performed in Section 3.3, where we investigate the performance in the presence of measurement error/noise.

2. Knowledge of the grid topology. However, the distributed algorithms furtherly introduced in Section 2.2.2 can relax this assumption.

Let $\mathcal{T}^{\text{hist}}$ be the set of time interval of historical measurements utilized as the input of the impedance estimation algorithm. Moreover, let any symbol with an accent $\hat{\cdot}$ represents measurement data, which also act as parameters in any optimization problem. We propose centralized and distributed methodologies for impedance estimation, as presented below.

2.2.1. Centralized Impedance Estimation

We first consider a centralized scenario, where the impedance estimation task is solved by a dedicated centralized entity (e.g., DSO or aggregator) who has access to measurement data, including prosumers' smart meter. The details of this case are summarized in Fig. 2.

By utilizing only voltage and bus net powers data, we propose a two-stage optimization problem as the algorithm for estimating the line impedance, as depicted in Fig. 3. For the centralized scenario, each stage of the algorithm is presented below.

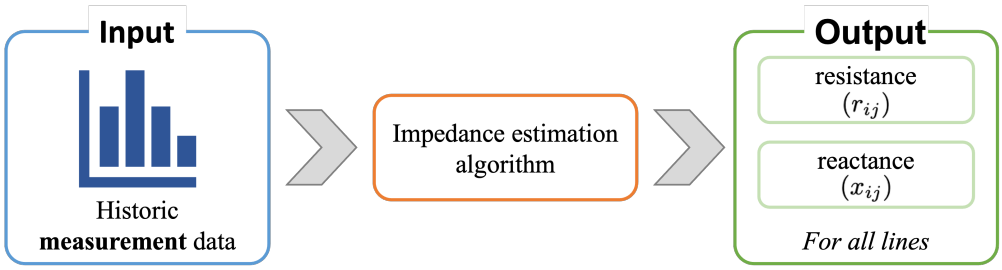
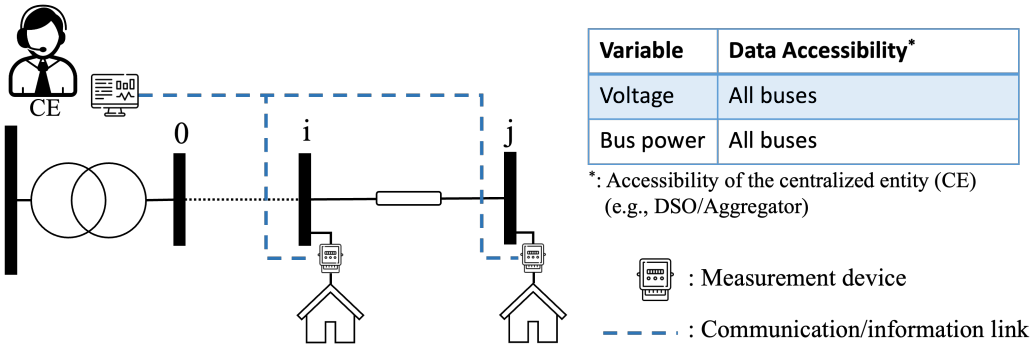


Figure 1: Impedance estimation algorithm.



Variable	Data Accessibility*
Voltage	All buses
Bus power	All buses

*: Accessibility of the centralized entity (CE) (e.g., DSO/Aggregator)

Figure 2: Illustration of the centralized scenario.

Stage 1: $p_{ij,t}$ and $q_{ij,t}$ lower bound estimation. The aim of the first stage is to find the lower bound for the active and reactive power flows of each branch at each time step. These bounds can be computed by solving a model fitting problem (2), by considering the linearized model of DistFlow (i.e., grid model) power flow equations [18] as the constraints. These linearized model neglect the impedance term in the equations, as shown in (2b) and (2c) (strikeout symbols). In this stage, the objective function aims at minimizing the sum of squared difference between the computed and the actual measurements of bus net powers -– summed over the time horizon of the measurements.

$$\min_{\boldsymbol{y}} \sum_{t\in\mathcal{T}^{\text{hist}}} \sum_{j\in\mathcal{B}} (\hat{p}^{\text{net}}_{j,t} - p^{\text{net}}_{j,t})^2 + (\hat{q}^{\text{net}}_{j,t} - q^{\text{net}}_{j,t})^2 \tag{2a}$$

subject to:

$$\forall t \in \mathcal{T}^{\text{hist}},\ \forall (i,j) \in \mathcal{E}: \quad \underline{p}_{ij,t} = \sum_{k:(j,k)\in\mathcal{E}} \underline{p}_{jk,t} + p^{\text{net}}_{j,t} + \cancel{r^2_{ij}\ell_{ij,t}} \tag{2b}$$

$$\underline{q}_{ij,t} = \sum_{k:(j,k)\in\mathcal{E}} \underline{q}_{jk,t} + q^{\text{net}}_{j,t} + \cancel{x^2_{ij}\ell_{ij,t}} \tag{2c}$$

where the vector $\boldsymbol{y} = (p^{\text{net}}_{j,t}, q^{\text{net}}_{j,t}, \underline{p}_{ij,t}, \underline{q}_{ij,t} |\ \forall (i,j) \in \mathcal{E},\ \forall t \in \mathcal{T}^{\text{hist}})$ aggregates the considered decision variables. Naturally, line impedances have a characteristic of $r_{ij}, x_{ij} \geq 0$ (i.e., resistive and inductive). Therefore, the presence of impedance term in the original branch flow equations will increase (additive effect) the magnitude of the power flows at the upstream branch $(p_{ij,t}, q_{ij,t})$. This, in fact, describes the natural behavior of the impedance that incur losses to the system. By neglecting r_{ij}, x_{ij} (i.e., neglecting the losses), solving (2) shall ultimately give the lower bounds of $p_{ij,t}, q_{ij,t}$.

