Privacy-preserving Distributed Learning for Renewable Energy Forecasting

Carla Gonçalves, Ricardo J. Bessa, *Senior Member, IEEE* and Pierre Pinson, *Fellow, IEEE*

***Abstract*—Data exchange between multiple renewable energy power plant owners can lead to an improvement in forecast skill thanks to the spatio-temporal dependencies in time series data. However, owing to business competitive factors, these different owners might be unwilling to share their data. In order to tackle this privacy issue, this paper formulates a novel privacy-preserving framework that combines data transformation techniques with the alternating direction method of multipliers. This approach allows not only to estimate the model in a distributed fashion but also to protect data privacy, coefficients and covariance matrix. Besides, asynchronous communication between peers is addressed in the model fitting, and two different collaborative schemes are considered: centralized and peer-to-peer. The results for a solar energy dataset show that the proposed method is robust to privacy breaches and communication failures, and delivers a forecast skill comparable to a model without privacy protection.**

***Index Terms*—Renewable energy, forecasting, vector autoregression, privacy-preserving, distributed learning**

I. Introduction

THE forecast skill of renewable energy sources (RES) has improved over the past two decades through R&D activities across the complete model chain, i.e., from numerical weather predictions (NWP) to statistical learning methods that convert weather variables into power forecasts [1]. The need to bring forecast skill to significantly higher levels is widely recognized in the majority of roadmaps that deal with high RES integration scenarios for the next decades. This is expected not only to facilitate RES integration in the system operation and electricity markets but also to reduce the need for flexibility and associated investment costs on remedies that aim to hedge RES variability and uncertainty like storage, demand response, and others.

In this context, intraday and hour-ahead electricity markets are becoming increasingly important to handle RES uncertainty and thus accurate hours-ahead forecasts are essential. Recent findings showed that feature engineering, combined with statistical models, can extract relevant information from spatially distributed weather and RES power time series and improve hours-ahead forecast skill [1]. Indeed, for very short-term lead times (from 15 minutes to 6 hours ahead), the vector autoregressive (VAR) model, when compared to univariate time series models, has shown competitive results for wind [2] and solar [3] power forecasting. Alternative models are also being applied to this problem, most notably deep learning techniques such as convolutional neural networks or long short-term memory networks [4]. While there may always be a debate about the interest and relevance of statistical modeling vs. machine learning approaches, VAR models have the advantages of flexibility, interpretability, acceptability by practitioners, as well as robustness in terms of forecast skill.

Four important challenges for RES forecasting have been identified when using VAR: (a) sparse structure of the coefficients' matrix [5], (b) uncertainty forecasting [6], (c) distributed [7], and online learning [8], and (d) data privacy.

Data privacy is a critical barrier to the application of collaborative forecasting models. Although multivariate time series models offer forecast skill improvement, the lack of privacy-preserving mechanisms makes data owners unwilling to cooperate. For instance, in the VAR model, the covariates are the lags of the target variable of each RES site, which means that agents (or data owners) cannot provide covariates without also providing their power measurements.

To the best of our knowledge, only three works have proposed privacy-preserving approaches for RES forecasting. Zhang and Wang described a privacy-preserving approach for wind power forecasting with off-site time series, which combined ridge linear quantile regression with alternating direction method of multipliers (ADMM) [9]. However, privacy with ADMM is not always guaranteed since it requires intermediate calculations, allowing the most curious competitors to recover the data at the end of several iterations [10]. Moreover, the central node can also recover the original and private data. Sommer et al. [11] considered an encryption layer, which consists of multiplying the data by a random matrix. However, the focus of this work was not data privacy, but rather online learning, and the private data are revealed to the central agent who performs intermediary computations. Berdugo et al. described a method based on local and global analog-search (i.e., template matching) that uses solar power time series from neighboring sites [12]. However, agents only share reference time-stamps and normalized weights of the analogs identified by the neighbors, hence forecast error is only indirectly reduced. In this paper, we also use ADMM as a central

The research leading to this work is being carried out as a part of the Smart4RES project (European Unions Horizon 2020, No. 864337). The sole responsibility of this publication lies with the author. The European Union is not responsible for any use that may be made of the information contained therein. The work of C. Gonçalves was supported by the Portuguese funding agency, FCT (Fundação para a Ciência e a Tecnologia), within the Ph.D. grant PD/BD/128189/2016 with financing from POCH (Operational Program of Human Capital) and the EU. (Corresponding author: Ricardo J. Bessa.)

Carla Gonçalves is with INESC TEC, 4200-465 Porto, Portugal, and also with the Faculty of Sciences, University of Porto, 4169-007 Porto, Portugal (e-mail:carla.s.goncalves@inesctec.pt).

Ricardo J. Bessa is with INESC TEC, 4200-465 Porto, Portugal (e-mail: ricardo.j.bessa@inesctec.pt).

Pierre Pinson is with the Technical University of Denmark, Department of Technology, Management and Economics, 2800 Kongens Lyngby, Denmark (e-mail: ppin@dtu.dk).

framework for distributed learning and forecasting, in view of its flexibility in terms of communication setup for all agents involved, the possibility to add a privacy-preserving layer, as well as the promising resulting forecast skill documented in the literature.

A literature analysis in [10] of privacy-preserving techniques for VAR has grouped these techniques as (a) *data transformation*, such as generation of random matrices that pre- or post-multiply the data [13] or using principal component analysis with differential privacy [14] (b) *secure multi-party computation*, such as linear algebra protocols [15] or homomorphic encryption (encrypting the original data in a way that arithmetic operations in the public space does not compromise the encryption [16]), and (c) *decomposition-based methods* like the ADMM [17] or the distributed Newton-Raphson method [18]. The main conclusions were that *data transformation* requires a trade-off between privacy and accuracy, *secure multi-party computations* either result in computationally demanding techniques or do not fully preserve privacy in VAR models, and that *decomposition-based methods* rely on iterative processes and after a number of iterations, the agents have enough information to recover private data.

With our focus on privacy-preserving protocols for very short-term forecasting with the VAR model, the main research outcome from this work is a novel combination of data transformation and decomposition-based methods so that the VAR model is fitted in another feature space without decreasing the forecast skill (which contrasts with [12]). The main advantage of this combination is that the ADMM algorithm is not affected and therefore: (a) asynchronous communication between peers can be addressed while fitting the model; (b) a flexible privacy-preserving collaborative model can be implemented using two different schemes, centralized communication with a neutral node and peer-to-peer communication, and in a way that original data cannot be recovered by central node or peers (this represents a more robust approach compared to the ADMM implementation in [9], [11]).

The remaining of this paper is organized as follows: Section II describes the distributed learning framework. Section III describes the VAR model and coefficients' estimators. Section IV formulates a novel privacy-preserving LASSO-VAR model. Then, a case study with solar energy data is considered in Section V. The work concludes in Section VI.

II. Distributed Learning Framework

This section discusses the distributed learning framework that enables different agents or data owners (e.g., RES power plant, market players, forecasting service providers) to exploit geographically distributed time series data (power and/or weather measurements, NWP, etc.) and improve forecast skill while keeping data private. In this context, data privacy can either refer to commercially sensitive data from grid-connected RES power plants or personal data (e.g., under European Union General Data Protection Regulation) from households with RES technology. Distributed learning (or collaborative forecasting) means that instead of sharing their data, the model fitting problem is solved in a distributed manner. Two collaborative schemes are possible: centralized communication with a central node (*central hub*) and peer-to-peer communication (*P2P*).

In the *central hub* model, the scope of the calculations performed by the agents is limited by their local data and the only information transmitted to the central node is statistics, e.g., average values or local data multiplied by locally estimated coefficients. The central node is responsible for combining these local estimators and, when considering iterative solvers like ADMM, coordinating the individual optimization processes to solve the main optimization problem. The central node can be either a transmission/distribution system operator (TSO/DSO) or a forecasting service provider. The TSO or DSO could operate a platform that promotes collaboration between competitive RES power plants in order to improve the forecasting accuracy and reduce system balancing costs. On the other hand, the forecasting service provider could host the central node and make available APIs and protocols for information (not data) exchange between different data owners, during model fitting, and receives a payment for this service.