Stage 2: Impedance estimation. In this stage, we propose a model fitting problem (3) in order to estimate the r_{ij}, x_{ij} of each line. We utilize both voltage measurements and the lower bounds of the branch powers obtained from the previous stage. The proposed formulation aims at minimizing the sum of squared difference between the computed and the actual measurements of bus voltages -– summed over the time horizon of the measurements.

$$\min_{\boldsymbol{v},\boldsymbol{z}} \sum_{j\in\mathcal{B}} \sum_{t\in\mathcal{T}^{\text{hist}}} (\hat{v}_{j,t} - v_{j,t})^2 \tag{3a}$$

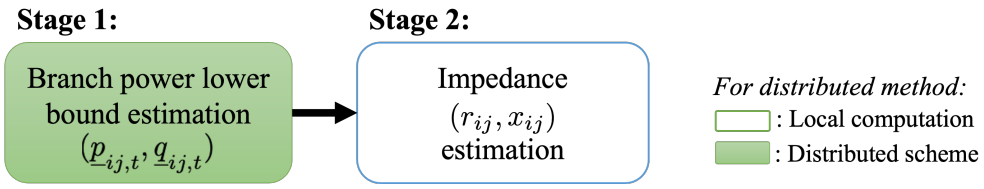


Figure 3: A two-stage impedance estimation algorithm for both centralized and distributed case.

subject to:

$$\forall t \in \mathcal{T}^{\text{hist}},\ \forall (i,j) \in \mathcal{E}: \quad v_{j,t} = v_{i,t} - 2(r_{ij,t}\underline{p}_{ij,t}, x_{ij,t}\underline{q}_{ij,t}) + \cancel{(r_{ij,t}^2 + x_{ij,t}^2)\ell_{ij,t}} \quad \text{(3b)}$$

$$\forall (i,j) \in \mathcal{E}: \quad r_{ij}, x_{ij} \geq 0 \quad \text{(3c)}$$

where $\boldsymbol{v} = (v_{j,t} | \ \forall j \in \mathcal{B},\ \forall t \in \mathcal{T}^{\text{hist}})$ and $\boldsymbol{z} = (r_j, x_{ij} | \ \forall (i,j) \in \mathcal{E})$ are the vectors that aggregate the desicion variables. In the proposed formulation, we utilize a linearized voltage equation [18] as in (3b). The main reason of this is to preserve the convexity of the problem. If we utilize the original voltage equation by including the losses term (the strikeout symbols) instead, the constraint becomes non-linear with respect to $\boldsymbol{r}, \boldsymbol{x}$ and the problem becomes non-convex. Lastly, the constraint (3c) represents the typical characteristic of lines that are resistive and inductive.

2.2.2. *Distributed Impedance Estimation*

In this section, we propose a distributed methodology to solve the impedance estimation problem presented in Section 2.2.1. The main advantage of distributed approach is that prosumers do not necessarily have to know the whole topology of the grid. Rather, they only require to know which prosumers they are connected with. Moreover, we can envision the distributed scenario as a coordination of prosumers without any third-party involvement (e.g., aggregator or DSO). In this case, prosumers only have access to local information and they can exchange information to its adjacent/neighboring prosumers.

The distributed scenario is illustrated in Fig. 4. Each prosumer indeed can easily access its local information (i.e., voltage and bus power) through their smart meter. Other grid state variables, such as branch power flows or voltage of the adjacent buses, can be determined in the course of coordination within the C-ADMM scheme.

As shown previously in Fig. 3, we can formulate the first stage as a distributed problem by relying on coordination among prosumers. On the other hand, the second stage can be solved individually by each prosumer without relying on any communication. The details of each stage of the distributed scenario are presented below.

Stage 1: distributed $p_{ij,t}$ and $q_{ij,t}$ lower bound estimation. For the distributed problem, we apply a consensus alternating direction method of multipliers (C-ADMM) [19] to solve the original problem (1) in a distributed-manner. The main principle of the C-ADMM is to decompose the augmented Lagrangian of a centralized optimizaton problem into several sub-problems, so that the problems can be disseminated to control agents (i.e., the prosumers). Each prosumer a then solves its respective sub/local problem and exchange information in the form of *local variables* ($\boldsymbol{x}_a$) with its adjacent prosumers. This process is done iteratively until everyone reach a consensus toward the global solution.

Let $\mathcal{E}_a \subset \mathcal{E}$ be the set of lines that connect prosumer a to its adjacent prosumers. For each prosumer a, the ADMM variables considered for our Stage 1 problem are described in (4).

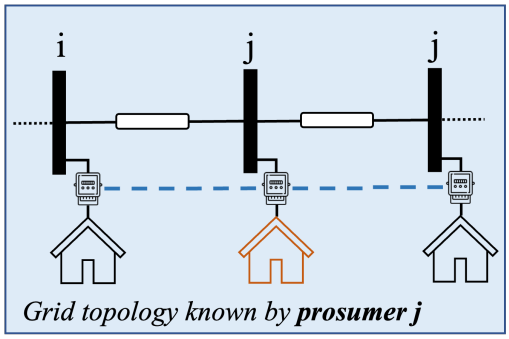
$$\boldsymbol{x}_a = (\underline{p}_{ij,t(j)}, \underline{q}_{ij,t(j)} | (i,j) \in \mathcal{E}_a,\ t \in \mathcal{T}^{\text{hist}}) \quad \text{(local variables)} \quad \text{(4a)}$$

$$\hat{\boldsymbol{x}}_a = (\underline{p}_{ij,t}, \underline{q}_{ij,t} | (i,j) \in \mathcal{E}_a,\ t \in \mathcal{T}^{\text{hist}}) \quad \text{(consensus variables)} \quad \text{(4b)}$$

$$\boldsymbol{\lambda}_a = (\lambda^{\underline{p}}_{ij,t}, \lambda^{\underline{q}}_{ij,t} | (i,j) \in \mathcal{E}_a,\ t \in \mathcal{T}^{\text{hist}}) \quad \text{(dual variables)} \quad \text{(4c)}$$

Then, let k be the C-ADMM iteration number and let the superscript $\cdot^{\{k\}}$ denotes the state of any variables/vectors at the iteration k. For each iteration k, a generic C-ADMM formulation consist of the following steps:

1) Local optimization: In the first step, each prosumer a performs a local optimization to determine the optimal local variables ($\boldsymbol{x}_a^{\{k\}}$) that will be shared to the adjacent prosumers. The considered local optimization of each prosumer a is described in (5), with the main objective is similar as in (3a) but only considering the prosumer's individual bus powers. The decision vari-



Variable	Data Accessibility for house j
Voltage	**Local** (measurement) **& adjacent prosumers** (communication)
Bus power	**Local** (measurement)

: Measurement device

: Communication/information link

Figure 4: Illustration of the distributed scenario.

ables consist of the local variables $\boldsymbol{x}_a^{\{k\}}$ and the vector $\boldsymbol{y}_a^{\{k\}}$ that aggregates the net bus powers $(p_{j,t}^{\text{net}}, q_{j,t}^{\text{net}})$ over the time horizon. Note that, we refer bus j in (5) to as the bus where the prosumer a is located.

$$\min_{\boldsymbol{x}_a^{\{k\}}, \boldsymbol{y}_a^{\{k\}}} \underbrace{\sum_{t\in\mathcal{T}^{\text{hist}}} (p_{j,t}^{\text{net}} - p_{j,t}^{\text{net}})^2 + (\hat{q}_{j,t}^{\text{net}} - q_{j,t}^{\text{net}})^2}_{\text{principal objective}} + \boldsymbol{\lambda}_a^{\{k-1\}} \cdot (\boldsymbol{x}_a^{\{k\}} - \hat{\boldsymbol{x}}_a^{\{k-1\}}) + \frac{\rho}{2}\left\|\boldsymbol{x}_a^{\{k\}} - \hat{\boldsymbol{x}}_a^{\{k-1\}}\right\|_2^2 \quad (5)$$

subject to to similar constraints as in (2), $\forall t \in \mathcal{T}^{\text{hist}}$, $\forall (i,j) \in \mathcal{E}_a$

where $\rho \geq 0$ denotes the C-ADMM penalty parameter (i.e., convergence rate) and $\hat{\boldsymbol{x}}_a^{\{k-1\}}, \boldsymbol{\lambda}_a^{\{k-1\}}$ are parameters obtained from the previous iteration. Additional objetive term is integrated in (5). This part of the objective minimizes the difference between the values of the local $\boldsymbol{x}_a^{\{k\}}$ and the consensus $\hat{\boldsymbol{x}}_a^{\{k-1\}}$ variables, weighted by the dual variables $\boldsymbol{\lambda}_a^{\{k-1\}}$ and ρ. This term is added due to the fact that the consesus variables can be interpreted as values in which all prosumers should ultimately agree. Each prosumer a then shares the optimal local variables vector $\boldsymbol{x}_a^{\{k\}}$ obtained in this step to its adjacent prosumers. Remind that by using the proposed coordination methodology, prosumers are not required to share their *behind the meter* assets data (i.e., load and PV), which are considered as sensitive information and potentially harmful for their privacy [20].

2) Consensus update: Next, each prosumer a receives local variable vectors from its adjacent prosumers, and utilizes this information to update locally the consensus variables $\hat{\boldsymbol{x}}_a^{\{k\}}$. The local variables received from the adjacent prosumers can be different in values as the ones computed locally (in the previous step). Therefore, the goal of the consensus variables is to drive the local variables of all the agents to reach a "*consensus*" state in each iteration.

The consensus variables are updated by taking the average of the locally computed variables and the values received from the adjacent prosumers. For each prosumer a, the consensus variables are updated as in (6)

$$\forall (i,j) \in \mathcal{E}_a : p_{ij}^{\{k\}} = \langle p_{ij(a)}^{\{k-1\}},\ p_{ij(n:n\in M_{a,ij})}^{\{k-1\}} \rangle \quad (6a)$$

$$q_{ij}^{\{k\}} = \langle q_{ij(a)}^{\{k-1\}},\ q_{ij(n:n\in M_{j,ij})}^{\{k-1\}} \rangle \quad (6b)$$

where the sets $M_{a,ij}$ consist of all the adjacent prosumers that also consider p_{ij}, q_{ij} as their local/global variables. Note that, the elements of $\boldsymbol{x}_a$ are denoted similarly to those of $\hat{\boldsymbol{x}}_a$ but with added subscript $\cdot_{(a)}$. This subscript implies that the variable is computed/owned by prosumer a and used to differentiate from the same variables computed by other prosumers.

3) Dual update: In the last step, all the prosumers update the dual variables using gradient ascent principles [19], as expressed in (7).

$$\boldsymbol{\lambda}_a^{\{k\}} = \boldsymbol{\lambda}_a^{\{k-1\}} + \frac{\rho}{2}(\boldsymbol{x}_a^{\{k\}} - \boldsymbol{z}_a^{\{k\}}) \quad (7)$$

The algorithm then repeats and starts the next iteration $k+1$. It will terminate when it reaches a maximum number of iteration k_{max}.

Stage 2: Impedance estimation. For the second stage, identical problem as (3) is utilized, but with only considering the upstream branch as the constraints, as presented in (8).

$$\min_{\boldsymbol{v}, r_{ij}, x_{ij}} \sum_{t\in\mathcal{T}^{\text{hist}}} (\hat{v}_{i,t} - v_{i,t})^2 + (\hat{v}_{j,t} - v_{j,t})^2 \quad (8a)$$

subject to:

$$\forall t \in \mathcal{T}^{\text{hist}},\ \forall (i,j) \in \mathcal{E}_a^{\text{up}}: \qquad v_{j,t} = v_{i,t} - 2(r_{ij,t}\underline{p}_{ij,t}, x_{ij,t}\underline{q}_{ij,t}) \tag{8b}$$

$$\forall (i,j) \in \mathcal{E}_a^{\text{up}}: \quad r_{ij}, x_{ij} \geq 0 \tag{8c}$$

where the considered decision variables consist of r_{ij}, x_{ij} and the vector $\boldsymbol{v} = (v_{j,t} | \ \forall t \in \mathcal{T}^{\text{hist}})$. In addition, $\mathcal{E}_a^{\text{up}} \subset \mathcal{E}_a$ is the set of upstream branches of prosumer a.

2.3. Strategic Impedance Tuning Algorithm

The proposed centralized and distributed impedance estimation algorithms introduced in Section 2.2 may not guarantee accurate impedance values. All the methodologies rely on historical data and the quantity of the data will certainly influence the estimation accuracy. Moreover, as we will study further in Section 3.3, the presence of measurement errors affects the estimation accuracy. In order to increase the preciseness of the impedance estimation, we propose a strategic method to successively fine-tune the impedances, as presented in Fig. 5. Generally, the impedances are tuned after several rounds of voltage management that we defined as *evaluation* period. The proposed tuning scheme can be conducted either in a centralized or a distributed manner, as we will further describe in this subsection.