In the P2P, the agents equally conduct a local computation of their estimators, but share their information with peers, meaning that each agent is itself agent and central node. While P2P tends to be more robust (i.e., lower points of failure), it is usually difficult to make it as efficient as the central hub model in terms of communication costs — when considering n agents, each agent communicates with the remaining $n-1$.

The P2P model is suitable for data owners that do not want to rely (or trust) upon a neutral agent. Potential business models could be: P2P forecasting between prosumers or RES power plants [19]; smart cities characterized by an increasing number of sensors and devices installed at houses, buildings, and transportation network [20].

In order to make these collaborative schemes feasible, the following fundamental principles must be respected: (a) ensure improvement in forecast skill, compared to a scenario without collaboration; (b) guarantee data privacy, i.e., agents and the central node cannot have access to (or recover) original data; (c) consider synchronous and asynchronous communication between agents. The formulation that will be described in Section IV fully guarantees these three core principles.

III. Background: Vector Autoregressive Model

This section summarizes the VAR model, as well as the most common model fitting algorithms. Throughout this paper, matrices are represented by bold uppercase letters, vectors by bold lowercase letters, and scalars by lowercase letters. Also, $\mathbf{a} = [a_1, a_2]^\top$ represents a column vector, while the column-wise operation between two vectors or matrices is denoted as $[\mathbf{a}, \mathbf{b}]$ or $[\mathbf{A}, \mathbf{B}]$, respectively.

A. VAR Model Formulation

Let $\{\mathbf{y}_t\}_{t=1}^T$ be an n-dimensional multivariate time series, where n is the number of data owners. Then, $\{\mathbf{y}_t\}_{t=1}^T$ follows a VAR model with p lags, denoted by $\text{VAR}_n(p)$, when

$$\mathbf{y}_t = \boldsymbol{\eta} + \sum_{\ell=1}^{p} \mathbf{y}_{t-\ell}\mathbf{B}^{(\ell)} + \boldsymbol{\varepsilon}_t, \tag{1}$$

for $t = 1, \ldots, T$, where $\boldsymbol{\eta} = [\eta_1, \ldots, \eta_n]$ is the constant intercept (row) vector, $\boldsymbol{\eta} \in \mathbb{R}^n$; $\mathbf{B}^{(\ell)}$ represents the coefficient matrix at lag $\ell = 1, ..., p$, $\mathbf{B}^{(\ell)} \in \mathbb{R}^{n \times n}$, and the coefficient associated with lag ℓ of time series i, to estimate time series j, is at position (i, j) of $\mathbf{B}^{(\ell)}$, for $i, j = 1, ..., n$; and $\boldsymbol{\varepsilon}_t = [\varepsilon_{1,t}, \ldots, \varepsilon_{n,t}]$, $\boldsymbol{\varepsilon}_t \in \mathbb{R}^n$, denotes a white noise vector that is independent and identically distributed with mean zero and nonsingular covariance matrix. By simplification, $\mathbf{y}_t$ is assumed to follow a centered process, $\boldsymbol{\eta} = \mathbf{0}$, i.e., as a vector of zeros of appropriate dimension. A VAR$_n(p)$ model can be written in matrix form as

$$\mathbf{Y} = \mathbf{ZB} + \mathbf{E}, \tag{2}$$

where

$$\mathbf{Y}=\begin{bmatrix} \mathbf{y}_1 \\ \ldots \\ \mathbf{y}_T \end{bmatrix}, \mathbf{B}=\begin{bmatrix} \mathbf{B}^{(1)} \\ \ldots \\ \mathbf{B}^{(p)} \end{bmatrix}, \mathbf{Z}=\begin{bmatrix} \mathbf{z}_1 \\ \ldots \\ \mathbf{z}_T \end{bmatrix}, \mathbf{E}=\begin{bmatrix} \boldsymbol{\varepsilon}_1 \\ \ldots \\ \boldsymbol{\varepsilon}_T \end{bmatrix},$$

are obtained by joining the vectors row-wise, and define, respectively, the $T \times n$ response matrix, the $np \times n$ coefficient matrix, the $T \times np$ covariate matrix and the $T \times n$ error matrix, with $\mathbf{z}_t = [\mathbf{y}_{t-1}, \ldots, \mathbf{y}_{t-p}]$.

B. VAR Model Estimation

Usually, when the number of covariates, np, is substantially smaller than the records, T, the VAR model is estimated through the multivariate least squares,

$$\hat{\mathbf{B}}_{\text{LS}} = \underset{\mathbf{B}}{\operatorname{argmin}} \left(\|\mathbf{Y} - \mathbf{ZB}\|_2^2 \right), \tag{3}$$

where $\|.\|_r$ represents both vector and matrix L_r norms. However, as the number of data owners increases, as well as the number of lags, it becomes indispensable to use regularization techniques, such as LASSO, aiming to introduce sparsity into the coefficient matrix estimated by the model. In the standard LASSO-VAR approach, the coefficients are estimated by

$$\hat{\mathbf{B}} = \underset{\mathbf{B}}{\operatorname{argmin}} \left(\frac{1}{2} \|\mathbf{Y} - \mathbf{ZB}\|_2^2 + \lambda \|\mathbf{B}\|_1 \right), \tag{4}$$

where $\lambda > 0$ is a scalar penalty parameter.

The LASSO penalty is convenient to use when handling high-dimensional data since the penalty function shrinks some of the coefficients to zero, performing variable selection. Instead of assuming that all lagged multivariate time series are contributing to the model, this framework extracts, with a small computational effort, the predictors with the strongest contribution to forecast the target variable. As showed in [6], [7], the introduction of sparsity in the model's coefficients can find sub-groups of spatio-temporal dependency between RES power plants, enabling the application of the LASSO-VAR model to a large spatial region. The outcome is an interpretable model, in terms of spatial and temporal dependency, which avoids noisy estimates and unstable forecasts. Some alternatives to LASSO are the partial spectral coherence with Bayesian information criterion [6] or a penalty term based on the correlation among the time series rate-of-change [21], among others.

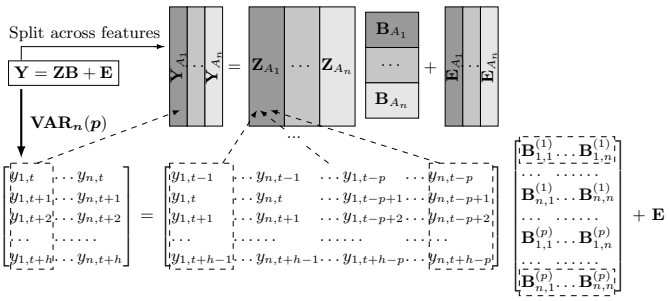


Fig. 1. Definition of VAR$_n(p)$ model and data structure.

Despite the many benefits, the LASSO regularization term makes the loss function in (4) non-differentiable, limiting the variety of optimization techniques that can be employed. In this domain, ADMM is a popular and computationally efficient technique allowing parallel estimation for data divided by records or features, which is an appealing property when designing a privacy-preserving approach.

1) Distributed ADMM and LASSO-VAR: When defining a VAR model, each time series is collected by a specific data owner, meaning that data are divided by features, i.e., $\mathbf{Y}=[\mathbf{Y}_{A_1}, \ldots, \mathbf{Y}_{A_n}]$ and $\mathbf{Z}=[\mathbf{Z}_{A_1}, \ldots, \mathbf{Z}_{A_n}]$, where $\mathbf{Y}_{A_i} \in \mathbb{R}^{T \times 1}$ and $\mathbf{Z}_{A_i} \in \mathbb{R}^{T \times p}$ denote the target and covariate matrix for the i-th data owner, respectively. Furthermore, $\mathbf{B}=[\mathbf{B}_{A_1}^{\mathsf{T}}, \ldots, \mathbf{B}_{A_n}^{\mathsf{T}}]^{\mathsf{T}}$, as illustrated in Fig. 1.