At the very first step, we utilize the initial estimated impedances returned from the impedance estimation algorithm (centralized or distribute approach depending on the architecture) as the "*starting point*" values of the parameters in our voltage controller. Then, we perform voltage management with these impedances over a certain evaluation time period $\mathcal{T}^{\text{eval}}$. At the end of the evaluation time, we compare the actual voltage and bus powers measurements with the estimated values returned by the controller, and tune the impedances based on the proposed rules. This procedure then repeats at every evaluation period.

In this paper, we consider a half-day time horizon for the evaluation period $\mathcal{T}^{\text{eval}}$ with 30 min time step (the time step in which voltage management is performed and measurements are extracted). This time horizon is selected because we aim to target a frequent update of the impedances and to have input measurements that can capture the variability of night and day profiles. Over the considered evaluation time $\mathcal{T}^{\text{eval}}$, the following information is utilized in the proposed tuning algorithm:

1. Actual voltage drop computed from the measurements (9) and estimated voltage drop computed (estimated) by the controller (10) at each branch.

$$\Delta\hat{v}_{ij,t} = \hat{v}_{j,t} - \hat{v}_{i,t} \tag{9}$$

$$\Delta v_{ij,t} = v_{j,t} - v_{i,t} \tag{10}$$

2. voltage drop deviation between the measurement and the controller estimation (11).

$$\vartheta_{ij,t} = \Delta\hat{v}_{ij,t} - \Delta v_{ij,t} \tag{11}$$

3. Active and reactive branch power flow direction. This information can be retrieved from the output of the controller ($p_{ij,t}, q_{ij,t}$). In our observation, although the magnitude of the branch power flow returned by the controller may not be precise (due to inacurracy of impedance parameters), the power flow directions, however, are accurate.

The proposed tuning strategy increases/decreases the impedances values based on the observed voltage drop deviations ($\vartheta_{ij,t}$) at each time t. This is done in two steps, as presented in Algorithm 1 and 2.

In the first step (Algorithm 1), our goal is to compute the *action variables* for both the resistance (τ_{ij}^r) and the reactance (τ_{ij}^r) of each branch. These variables will determine if we shall either tune the resistance or the reactance for each impedance at the end of evaluation period. Principally, we would like to tune the impedance if the controller prediction on voltage drop ($\Delta v_{ij,t}$) is far-away from the actual measurement value ($\Delta\hat{v}_{ij,t}$). This criterion can be described when $|\Delta v_{ij,t} - \Delta\hat{v}_{ij,t}| > \epsilon_\vartheta$, with ϵ_ϑ is configured to 10^{-5} in this paper. On the opposite, we consider an accepted estimation accuracy when $|\Delta v_{ij,t} - \Delta\hat{v}_{ij,t}| \leq \epsilon_\vartheta$.

The tuning rules are formulated based on the voltage equation (3b). At each time step t, the algorithm observes if $\Delta v_{ij,t}$ is either under-estimated or over-estimated for each branch. In fact, this can be defined by looking at $\vartheta_{ij,t}$ of each branch, as presented in Algorithm 1, line 4 and 7. When $\Delta v_{ij,t}$ is under-estimated, our goal is to decrease the term $r_{ij}p_{ij,t} + x_{ij}q_{ij,t}$ since it has substracting effect to the voltage drop (3b). On the opposite, the term $r_{ij}p_{ij,t} + x_{ij}q_{ij,t}$ shall be increased when $\Delta v_{ij,t}$ is over-estimated.

The active and reactive power flow directions further determine the action that has to be taken for the resistance and the reactance (either increase or decrease it). The direction of the power flows can be determined by simply looking at their sign (i.e., $\text{sgn}(p_{ij,t}), \text{sgn}(q_{ij,t})$). To recall, the power flows are returned by the controller, not from the measurements. Then, at each time step, (τ_{ij}^r) records if the branch resistance (r_{ij}) should in-

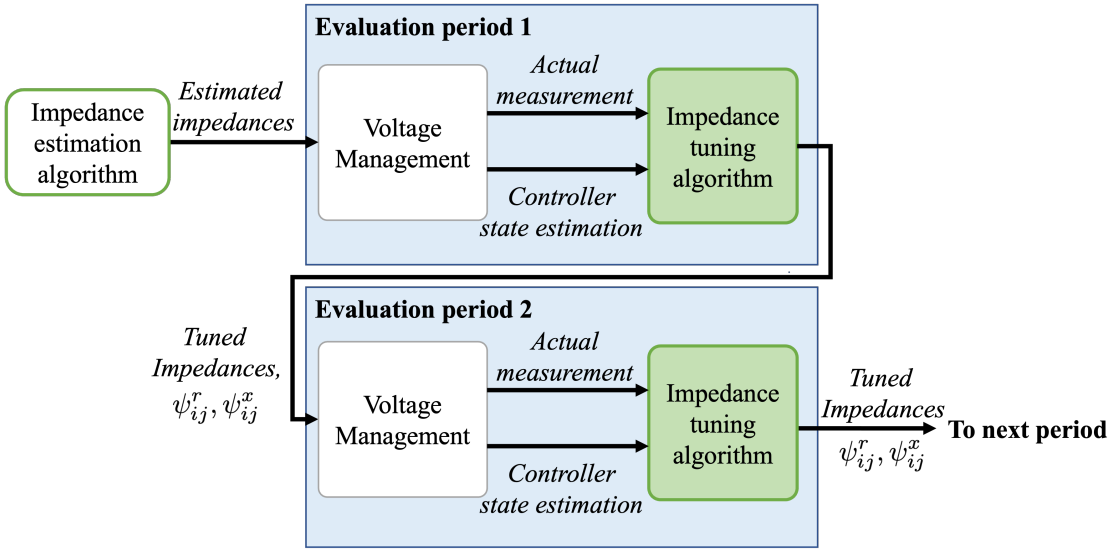


Figure 5: The proposed strategic impedance tuning scheme method.

creases (+1) or decreases (−1), while (τ^x_{ij}) records it for the reactance (x_{ij}). The *action variables* for each line (τ^r_{ij}, τ^x_{ij}) are updated over the whole evaluation period, and their final values will decide the action to be given to both r_{ij} and x_{ij} for each line.

Algorithm 1: *Strategic impedance tuning step 1: action variables* computation

for $(i, j) \in \mathcal{E}$ **do**
  $\tau^r_{ij} \leftarrow 0$ and $\tau^x_{ij} \leftarrow 0$
  for $t \in \mathcal{T}^{eval}$ **do** `/* Evaluate at each time step */`
    if $\vartheta_{ij,t} \geq \epsilon_\vartheta$ **then** `/* Voltage drop under-estimated */`
      $\tau^r_{ij} \leftarrow \tau^r_{ij} - \text{sgn}(p_{ij,t})$;
      $\tau^x_{ij} \leftarrow \tau^x_{ij} - \text{sgn}(q_{ij,t})$;
    else if $\vartheta_{ij,t} \leq -\epsilon_\vartheta$ **then** `/* Voltage drop over-estimated */`
      $\tau^r_{ij} \leftarrow \tau^r_{ij} + \text{sgn}(p_{ij,t})$;
      $\tau^x_{ij} \leftarrow \tau^x_{ij} + \text{sgn}(q_{ij,t})$;
    else
      Continue ;
    end
  end
end

Algorithm 2: *Strategic impedance tuning step 2:* Action decision

for $(i, j) \in \mathcal{E}$ **do**
  if $|\tau^r_{ij}| > |\tau^x_{ij}|$ **then** `/* Decision to act on resistance */`
    if $action^r_{ij} = -sgn(\tau^r_{ij})$ **then**
      $\psi^r_{ij} \leftarrow \psi^r_{ij} * \phi$; `/* Decrease to prevent oscillation */`
    end
    $r_{ij} \leftarrow r_{ij} * \left(1 + (\psi^r_{ij} \cdot \text{sgn}(\tau^r_{ij}))\right)$;
    $action^r_{ij} \leftarrow \text{sgn}(\tau^r_{ij})$;
  else if $|\tau^r_{ij}| < |\tau^x_{ij}|$ **then** `/* Decision to act on reactance */`
    if $action^x_{ij} = -sgn(\tau^x_{ij})$ **then**
      $\psi^x_{ij} \leftarrow \psi^x_{ij} * \phi$; `/* Decrease to prevent oscillation */`
    end
    $x_{ij} \leftarrow x_{ij} * \left(1 + (\psi^x_{ij} \cdot \text{sgn}(\tau^x_{ij}))\right)$;
    $action^x_{ij} \leftarrow \text{sgn}(\tau^x_{ij})$;
  else
    Continue ;
  end
end

The objective of the second step (Algorithm 2) is to decide either to act on r_{ij} or x_{ij} based on the obtained *action variables* of each line. Simply, we act based on the absolute largest *action variable* (line 2 and 20). This variable can represent the most frequent tuning action (i.e., increase/decrease either r_{ij} or x_{ij}) identified during the evaluation time, as described in Algorithm 1. Then, the final tuning action can be decided by observing the sign of this largest *action variables*. If it is positive, the final action will be to increase the corresponding largest impedance component, i.e., either the resistance or the reactance, as described in line 11 and 29 of Algorithm 2, respectively. On the opposite, we will decrease the corresponding largest impedance component when the obtained *action variable* is negative (line 4 and 21 of Algorithm 2).

For each line, the impedance is updated using a tuning parameter ψ^r_{ij} (for resistance) and ψ^x_{ij} (for reactance), which are initially set to 0.1. From our observation, this value is not excessively high (to prevent tuning oscillation) or extremely low (to prevent slow convergence). Additionally, we also implement an adaptive parameter update (i.e., line 3 and 9 of Algorithm 2) to prevent oscillation during impedance tuning. In this control function, we decrease ψ^r_{ij} and ψ^x_{ij} by ϕ (50 % in this paper) if the obtained tuning action differs from the past. $action^r_{ij} = 1$ means that the we increased r_{ij} in the last action, while we decreased r_{ij} if $action^r_{ij} = -1$. Similar rules are also applied for $action^x_{ij}$.

As depicted in Fig. 5, the performance of the tuning algorithm depends on the initial values of the impedance estimation. The preciseness of the initial estimation undoubtedly influences the number of tuning rounds required until the impedances reach the convergence values (ϵ_ϑ). Despite of that, the main advantage of the proposed tuning algorithm is that it only requires voltage measurements. Therefore, the algorithm can be carried out either centrally or locally by each prosumer in the context of distributed architecture. The only requirement in distributed approach is that each prosumer shall have the information of its neighbors' voltage measurements, which can be easily retrieved using a single a round of communication (unlike the C-ADMM approach that requires a few rounds of iteration).

3. Simulations and Results

3.1. Case Study

In this paper, the IEEE 33-bus distribution system [21] with additional integration of ten 1 MWp of PV units is considered as the primary test system to evaluate the proposed methodologies. The PV profiles utilized in the test system are obtained from real irradiance data in Grenoble, France, and the load profiles are adopted from the database that are available in [22] using a 30 min time step profiles. All the tests conducted in this paper were performed using a computer with an Intel® i3-430U processor and 8 GB of RAM.

In order to assess the performance of the proposed methodologies, a voltage error criterion (δ_v) is defined. This index measures how accurate the voltage controller is (using the estimated impedances) to compute the voltage compared to the actual measurements. The formulation of this computation criteria is obtained by the study from [23], in which the voltage errors are normalized with regard to the mean deviation of the voltage computed over the total number of busses (12).

$$\delta_v(\%) = \frac{100}{|\mathcal{B}|} \sum_{t\in\mathcal{T}} \sum_{i\in\mathcal{B}} \frac{|v_{i,t} - \hat{v}_{i,t}|}{\frac{1}{|\mathcal{B}|} \sum_{t\in\mathcal{T}} \sum_{i\in\mathcal{B}} |1 - \hat{v}_{i,t}|} \tag{12}$$

3.2. Validation of the Methodology

The first objective of the test is to assess the performance of the centralized and distributed impedance estimation algorithms. We test four different amount of input scenarios (i.e., one to four days of measurements data) for each proposed methodology. After the estimated impedances are obtained, we conduct a validation test by running the voltage controller over one day simulation horizon of 30 minutes time step and using determinstic PV and load profiles, as shown in Fig. 6. Moreover, we utilize an open-source software Pandapower [24] to emulate a real distribution grid and to generate emulated RMS voltage measurements at each 30 minutes over the considered simulation horizon. By doing this, it will allow us to compute the δ_v that will be used to evalute the accuracy of the impedance estimation. Note that, we do not display the resulting voltage profiles with voltage control since it is not the main focus on this paper.