Consequently, the problem in (4) can be re-written as

$$\underset{\mathbf{B}}{\operatorname{argmin}} \left(\frac{1}{2} \|\mathbf{Y} - \sum_i \mathbf{Z}_{A_i} \mathbf{B}_{A_i}\|_2^2 + \lambda \sum_i \|\mathbf{B}_{A_i}\|_1 \right). \tag{5}$$

This decomposition of the loss function allows parallel computation of $\mathbf{B}_{A_i}$, being the ADMM solution provided by system of equations (6) – see [7],

$$\mathbf{B}_{A_i}^{k+1} = \underset{\mathbf{B}_{A_i}}{\operatorname{argmin}} \Big(\frac{\rho}{2} \|\mathbf{Z}_{A_i} \mathbf{B}_{A_i}^k + \overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k - \mathbf{U}^k - \mathbf{Z}_{A_i} \mathbf{B}_{A_i}\|_2^2 + \lambda \|\mathbf{B}_{A_i}\|_1 \Big), \tag{6a}$$

$$\overline{\mathbf{H}}^{k+1} = \frac{1}{N + \rho} \left(\mathbf{Y} + \rho \overline{\mathbf{ZB}}^{k+1} + \rho \mathbf{U}^k \right), \tag{6b}$$

$$\mathbf{U}^{k+1} = \mathbf{U}^k + \overline{\mathbf{ZB}}^{k+1} - \overline{\mathbf{H}}^{k+1}, \tag{6c}$$

where $\overline{\mathbf{ZB}}^{k+1} = \frac{1}{n} \sum_{j=1}^{n} \mathbf{Z}_{A_j} \mathbf{B}_{A_j}^{k+1}$, and $\mathbf{B}_{A_i}^{k+1} \in \mathbb{R}^{p \times n}$, $\mathbf{Z}_{A_i} \in \mathbb{R}^{T \times p}$, $\mathbf{Y}, \overline{\mathbf{H}}^k, \mathbf{U} \in \mathbb{R}^{T \times n}$, i=1, ..., n. $\mathbf{B}_{A_i}$ is estimated through standard ADMM, as described in Appendix A.

2) Privacy issues: In the collaboration schemes of Section II, each agent determines and transmits (6a) and then it is up to the central agent or peers (depending on the adopted structure) to compute the quantities in (6b) and (6c). Although there is no direct exchange of private data, the computation of (6b) and (6c) provides indirect information about these data, meaning that confidentiality breaches can occur after a number of iterations. The term "confidentiality breach" means the reconstruction of the entire private dataset by another party.

To reduce the possibility of such confidentiality breaches, recent work combined distributed ADMM with differential privacy, which consists of adding random noise (with certain

statistical properties) to the data itself or coefficients [22], [23]. However, these mechanisms can deteriorate the performance of the model even under moderate privacy guarantees [10].

IV. Privacy-preserving Distributed LASSO-VAR

This section describes the novel privacy-preserving collaborative forecasting method, which combines multiplicative randomization of the data (Section IV-A) with the distributed ADMM for the generalized LASSO-VAR model (Section IV-B). Communication issues (Section IV-E) are also addressed since they are common in distributed systems.

A. Data Transformation with Multiplicative Randomization

Multiplicative randomization of the data [24] consists of multiplying the data matrix $\mathbf{X} \in \mathbb{R}^{T\times ns}$ by full rank perturbation matrices. If the perturbation matrix $\mathbf{M} \in \mathbb{R}^{T\times T}$ pre-multiplies $\mathbf{X}$, i.e., $\mathbf{MX}$, the records are randomized. On the other hand, if perturbation matrix $\mathbf{Q} \in \mathbb{R}^{ns\times ns}$ post-multiplies $\mathbf{X}$, i.e., $\mathbf{XQ}$, then the features are randomized. The challenges related to such transformations are two-fold: (i) $\mathbf{M}$ and $\mathbf{Q}$ are algebraic encryption keys, and consequently should be fully unknown by agents, (ii) data transformations need to preserve the relationship between the original time series.

When $\mathbf{X}$ is divided by features, as is the case with matrices $\mathbf{Z}$ and $\mathbf{Y}$ when defining VAR models, $\mathbf{Q}$ can be constructed as a diagonal matrix – see (7), where matrices in diagonal, $\mathbf{Q}_{A_i} \in \mathbb{R}^{s\times s}$, are privately defined by agent $i = 1,\dots,n$. Then, agents post-multiply their data without sharing $\mathbf{Q}_{A_i}$, since

$$\underbrace{\left[\mathbf{X}_{A_1},\dots,\mathbf{X}_{A_n}\right]}_{=\mathbf{X}} \underbrace{\begin{bmatrix}\mathbf{Q}_{A_1} & & \mathbf{0}\\ & \ddots & \\ \mathbf{0} & & \mathbf{Q}_{A_n}\end{bmatrix}}_{=\mathbf{Q}} = \left[\mathbf{X}_{A_1}\mathbf{Q}_{A_1},\dots,\mathbf{X}_{A_n}\mathbf{Q}_{A_n}\right]. \tag{7}$$

Unfortunately, the same reasoning is not possible when defining $\mathbf{M}$, because all elements of column j of $\mathbf{M}$ multiplies all elements of row j in $\mathbf{X}$ (containing data from every agent). Therefore, the challenge is to define a random matrix $\mathbf{M}$, unknown but at the same time built by all agents.

We propose to define $\mathbf{M}$ as

$$\mathbf{M} = \mathbf{M}_{A_1}\mathbf{M}_{A_2}\dots\mathbf{M}_{A_n}, \tag{8}$$

where $\mathbf{M}_{A_i} \in \mathbb{R}^{T\times T}$ is privately defined by agent i. This means that

$$\mathbf{MX} = [\underbrace{\mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_1}}_{=\mathbf{MX}_{A_1}},\dots,\underbrace{\mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_n}}_{=\mathbf{MX}_{A_n}}]. \tag{9}$$

Some linear algebra-based protocols exist for secure matricial product, but they were designed for matrices with independent observations and have proven to fail when applied to such matrices as $\mathbf{Z}$ and $\mathbf{Y}$(see [10] for a proof). The calculation of $\mathbf{MX}_{A_i}$ is described in Algorithm 1.

The privacy of this protocol depends on r, which is chosen according to the number of unique values on $\mathbf{X}_{A_i}$. The optimal value for r is discussed in Proposition 1 of Appendix B.

Algorithm 1 Data Encryption.

Input from ith agent: $\mathbf{X}_{A_i} \in \mathbb{R}^{T\times s}$ and $\mathbf{M}_{A_i} \in \mathbb{R}^{T\times T}$
Input from jth agent ($j \neq i$): $\mathbf{M}_{A_j} \in \mathbb{R}^{T\times T}$
Output: $\mathbf{MX}_{A_i} = \mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_i}$

1: **Initialization:** Agent i generates random invertible matrices $\mathbf{C}_{A_i} \in \mathbb{R}^{T\times(r-s)}$, $\mathbf{D}_{A_i} \in \mathbb{R}^{r\times r}$, and shares $\mathbf{W}_{A_i} \in \mathbb{R}^{T\times r}$ with the n-th agent,

$$\mathbf{W}_{A_i} = [\mathbf{X}_{A_i}, \mathbf{C}_{A_i}]\mathbf{D}_{A_i}. \tag{10}$$

2: Agent n receives $\mathbf{W}_{A_i}, \forall i$.
3: Agent n shares $\mathbf{M}_{A_n}\mathbf{W}_{A_i}$ with the $(n-1)$-th agent.
4: **for** agent $j = n-1,\dots,1$ **do**
5: Agent j receives $\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$, and
6: **if** $j > 1$ **then**
7: shares $\mathbf{M}_{A_j}\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$ with agent $j-1$
8: **else**
9: shares $\mathbf{M}_{A_j}\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$ with agent i
10: **end if**
11: **end for**
12: Agent i receives $\mathbf{MW}_{A_i}$ from the 1-st agent and recovers $\mathbf{MX}_{A_i}$,

$$[\mathbf{MX}_{A_i}, \mathbf{MC}_{A_i}] = \mathbf{MW}_{A_i}\mathbf{D}_{A_i}^{-1}. \tag{11}$$

B. Formulation of the Collaborative Forecasting Model

When applying the ADMM algorithm, the protocol presented in the previous section should be applied to transform matrices $\mathbf{Z}$ and $\mathbf{Y}$ in such a way that: (i) the estimated coefficients do not coincide with the originals, instead they are a secret transformation of them, (ii) agents are unable to recover the private data through the exchanged information, and (iii) cross-correlations cannot be obtained, i.e., agents are unable to recover $\mathbf{Z}^\top\mathbf{Z}$ nor $\mathbf{Y}^\top\mathbf{Y}$.

To fulfill these requirements, both covariate and target matrices are transformed through multiplicative noise. Both $\mathbf{M}$ and $\mathbf{Q}$ must be invertible, which is ensured if $\mathbf{M}_{A_i}$ and $\mathbf{Q}_{A_i}$ are invertible for $i = 1,\dots,n$.

1) Formulation: Let $\mathbf{ZQ}$ be the covariate matrix obtained through (7) and $\mathbf{Y}$ the target matrix. Covariate matrix $\mathbf{ZQ}$ is divided by features, and the optimization problem which allows recovering the solution of (5) is

$$\underset{\mathbf{B}^{\text{post}}}{\operatorname{argmin}}\left(\frac{1}{2}\|\mathbf{Y}-\sum_i \mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}_{A_i}^{\text{post}}\|_2^2+\lambda\sum_i\|\mathbf{Q}_{A_i}\mathbf{B}_{A_i}^{\text{post}}\|_1\right). \tag{12}$$

After a litle algebra, the relation between the ADMM solution for (5) and (12) is

$$\mathbf{B}_{A_i}^{\text{post}\,k+1} = \mathbf{Q}_{A_i}\mathbf{B}_{A_i}^{k+1}, \tag{13}$$

suggesting coefficients privacy since the original $\mathbf{B}$ is no longer used. However, the limitations identified in a previous work [10] for (5) are valid for (12). That is, a curious agent can obtain both $\mathbf{Y}$ and $\mathbf{ZQ}$, and because $\mathbf{Y}$ and $\mathbf{Z}$ share a large proportion of values, $\mathbf{Z}$ can also be recovered.

Taking covariate matrix $\mathbf{MZQ}$ and target $\mathbf{MY}$, the ADMM solution for the optimization problem

$$\underset{\mathbf{B}'}{\operatorname{argmin}} \left(\frac{1}{2}\|\mathbf{MY}-\sum_i \mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'_{A_i}\|_2^2+\lambda\sum_i \|\mathbf{Q}_{A_i}\mathbf{B}'_{A_i}\|_1\right), \tag{14}$$

preserves the relation between the original time series if $\mathbf{M}$ is orthogonal, i.e., $\mathbf{MM}^\top=\mathbf{I}$. In this case, a curious competitor can only obtain $\mathbf{MY}$ without distinguishing between $\mathbf{M}$ and $\mathbf{Y}$. But the orthogonality of $\mathbf{M}$ ensures that $(\mathbf{MY})^\top\mathbf{MY} = \mathbf{Y}^\top\mathbf{Y}$, meaning that the covariance matrix is not protected.

Note that the orthogonality of $\mathbf{M}$ is necessary to ensure that, while computing $\mathbf{B}'_{A_i}$,

$$\begin{aligned}\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^\top\left[\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^k_{A_i} - \overline{\mathbf{MZQB}'}^k + \dots\right] = \\ \mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\left[\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^k_{A_i} - \overline{\mathbf{ZQB}'}^k + \dots\right].\end{aligned} \tag{15}$$

We deal with this limitation by using $\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}$ instead of $\mathbf{Z}_{A_i}^\top\mathbf{M}^\top$. Our proposal requires agents to compute $\mathbf{MZ}_{A_i}$, $\mathbf{MY}_{A_i}$ and $\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}$. Algorithm 2 summarizes our proposal for estimating a privacy-preserving LASSO-VAR model.

$\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}$ is obtained by adapting the protocol in (10)–(11). In this case, the value of r is more restrictive because we need to ensure that agent i does not obtain both $\mathbf{Y}_{A_i}^\top\mathbf{M}^{-1}$ and $\mathbf{MY}_{A_i}$. Otherwise, the covariance and cross-correlation matrices are again vulnerable. Let us assume that $\mathbf{Z}_{A_i}$ has u unique unknown values and $\mathbf{Y}_{A_i}$ has v unique unknown values that are not in $\mathbf{Z}_{A_i}$. Then, privacy is ensured by computing $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}$ and $\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}$ using the smaller integer r such that $\sqrt{Tp-u}<r<T/2 \wedge r > p$, and then $\mathbf{MY}_{A_i}$ with $\sqrt{T-v}<r'<T-2r \wedge r' > 1$ (see Proposition 2 in Appendix B for determination of the optimal r). Appendix C presents an analysis of the data privacy for scenarios without and with collusion between agents (data owners) during encrypted data exchange.

Finally, it is important to underline that Algorithm 2 can be applied to both *central hub model* and *P2P model* schemes without any modification – depending on who (central node or peers, respectively) receives $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{k+1}_{A_i}$ and computes (17)–(19).

2) Malicious agents: The proposed approach assumes that agents should only trust themselves, requiring control mechanisms to detect when agents share wrong estimates of their coefficients, compromising the global model. Since $\mathbf{MY}$ and $\mathbf{MZQB}'^k$ can be known by agents without exposing private data, a malicious agent is detected through the analysis of the global error $\|\mathbf{MY}-\mathbf{MZQB}'^k\|_2^2$. That is, during the iterative process, this global error should smoothly converge, as depicted in Fig. 2 (left plot), and the same is expected for the individual errors $\|\mathbf{MY}-\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^k_{A_i}\|_2^2, \forall i$.

In the example of Fig. 2, two agents are assumed to add random noise to their coefficients. This results in the erratic curve for the global error shown in Fig. 2. An analysis of individual errors, in Fig. 2 (right plot), shows that all agents have smooth curves, except the two who shared distorted information.

Algorithm 2 Synchronous Privacy-preserving LASSO-VAR.

Input: Randomized data $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}$, $\mathbf{MY}_{A_i}$, $\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}$
Output: Transformed coefficients $\mathbf{B}'_{A_i}=\mathbf{Q}_{A_i}\mathbf{B}_{A_i}, i=1,\dots,n$

Initialization: $\mathbf{B}'^0_{A_i}$, $\overline{\mathbf{H}}^0$, $\mathbf{U}^0=\mathbf{0}, \rho\in\mathbb{R}^+, k=0$
for agent $i=1,\dots,n$ **do**
  $\mathbf{P}_{A_i} = \left((\mathbf{Z}_{A_i}\mathbf{Q}_{A_i})^\top(\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}) + \rho\mathbf{Q}_{A_i}^\top\mathbf{Q}_{A_i}\right)^{-1}$
end for
while stopping criteria not satisfied **do**
  for agent $i=1,\dots,n$ **do**
    Initialization: $\widetilde{\mathbf{B}}^0_{A_i}$, $\overline{\widetilde{\mathbf{H}}}^0$, $\widetilde{\mathbf{U}}^0=\mathbf{0}, j=0$
    $\mathbf{K}_{A_i}=\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^k_{A_i}+\overline{\mathbf{H}}^k-\overline{\mathbf{MZQB}'}^k-\mathbf{U}^k$ (16)
    while stopping criteria not satisfied **do**
      $\widetilde{\mathbf{B}}^{j+1}_{A_i} = \mathbf{P}_{A_i}\left(\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}\mathbf{K}_{A_i}+\rho(\overline{\widetilde{\mathbf{H}}}^j-\widetilde{\mathbf{U}}^j)\right)$
      $\overline{\widetilde{\mathbf{H}}}^{j+1} = S\left(\mathbf{Q}_{A_i}\widetilde{\mathbf{B}}^{j+1}_{A_i}+\widetilde{\mathbf{U}}^j\right)$
      $\widetilde{\mathbf{U}}^{j+1} = \widetilde{\mathbf{U}}^j + \mathbf{Q}_{A_i}\widetilde{\mathbf{B}}^{j+1}_{A_i} - \overline{\widetilde{\mathbf{H}}}^{j+1}$
      $j=j+1$
    end while
    $\mathbf{B}'^{k+1}_{A_i} = \widetilde{\mathbf{B}}^j_{A_i}$
  end for
  $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^k_{A_i}$ is shared with peers or central node, who computes (17)–(19),
  $\overline{\mathbf{MZQB}'}^k = \frac{1}{N}\sum_i \mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^k_{A_i}$ (17)
  $\overline{\mathbf{H}}^{k+1} = \frac{1}{N+\rho}\left(\mathbf{MY}+\overline{\mathbf{MZQB}'}^k+\rho\mathbf{U}^k\right)$ (18)
  $\mathbf{U}^{k+1} = \mathbf{U}^k + \overline{\mathbf{MZQB}'}^{k+1} - \overline{\mathbf{H}}^{k+1}$ (19)
  $k=k+1$
end while