Fig. 7 shows the resulting δ_v as a function of the number of input data for both centralized and distributed methodologies. The proposed algorithms can predict the voltage values with only around 7 % of voltage estimation error with only utilizing one-day of measurement data. It should be noted that we assume that we have no knowledge of the impedance values prior running the algortihms. Both distributed and centralized methods return comparable performances, where the obtained δ_v are still close from one method to another.

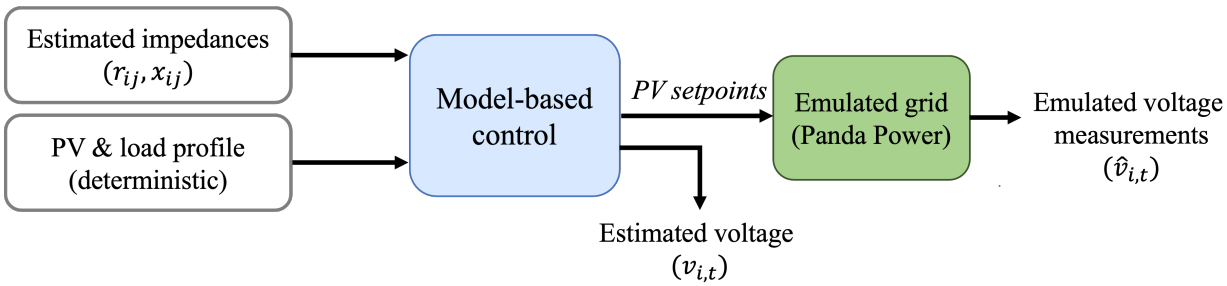


Figure 6: Simulation flow chart for validation test.

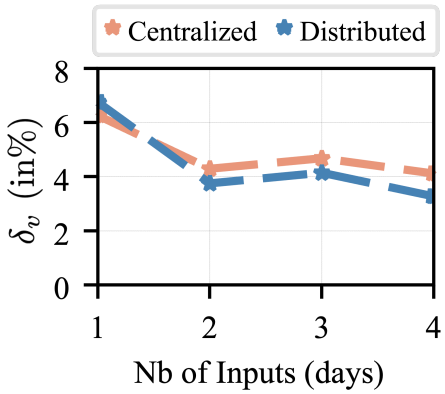


Figure 7: Comparison of δ_v between centralized and distributed methods.

Furthermore, the accuracy of the impedance estimation can also be further improved by increasing the number of input data. We only consider four different input scenarios due to the computational time reason. As shown in Table 2, there is a significant difference in computational time between the centralized and distributed methods. This is obviously due to the iterative nature of the distributed algorithm (i.e., C-ADMM) that requires several rounds of iterations before reaching the convergence state.

Our next objective is to observe the effectiveness of the proposed strategic tuning algorithm in order to further improve the accuracy of impedance estimation. In this test, we tune the impedances successively over a 10 days of simulation horizon, with the evaluation time horizon ($\mathcal{T}^{\text{eval}}$) of 12 hours (i.e., two evaluations per day).

Fig. 8 shows the evolution of δ_v over 10 days, where δ_v is computed at the end of each evaluation period (every 12 hours) and day 0 refers to the initial estimated impedance (before the tuning). The results show that the proposed tuning algorithm is effective, and able to lower δ_v below 1 % for all the methods at the end of the simulation period.

To better visualize the obtained results, Fig. 9 shows the absolute difference between the estimated (i.e., voltage state returned by the controller) and the actual voltage drop (based on the measurements), which is the $|\vartheta_{ij,t}|$, over all branches at a particular time step. Fig. 9(a) shows the obtained $|\vartheta_{ij,t}|$ by using the estimated impedance with 1 day of training data (Fig. 7). Yet, there are still some branches that could not achieve the tolerance accuracy (i.e., $|\vartheta_{ij,t}| \geq \epsilon_\vartheta$). Fig. 9(b) shows $|\vartheta_{ij,t}|$ with the final impedances after 10 days of tuning. The result shows that the tuning algorithm is very effective to fine-tune the impedance. Ultimately, $|\vartheta_{ij,t}| \leq \epsilon_\vartheta$ can be achieved for all branches.

3.3. Robustness Test

The final study aims at validating the robustness of the methodology to measurement errors and/or non-synchronized measurements. To simulate this, we apply a ramdom noise [25] with a standard deviation of 2 % (a typical allowable range of measurement error [26]) to the measurements data.

The impact of measurement errors is very significant, as shown in Fig. 10. Compared to the reference values, i.e., a maximum of 7 % with one-day ideal input

Table 2: Comparison of computational time (in seconds) between different methodologies.

	Input data			
	1 day	2 days	3 days	4 days
Centralized	8	17	33	48
Distributed	950	1773	2025	3500

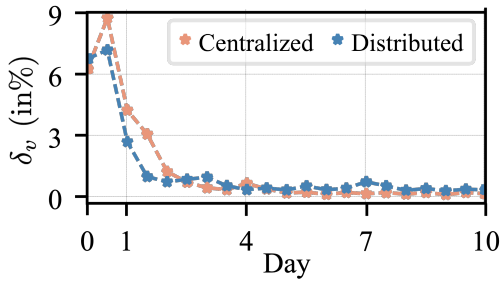


Figure 8: Evolution of δ_v with the tuning algorithm.

measurements (Fig. 7), both centralized and distributed methods with measurement noise end up with a significant voltage error (δ_v).

By applying the tuning algorithm, the accuracy of the impedance estimation can be massively improved. In the simulations, we try to tune the impedance over a year time horizon using similar evaluation period of 12 hours. We can see from Fig. 11 that δ_v converges toward small values after around 80 days of tuning. At the final evaluation period, δ_v can reach a value of around 1 %, which is a vast improvement compared to the initial estimated impedance without the tuning (above 100 % error).

Lastly, we intent to observe the effectiveness of the tuning algorithm under three additional noise levels – 5 %, 8 % and 10 %. Due to the computational time reason, the final tests only conducted with centralized method and with one-year of tuning horizon ($\mathcal{T}^{\text{eval}}$) using the same evaluation period of 12 hours.