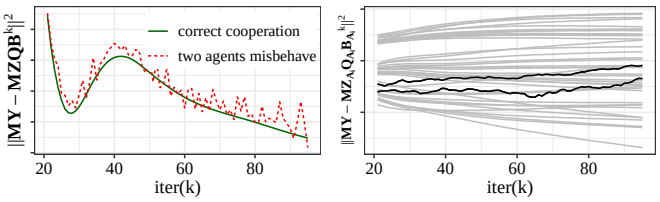


Fig. 2. Error evolution (left: global error; right: error by agent with black lines representing the two agents who add random noise to $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^k_{A_i}$).

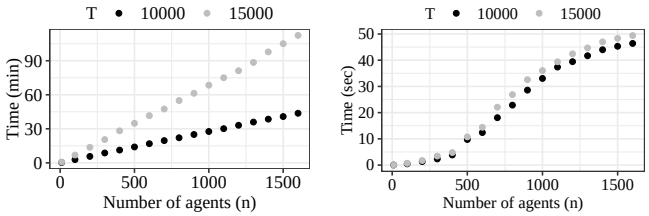
C. Tuning of Hyper-parameters

Since the ADMM solutions for (4) and (14) are the same, agents can tune hyper-parameters (ρ and λ) by applying common techniques, such as cross-validation grid-search, Nelder-Mead optimization, Bayesian optimization, etc., to minimize the loss function in (14). This requires the definition of fitting and validation datasets and corresponding encryption by Algorithm 1, taking into account that, for each fitting and validation pair, the matrix $\mathbf{Q}_{A_i}$ needs to be the same, but all the others should be changed to keep data private.

TABLE I
FLOATING-POINT OPERATIONS IN ALGORITHM 1.

Encrypted information	Operations
$(\mathbf{M}\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}, \mathbf{Q}_{A_i}^\top \mathbf{Z}_{A_i}^\top \mathbf{M}^{-1})$	$\mathcal{O}(2Tr^2 + 2T^2nr + T(p^2 + r^2))$
$\mathbf{M}\mathbf{Y}_{A_i}$	$\mathcal{O}(Tr'^2 + T^2nr' + Tr'^2)$

* $r = \max(\lceil\sqrt{Tp - u}\rceil, p + 1)$ and $r' = \max(\lceil\sqrt{T - v}\rceil, 1)$



(a) Data encryption by Algorithm 1. (b) ADMM (Algorithm 2) iteration.

Fig. 3. Mean running time as a function of the number of agents.

D. Computational Complexity

Typically, the computational complexity of an algorithm is estimated by the number of required floating-point operations (defined as one addition, subtraction, multiplication, or division of two floating-point numbers). When compared to the existing distributed ADMM literature applied to the LASSO-VAR model (e.g., [7], [25]), the computational complexity of the ADMM algorithm remains almost the same – only p^2n extra floating-point operations come from considering $\mathbf{Q}_{A_i}\widetilde{\mathbf{B}}_{A_i}^{j+1}$ instead of $\widetilde{\mathbf{B}}_{A_i}^{j+1}$ in line 11 and 12 of Algorithm 2. However, there is also the computational cost related to the data transformation, performed before running the ADMM algorithm. Table I summarizes the floating-point operations necessary to encrypt the data matrices $\mathbf{Z}_{A_i}$ and $\mathbf{Y}_{A_i}$. The computational time for such data encryption is expected to increase linearly with the number of agents, and quadratically with the number of records. A numerical analysis was performed by simulating data from VAR models with $n \in \{10, 100, 200, \ldots, 1600\}, T \in \{10000, 15000\}$ and $p = 5$.

Fig. 3 summarizes the mean running times using an i7-8750H @ 2.20GHz with 16 GB of RAM. To properly analyze the mean time per ADMM iteration, the computational times for the cycle between lines 6 to 15 of Algorithm 2 (coefficients' update) is measured assuming that the n agents update it in parallel. That said, considering for example a case with 10000 records and 500 agents, the data encryption takes around 15 minutes, and then the Algorithm 2 takes around 10 seconds per iteration.

E. Asynchronous Communication

When applying the proposed method, the matrices (17)–(19) combine the solutions of all data owners, meaning that the "slowest" agent dictates the duration of each iteration. Since communication delays and failures may occur due to computation or communication issues, the proposed algorithm should be robust to this scenario. Otherwise, the convergence to the optimal solution may require too much time. The proposed approach deals with these issues by considering the last information sent by agents, but different strategies are followed according to the adopted collaborative scheme.

Regarding the centralized scheme, let Ω_i^k be the set of iterations for which agent i communicated its information, until current iteration k. After receiving the local contributions, central agent computes $\overline{\mathbf{H}}^k$ and $\mathbf{U}^k$, in (18)–(19), by using $\sum_{i=1}^n \mathbf{M}\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{\max(\Omega_i^k)}_{A_i}$. Then, central agent returns $\overline{\mathbf{H}}^k$ and $\mathbf{U}^k$, informing agents about $\max(\Omega_i^k)$. To proceed, $\mathbf{B}'^{k+1}_{A_i}$ is updated by using $\mathbf{M}\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{\max(\Omega_i^k)}_{A_i}$ in (16).

For the P2P approach, let Λ_i^k be the set of agents sharing information computed at iteration k, with agent i, i.e., $\Lambda_i^k = \{j : \text{agent } j \text{ sent } \mathbf{M}\mathbf{Z}_{A_j}\mathbf{Q}_{A_j}\mathbf{B}'^k_{A_j} \text{ to agent } i\}$. After computing and sharing $\mathbf{M}\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^k_{A_i}$, a second round of peer-to-peer communication is proposed, where agents share both Λ_i^k and $\sum_{j\in\Lambda_i^k}\mathbf{M}\mathbf{Z}_{A_j}\mathbf{Q}_{A_j}\mathbf{B}'^k_{A_j}$. After this extra communication round, agent i can obtain missing information when $\Lambda_i^k \neq \Lambda_j^k$, $\forall i, j$.

V. CASE-STUDY

A. Data Description and Experimental Setup

The proposed algorithm is applied to forecast solar power up to 6 hours ahead. The data is publicly available in [10] and consists of hourly time series of solar power from 44 micro-generation units, located in a Portuguese city, and covers the period from February 1, 2011 to March 6, 2013. Since the VAR model requires the data to be stationary, the solar power is normalized through a clear sky model, which gives an estimate of the solar power in clear sky conditions at any given time [26]. This clear-sky model is fully data-driven and does not require any site-specific information (coordinates, rated power, etc.) since it estimates the clear-sky power time series exclusively from historical on-site power observations. Also, night-time hours are excluded by removing data for which the solar zenith angle is larger than 90. Based on previous work [3], a LASSO-VAR model to forecast $y_{i,t+h}$ at time t (using lags $t-1$, $t-2$ and $t+h-23$) is evaluated with a sliding-window of one month and the model's fitting period consists of 12 months, $h \leq 6$.

It is important to note that the LASSO-VAR model can be applied to both solar and wind power time series without any modification (see [7] for wind power forecasting). Nevertheless, a different set of lags should be selected for wind power. Furthermore, when compared to wind power, solar power forecasting is more challenging because the lags 1 and 2 are zero for the first daylight hours, i.e., there are fewer unknown data, and this makes it easier to recover original data. In our protocol, this means more restrictive values for u and v, which are crucial when defining r and r', as stated in Proposition 2.

To simulate the proposal, communication failures are modeled through Bernoulli random variables F_{it}, with failure probability p_i, $F_{it} \sim Bern(p_i)$, for each agent $i=1,\ldots,n$ at each communication time t.

The performance of the models is accessed through the normalized root mean squared error (NRMSE) calculated for agent i and lead-time h, with h=1, . . . , 6, as

$$\text{NRMSE}_{i,h} = \frac{\sqrt{\frac{1}{k}\sum_{t=1}^{T}(\hat{\mathbf{y}}_{i,t+h} - \mathbf{y}_{i,t+h})^2}}{(\sum_{t=1}^{T}\mathbf{y}_{i,t+h})/T}, \quad (20)$$

where $\hat{\mathbf{y}}_{i,t+h}$ represents the forecast generated at time t.

The ADMM process stops when all agents achieve $\|\mathbf{B}_{A_i}^{k+1} - \mathbf{B}_{A_i}^{k}\|_2 / \max(1, \min(\|\mathbf{B}_{A_i}^{k+1}\|_1, \|\mathbf{B}_{A_i}^{k}\|_1)) \leq \epsilon$, where ϵ is the tolerance parameter.

B. Benchmark Models

The persistence and LASSO-autoregressive (LASSO-AR) models are implemented to assess the impact of collaboration over a model without collaboration. Two persistence models are considered: $\hat{y}_{i,t+h} = \hat{y}_{i,t}$ (last measured power) and $\hat{y}_{i,t+h} = \hat{y}_{i,t+h-23}$ (power measured 24 hours before).

The analog method described in [12] was also implemented as a benchmark model because: (a) it is the only work in the RES forecasting literature that implements collaborative forecasting without data disclosure; (b) when the forecasting algorithm was designed, a trade-off between accuracy and privacy was necessary and the choice was privacy over accuracy.

Firstly, agent i searches the k situations most similar to the current power production values $\mathbf{y}_{i,t-\ell+1}, \ldots, \mathbf{y}_{i,t}$. This similarity is measured through the Euclidean distance. Secondly, the k most similar situations (called analogs) are weighted according to the corresponding Euclidean distance. Agent i attributes the weight $w_{A_i}(a)$ to the analog a. The forecast for h steps ahead is obtained by applying the computed weights on the h values registered immediately after the k analogs. The collaboration between agents requires the exchange of the time indexes for the selected analogs and corresponding weights. Two analogs belong to the same global situation if they occur at the same or at close timestamps. Agent i scores the analog a, observed at timestamps t_a, by performing

$$s_{A_i}(a) = \underbrace{(1-\alpha) w_{A_i}(a)}_{\text{own contribution}} + \underbrace{\frac{\alpha}{n}\sum_{i=1}^{n}\sum_{j=1}^{k} w_{A_j}(j) I_\epsilon(t_a, t_j)}_{\text{others' weights for close timestamps}}, \quad (21)$$

where α is the weight given to neighbor information, j are the analogs from other agents, registered at timestamps t_j, and $I_\epsilon(t_a, t_j)$ is the indicator function taking value 1 if $|t_j - t_a| \leq \epsilon$, with ϵ being the maximum time difference for two analogs to be considered part of the same global situation.

The results in the next section will show that our approach does not degrade accuracy (the same results of a LASSO-VAR without privacy constraints are obtained), while offering robustness to data privacy.

C. Numerical Results

To access the quality of the proposed collaborative forecasting model, the synchronous LASSO-VAR is compared with benchmark models. Both *central hub* and *P2P model* have the same accuracy when considering synchronous communication.

The hyper-parameters ρ and λ were determined by cross-validation (12 folds) in the initial model's fitting dataset, by considering the values of $\rho, \lambda \in \{0.5, 1, 2, 3, 4, 5, 10, 15, 20, 25\}$. Figure 4 illustrates the results in terms of NRMSE, for $h = 1$.

Table II presents the NRMSE error for all agents, distinguishing between lead-times. In general, the smaller the forecasting horizon, the larger is the NRMSE improvement, i.e., $(NRMSE_{Bench.} - NRMSE_{VAR})/NRMSE_{Bench.} \cdot 100\%$. Besides, since the proposed LASSO-VAR and the LASSO-AR models have similar NRMSE for $h > 3$, the Diebold-Mariano test [27] is applied to test the superiority of the proposal, assuming a confidence level of 5%. This test showed that the improvement is statistically significant for all horizons. It is important to note that the decrease in the improvement is explained by the cross-correlation between the geographically distributed time series data. Since the dataset is from a small municipality in Portugal, it is expected that the highest improvement occurs for the first lead times (in particular the first one), where the cross-dependencies between time series have the most effect. However, this depends on the geographical layout and distance between power plants. For instance, in [7], the results for wind power plants show the highest improvement for the second lead time; in the test case of western Denmark [28], the highest cross-dependency between two groups of wind farms was observed for lag two.

Fig. 5 depicts the relative improvement in terms of NRMSE for the 44 agents. According to the Diebold-Mariano test, the LASSO-VAR model outperforms benchmarks in all lead-times for at least 25 of the 44 agents. Indeed, some agents contribute to improving the competitors' forecast without having a benefit to their own forecasting accuracy. Then, even if privacy is ensured, such agents can be unwilling to collaborate, which motivates data monetization through data markets [29].

For asynchronous communication, equal failure probabilities p_i are assumed for all agents. Since a specific p_i can generate various distinct failure sequences, 20 simulations were performed for each p_i, $p_i \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. Table III shows the mean NRMSE improvement for different failure probabilities $p_i, i = 1, \ldots, n$. In general, the greater the p_i the smaller the improvement. Despite the model's accuracy decreases slightly, the LASSO-VAR model continues to outperform the AR model for both collaborative schemes, which demonstrates high robustness to communication failures.

Fig. 6 depicts the evolution of the loss while fitting the LASSO-VAR model, considering $p_i \in \{0.5, 0.9\}$. For the centralized approach, the loss tends to stabilize around larger values. In general, the results are better for the P2P scheme since in the centralized approach if an agent fails the algorithm proceeds with no chance of obtaining its information. In P2P, this agent may have communicated his contribution to some peers and the probability of losing information is smaller.

Finally, Table IV presents the mean running times and the number of iterations of both non-distributed and distributed approaches. The proposed schemes require larger execution times since they require estimating $\mathbf{B'}_{A_i}^{k}$ through a second ADMM cycle (Algorithm 2). However, the non-distributed LASSO-VAR requires more iterations to converge (ϵ=5×10^{-4}).

TABLE II
NRMSE FOR SYNCHRONOUS MODELS.

	h=1	h=2	h=3	h=4	h=5	h=6
Persistence (t)*	0.4054	0.7049	0.9511	1.1385	1.2671	1.3445
Persistence ($t+h$-23)*	0.4362	0.4362	0.4362	0.4362	0.4362	0.4362
Analogs [12]†	0.2659	0.3319	0.3753	0.4011	0.4139	0.4191
LASSO-AR*	0.2580	0.3359	0.3641	0.3757	0.3798	0.3815
LASSO-VAR†	**0.2363**✓	**0.3155**✓	**0.3533**✓	**0.3699**✓	**0.3745**✓	**0.3780**✓

∗ non-collaborative † collaborative ✓ statistically significant improvement

TABLE III
MEAN RELATIVE NRMSE IMPROVEMENT [%] OVER THE LASSO-AR MODEL.

	h=1		h=2		h=3		h=4		h=5		h=6	
p_i	central	P2P	central	P2P	central	P2P	central	P2P	central	P2P	central	P2P
0	8.41		6.05		2.95		1.52		1.39		0.93	
0.1	7.93	8.41	5.98	6.05	2.91	2.95	1.49	1.52	1.35	1.39	0.89	0.93
0.3	7.45	”	5.89	”	2.89	”	1.40	”	1.18	”	0.69	”
0.5	6.69	”	5.77	”	2.88	”	1.30	”	1.00	”	0.52	”
0.7	5.71	”	5.54	”	2.84	”	1.24	”	0.89	”	0.33	”
0.9	3.75	8.10	5.19	5.75	2.74	2.78	0.75	1.47	0.62	1.38	-0.82	0.88

TABLE IV
MEAN RUNNING TIMES (IN SEC) PER ITERATION AND NUMBER OF ITERATIONS UNTIL CONVERGENCE.