As highlighted in Fig. 12, the algorithm is able to significantly improve δ_v after several days of tuning rounds. We can see the impact of higher noise levels at the final evaluation period, where the obtained δ_v is higher as we impose more measurement noise. This also implies that the algorithm needs more evaluation days in order to reach the same level of δ_v as the previous case with 2 % of measurement error. The performance of the tuning algorithm can also be improved by a proper selection of the tuning parameters (i.e., ψ^r_{ij} and ψ^x_{ij}) or by improving the criteria in the adaptive parameter update (Algorithm 2). However, all of the aforementioned strategies have not been studied yet in this paper.

4. Conclusion

In this paper, methodologies to estimate grid impedances (i.e., line resistance and reactance) using historical measurements data are presented. We proposed two types of approaches: i) centralized and ii) distributed that can recover line impedances using only voltage and bus power data.

The first simulations showed that both centralized and distributed methodologies were able to estimate the grid impedance within an acceptable accuracy (minimum 2 % and maximum 7 % of error). In order to improve the preciseness of the estimation, a strategic impedance tuning scheme was also proposed to successively fine-tune the impedances. The second part of the simulations highlighted the effectiveness of the tuning strategy, which it turns was able to significantly improve the accuracy of the impedance estimation to under 1 %.

The final part of this chapter focused on robustness tests of the algorithm, by imposing measurement noises into account. The impact on measurement noise to the proposed estimation algorithms is significant, where the estimation error can reach values up to 100 times higher than the ones from the case with ideal measurements. Despite of that, the proposed tuning algorithm is yet very effective to improve this massive error, lowering

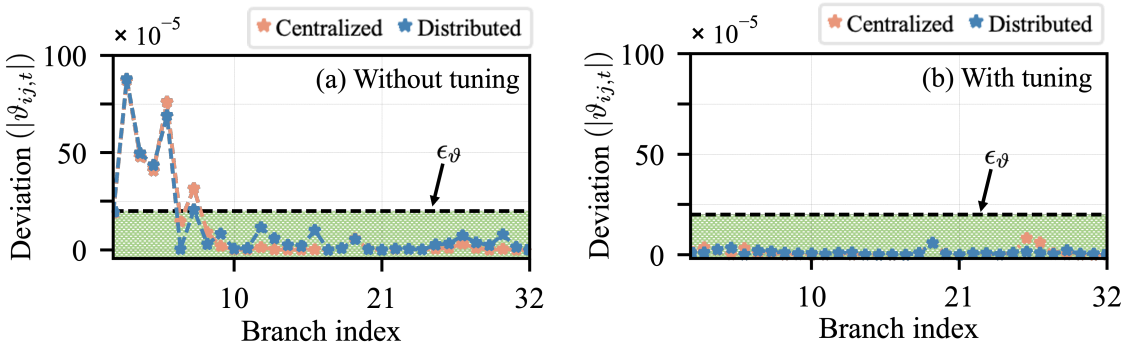


Figure 9: Resulting $\vartheta_{ij,t}$ at a particular time step: (a) without tuning and (b) with tuning.

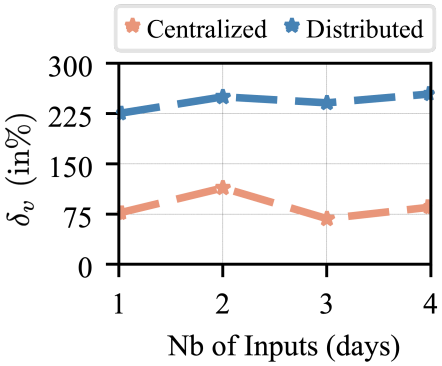


Figure 10: Comparison of δ_v with measurement noise between centralized and distributed methods.

it to 4 % even further to 1 % after several tuning rounds. Finally, it is worth reminding that the controller tuning occurs over successive days which motivated the simulation setup to operate in steady state (at a 30 min resolution here). The proposed approach shall then be further validated with closer to real time simulations and while embedding more realistic measurements errors (e.g. lag and synchronization issues).

References

[1] A. Hirsch, Y. Parag, J. Guerrero, Microgrids: A review of technologies, key drivers, and outstanding issues, Renewable and Sustainable Energy Reviews 90 (2018) 402–411. doi:https://doi.org/10.1016/j.rser.2018.03.040.

[2] M. Gržanić, T. Capuder, N. Zhang, W. Huang, Prosumers as active market participants: A systematic review of evolution of opportunities, models and challenges, Renewable and Sustainable Energy Reviews 154 (2022) 111859. doi:https://doi.org/10.1016/j.rser.2021.111859.

[3] P. N. Vovos, A. E. Kiprakis, A. R. Wallace, G. P. Harrison, Centralized and distributed voltage control: Impact on distributed generation penetration, IEEE Transactions on power systems 22 (1) (2007) 476–483.

[4] M. Ebeed, S. Kamel, F. Jurado, Chapter 7 - optimal power flow using recent optimization techniques, in: A. F. Zobaa, S. H. Abdel Aleem, A. Y. Abdelaziz (Eds.), Classical and Recent Aspects of Power System Optimization, Academic Press, 2018, pp. 157–183. doi:https://doi.org/10.1016/B978-0-12-812441-3.00007-0.

[5] J. Yu, Y. Weng, R. Rajagopal, Patopa: A data-driven parameter and topology joint estimation framework in distribution grids, IEEE Transactions on Power Systems 33 (4) (2018) 4335–4347. doi:10.1109/TPWRS.2017.2778194.

[6] H. Li, Y. Weng, Y. Liao, B. Keel, K. E. Brown, Distribution grid impedance & topology estimation with limited or no micro-pmus, International Journal of Electrical Power & Energy Systems 129 (2021) 106794. doi:https://doi.org/10.1016/j.ijepes.2021.106794.

[7] P. Zarco, A. Exposito, Power system parameter estimation: a survey, IEEE Transactions on Power Systems 15 (1) (2000) 216–222. doi:10.1109/59.852124.

[8] X. Miao, M. Ilić, X. Wu, U. Münz, Distribution grid admittance estimation with limited non-synchronized measurements, in: 2019 IEEE Power Energy Society General Meeting (PESGM), 2019, pp. 1–5. doi:10.1109/PESGM40551.2019.8973562.