Non distributed LASSO-VAR	Central LASSO-VAR		P2P LASSO-VAR	
	Enc. data	ADMM	Enc. data	ADMM
0.035 (≈ 410)	65.46	0.052 (≈ 300)	65.46	0.1181 (≈ 300)

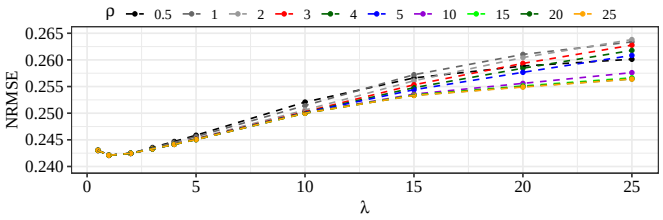


Fig. 4. Impact of hyper-parameters for $h = 1$.

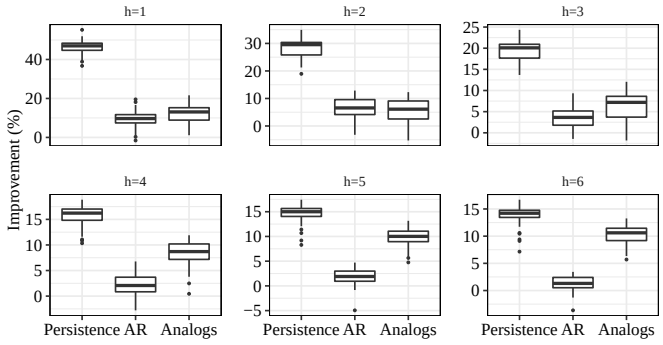


Fig. 5. Relative NRMSE improvement [%] over the baseline models.

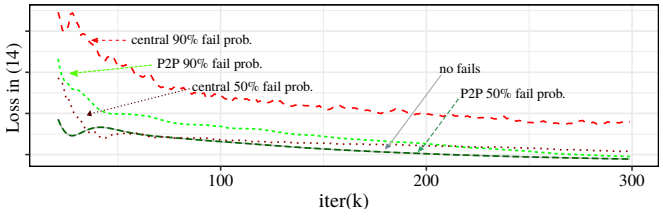


Fig. 6. Loss while fitting LASSO-VAR model.

VI. CONCLUSION

RES forecast skill can be improved by combining data from multiple geographical locations. One of the simplest and most effective collaborative models for very short-term forecasts is the vector autoregressive model. However, different data owners might be unwilling to share their time series data. In order to ensure data privacy, this work combined the advantages of the ADMM decomposition method with data encryption through linear transformations of data. It is important to underlines that the coefficients matrix obtained with the privacy-preserving protocol is the same one obtained without any privacy protection.

This novel method also included an asynchronous distributed ADMM algorithm, making it possible to update the forecast model based on information from a subset of agents and improve the computational efficiency of the proposed model. The mathematical formulation is flexible enough to be applied in two different collaboration schemes (central hub model and P2P) and paved the way for learning models distributed by features, instead of observations.

The results obtained for a solar energy dataset show that the privacy-preserving LASSO-VAR model delivers a forecast skill comparable to a model without privacy protection and outperformed a state-of-the-art method based on analog search. Furthermore, it exhibited high robustness to communication failures, in particular for the P2P scheme.

Two aspects not addressed in this paper were uncertainty forecasting and application to non-linear models (and consequently longer lead times), which we plan to investigate in a forthcoming work. Nevertheless, uncertainty forecast can be readily generated by transforming original data using a logit-normal distribution [6]. The proposed privacy-preserving protocol can be applied to non-linear regression by extending the additive model structure to a multivariate setting [30] or by local linear smoothing [31]. However, the extension to other non-linear multivariate models, such as long short-term memory networks and variants, requires further researcher and significant changes in the protocol. For instance, the rectifier (ReLU), which is an activation function commonly applied in neural networks and defined as $f(x) = \max(0, x)$, has the problem that $f(\mathbf{MZQB}) \neq \mathbf{M}f(\mathbf{ZQB})$.

APPENDIX A
STANDARD ADMM AND LASSO-VAR

The ADMM solution for (4) is obtained by splitting the $\mathbf{B}$ variable into two variables ($\mathbf{B}$ and $\mathbf{H}$) and adding the constraint $\mathbf{H} = \mathbf{B}$,

$$\underset{\mathbf{B}}{\operatorname{argmin}} \left(\frac{1}{2} \|\mathbf{Y} - \mathbf{ZB}\|_2^2 + \lambda \|\mathbf{H}\|_1 \right) \text{ subject to } \mathbf{H} = \mathbf{B}. \quad (22)$$

Then, based on the augmented Lagrangian of (22), the solution is provided by the following system of equations – see [7],

$$\mathbf{B}^{k+1} = (\mathbf{Z}^\top \mathbf{Z} + \rho \mathbf{I})^{-1} (\mathbf{Z}^\top \mathbf{Y} + \rho(\overline{\mathbf{H}}^k - \mathbf{U}^k)) \quad (23a)$$

$$\mathbf{H}^{k+1} = S_{\lambda/\rho}(\mathbf{B}^{k+1} + \mathbf{U}^k) \quad (23b)$$

$$\mathbf{U}^{k+1} = \mathbf{U}^k + \mathbf{B}^{k+1} - \mathbf{H}^{k+1}, \quad (23c)$$

where $\mathbf{U}$ is the scaled dual variable associated with the constraint $\mathbf{H}=\mathbf{B}$, $\mathbf{I}$ is the identity matrix with proper dimension, and $S_{\lambda/\rho}$ is the soft thresholding operator. Consequently, (6a) can be estimated by adapting (22),

$$\underset{\mathbf{B}}{\operatorname{argmin}} \left(\frac{1}{2}\|\hat{\mathbf{Y}}-\mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2+\hat{\lambda}\|\mathbf{H}_{A_i}\|_1\right) \text{ s.t. } \mathbf{H}_{A_i}=\mathbf{B}_{A_i}, \tag{24}$$

where $\hat{\mathbf{Y}}_{A_i}=\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^k+\overline{\mathbf{H}}^k-\overline{\mathbf{Z}\mathbf{B}}^k-\mathbf{U}^k$ and $\hat{\lambda}=\lambda/\rho$.

APPENDIX B
OPTIMAL VALUE OF r

Proposition 1. Let $\mathbf{X}_{A_i}\in\mathbb{R}^{T\times s}$ be the sensible data from agent i, with u unique values, and $\mathbf{M}_{A_j}\in\mathbb{R}^{T\times T}$ be the private encryption matrix from agent j. If agents compute $\mathbf{M}_{A_j}\mathbf{X}_{A_i}$ applying the protocol in (10)–(11), then two invertible matrices $\mathbf{D}_{A_i}\in\mathbb{R}^{r\times r}$ and $\mathbf{C}_{A_i}\in\mathbb{R}^{T\times(r-s)}$ are generated by agent i and data privacy is ensured for

$$\sqrt{Ts-u}<r<T. \tag{25}$$

Proof. Since agent i only receives $\mathbf{M}_{A_j}[\mathbf{X}_{A_i}\mathbf{C}_{A_i}]\mathbf{D}_{A_i}\in\mathbb{R}^{T\times r}$, the matrix $\mathbf{M}_{A_j}\in\mathbb{R}^{T\times T}$ is protected if $r<T$. Furthermore, agent j receives $[\mathbf{X}_{A_i}\mathbf{C}_{A_i}]\mathbf{D}_{A_i}\in\mathbb{R}^{T\times r}$ and does not know $\mathbf{X}_{A_i}\in\mathbb{R}^{T\times s}, \mathbf{C}_{A_i}\in\mathbb{R}^{T\times r-s}$ and $\mathbf{D}_{A_i}\in\mathbb{R}^{r\times r}$. Although $\mathbf{X}_{A_i}\in\mathbb{R}^{T\times s}$, we assume this matrix has u unique values whose positions are known by all agents – when defining a VAR model with p consecutive lags $\mathbf{Z}_{A_i}$ has $T+p-1$ unique values, see Fig. 1 – meaning there are fewer values to recover.