[9] M. Putratama, R. Rigo-Mariani, V. Debusschere, y. Bésanger, Uncertainties Impact and Mitigation with an Adaptive Model-Based Voltage Controller, in: ELECTRIMACS 2022, Nancy, France, 2022.

[10] S. Park, D. Deka, S. Backhaus, M. Chertkov, Learning with end-users in distribution grids: Topology and parameter estimation, IEEE Transactions on Control of Network Systems 7 (3) (2020) 1428–1440. doi:10.1109/TCNS.2020.2979882.

[11] D. E. Olivares, A. Mehrizi-Sani, A. H. Etemadi, C. A.

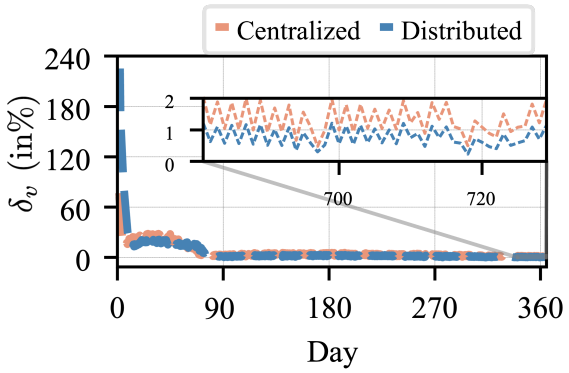


Figure 11: Evolution of δ_v with the tuning algorithm in the presence of measurement noise.

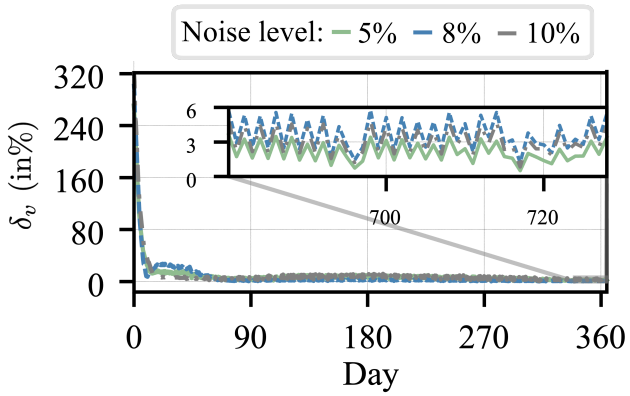


Figure 12: Evolution of δ_v with the tuning algorithm (centralized scenario) with three different noise levels.

Cañizares, R. Iravani, M. Kazerani, A. H. Hajimiragha, O. Gomis-Bellmunt, M. Saeedifard, R. Palma-Behnke, et al., Trends in microgrid control, IEEE Transactions on smart grid 5 (4) (2014) 1905–1919.

[12] M. A. Putratama, R. Rigo-Mariani, V. Debusschere, Y. Besanger, Parameter tuning for lv centralized and distributed voltage control with high pv production, in: 2021 IEEE Madrid PowerTech, 2021, pp. 1–6. doi:10.1109/PowerTech46648.2021.9494802.

[13] S. Alyami, Y. Wang, C. Wang, J. Zhao, B. Zhao, Adaptive real power capping method for fair overvoltage regulation of distribution networks with high penetration of pv systems, IEEE Transactions on Smart Grid 5 (6) (2014) 2729–2738.

[14] A. D. Mustika, R. Rigo-Mariani, V. Debusschere, A. Pachurka, A two-stage management strategy for the optimal operation and billing in an energy community with collective self-consumption, Applied Energy 310 (2022) 118484.

[15] T. Morstyn, M. D. McCulloch, Multiclass energy management for peer-to-peer energy trading driven by prosumer preferences, IEEE Transactions on Power Systems 34 (5) (2018) 4005–4014.

[16] M. A. Putratama, R. Rigo-Mariani, A. D. Mustika, V. Debusschere, A. Pachurka, Y. Bésanger, A three-stage strategy with settlement for an energy community management under grid constraints, IEEE Transactions on Smart Grid (2022) 1–1doi:10.1109/TSG.2022.3167862.

[17] M. Farivar, S. H. Low, Branch flow model: Relaxations and convexification—part i, IEEE Transactions on Power Systems 28 (3) (2013) 2554–2564. doi:10.1109/TPWRS.2013.2255317.

[18] M. Baran, F. Wu, Network reconfiguration in distribution systems for loss reduction and load balancing, IEEE Transactions on Power Delivery 4 (2) (1989) 1401–1407. doi:10.1109/61.25627.

[19] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Foundations and Trends® in Machine Learning 3 (1) (2011) 1–122. doi:10.1561/2200000016.

[20] E. Liu, P. Cheng, Achieving privacy protection using distributed load scheduling: A randomized approach, IEEE Transactions on Smart Grid 8 (5) (2017) 2460–2473. doi:10.1109/TSG.2017.2703400.

[21] V. Vita, Development of a decision-making algorithm for the optimum size and placement of distributed generation units in distribution networks, Energies 10 (9). doi:10.3390/en10091433.

[22] D. Murray, L. Stankovic, V. Stankovic, An electrical load measurements dataset of united kingdom households from a two-year longitudinal study, Scientific data 4 (1) (2017) 1–12.

[23] R. Rigo-Mariani, V. Vai, An iterative linear distflow for dynamic optimization in distributed generation planning studies, International Journal of Electrical Power & Energy Systems 138 (2022) 107936. doi:https://doi.org/10.1016/j.ijepes.2021.107936.

[24] L. Thurner, A. Scheidler, F. Schäfer, J. Menke, J. Dollichon,

F. Meier, S. Meinecke, M. Braun, pandapower — an open-source python tool for convenient modeling, analysis, and optimization of electric power systems, IEEE Transactions on Power Systems 33 (6) (2018) 6510–6521. doi:10.1109/TPWRS.2018.2829021.

[25] K. Li, State estimation for power distribution system and measurement impacts, IEEE Transactions on Power Systems 11 (2) (1996) 911–916. doi:10.1109/59.496174.

[26] X. Kong, X. Zhang, N. Lu, Y. Ma, Y. Li, Online smart meter measurement error estimation based on ekf and lmrls method, IEEE Transactions on Smart Grid 12 (5) (2021) 4269–4279. doi:10.1109/TSG.2021.3077693.