Given that, agent j receives Tr values and wants to determine $u+T(r-s)+r^2$. The solution of the inequality $Tr<u+T(r-s)+r^2$, in r, determines that data from agent i is protected when $r>\sqrt{Ts-u}$. ■

Proposition 2. Let $\mathbf{X}_{A_i}\in\mathbb{R}^{T\times s}$ and $\mathbf{G}_{A_i}\in\mathbb{R}^{T\times g}$ be private data matrices, such that $\mathbf{X}_{A_i}$ has u unique values to recover and $\mathbf{G}_{A_i}$ has v unique values that are not in $\mathbf{X}_{A_i}$. Assume the protocol in (10)–(11) is applied to compute $\mathbf{M}\mathbf{X}_{A_i}$, $\mathbf{X}_{A_i}^\top\mathbf{M}^{-1}$ and $\mathbf{M}\mathbf{G}_{A_i}$, with $\mathbf{M}$ as defined in (8). Then, to ensure privacy while computing $\mathbf{M}\mathbf{X}_{A_i}$ and $\mathbf{X}_{A_i}^\top\mathbf{M}^{-1}$, the protocol requires

$$\sqrt{Ts-u}<r<T/2\wedge r>s. \tag{26}$$

In addition, to compute $\mathbf{M}\mathbf{G}_{A_i}$, the protocol should take

$$\sqrt{Tg-v}<r'<T-2r\wedge r'>g. \tag{27}$$

Proof. *(i)* To compute $\mathbf{M}\mathbf{X}_{A_i}$, the i-th agent shares $\mathbf{W}_{A_i}=[\mathbf{X}_{A_i},\mathbf{C}_{A_i}]\mathbf{D}_{A_i}\in\mathbb{R}^{T\times r}$ with the n-th agent, $\mathbf{C}_{A_i}\in\mathbb{R}^{T\times(r-s)}$, $\mathbf{D}_{A_i}\in\mathbb{R}^{r\times r}$, $r>s$. Then, the process repeat until the 1-st agent receives $\mathbf{M}_{A_2}\dots\mathbf{M}_{A_n}\mathbf{W}_{A_i}$ and computes $\mathbf{M}\mathbf{W}_{A_i}=\mathbf{M}_{A_1}\mathbf{M}_{A_2}\dots\mathbf{M}_{A_n}\mathbf{W}_{A_i}$. Consequently, agent $j=1,\dots,n$ receives Tr values during the protocol.

(ii) $\mathbf{X}_{A_i}^\top\mathbf{M}^{-1}$ is computed using the matrix $\mathbf{W}_{A_i}$ defined before. Since $\mathbf{M}^{-1}=\mathbf{M}_{A_n}^{-1}\dots\mathbf{M}_{A_1}^{-1}$, the n-th agent computes $\mathbf{W}_{A_i}^\top\mathbf{M}_{A_n}^{-1}$. Then, the process repeat until the 1-st agent receives $\mathbf{W}_{A_i}^\top\mathbf{M}_{A_n}^{-1}\dots\mathbf{M}_{A_2}^{-1}$ and computes $\mathbf{W}_{A_i}^\top\mathbf{M}^{-1}=\mathbf{W}_{A_i}^\top\mathbf{M}_{A_n}^{-1}\dots\mathbf{M}_{A_2}^{-1}\mathbf{M}_{A_1}^{-1}$. Again, the j-th agent receives Tr values related to the unknown data from the i-th agent.

In summary, the n-th agent receives Tr values and unknowns $u+T(r-s)+r^2$ (from $\mathbf{X}_{A_i}$, $\mathbf{C}$, $\mathbf{D}$). The solution for $Tr<u+T(r-s)+r^2$ allows to infer that $\mathbf{X}_{A_i}$ is protected if

$$r>\sqrt{Ts-u}.$$

On the other hand, the i-th agent receives $2Tr$ values ($\mathbf{M}\mathbf{W}_{A_i}$, $\mathbf{W}_{A_i}^\top\mathbf{M}^{-1}$) and unknowns T^2 from $\mathbf{M}\Rightarrow r<T/2$.

(iii) Finally, to compute $\mathbf{M}\mathbf{G}_{A_i}$, the i-th agent should define new matrices $\mathbf{C}'_{A_i}\in\mathbb{R}^{T\times(r'-g)}$ and $\mathbf{D}'_{A_i}\in\mathbb{R}^{r'\times r'}$ sharing $\mathbf{W}'_{A_i}=[\mathbf{G}_{A_i},\mathbf{C}'_{A_i}]\mathbf{D}'_{A_i}\in\mathbb{R}^{T\times r'}$, $r'>g$. The computation of $\mathbf{M}\mathbf{W}'$ provides Tr' new values, meaning that after computing $\mathbf{M}\mathbf{X}_{A_i}$, $\mathbf{X}_{A_i}^\top\mathbf{M}^{-1}$ and $\mathbf{M}\mathbf{G}_{A_i}$, the n-th agent has $Tr+Tr'$ values and does not know $u+T(r-s)+r^2+v+T(r'-g)+r'^2$ (from $\mathbf{X}_{A_i}$, $\mathbf{C}_{A_i}$, $\mathbf{D}_{A_i}$, $\mathbf{G}_{A_i}$, $\mathbf{C}'_{A_i}$ and $\mathbf{D}'_{A_i}$ respectively). The solution of the inequality $Tr+Tr'<u+T(r-s)+r^2+v+T(r'-g)+r'^2$ allows to infer that $r'>\sqrt{Ts-u-r^2-v+Tg}>\sqrt{Tg-v}$.

On the other hand, the i-th agent receives $2Tr+Tr'$ and does not know T^2, meaning that $r'<T-2r$. ■

APPENDIX C
PRIVACY ANALYSIS

The proposed approach requires agents to encrypt their data and then exchange that encrypted data. This appendix section analyzes the global exchange of information. First, we show that the proposed privacy protocol is secure in a scenario without collusion, i.e., no alliances between agents (data owners) to determine the private data. Then, we analyze how many agents have to collude for a privacy breach to occur.

A. No collusion between agents

While encrypting sensible data $\mathbf{X}_{A_i}\in\mathbb{R}^{T\times s}$ and $\mathbf{G}_{A_i}\in\mathbb{R}^{T\times g}$ such that $\mathbf{X}_{A_i}$ has u unique values to recover and $\mathbf{G}_{A_i}$ has v unique values that are not in $\mathbf{X}_{A_i}$, the 1-st agent obtains $\mathbf{M}[\mathbf{X}_{A_i},\mathbf{C}_{A_i}]\mathbf{D}_{A_i}\in\mathbb{R}^{T\times r}$, $[[\mathbf{X}_{A_i},\mathbf{C}_{A_i}]\mathbf{D}_{A_i}]^\top\mathbf{M}^{-1}\in\mathbb{R}^{T\times r}$ and $\mathbf{M}[\mathbf{G}_{A_i},\mathbf{C}'_{A_i}]\mathbf{D}'_{A_i}\in\mathbb{R}^{T\times r'}$, $\forall i$, which provides $2nTr+nTr'$ values. At this stage, the agent does not know $\underbrace{T^2}_{\mathbf{M}}+\underbrace{(n-1)u}_{\mathbf{X}_{A_i},\forall i\neq 1}+\underbrace{(n-1)v}_{\mathbf{G}_{A_i},\forall i\neq 1}+\underbrace{(n-1)T(r-s)}_{\mathbf{C}_{A_i},\forall i\neq 1}+\underbrace{(n-1)r^2}_{\mathbf{D}_{A_i},\forall i\neq 1}+\underbrace{(n-1)T(r'-g)}_{\mathbf{C}'_{A_i},\forall i\neq 1}+\underbrace{(n-1)r'^2}_{\mathbf{D}'_{A_i},\forall i\neq 1}$ values. Then, while fitting the LASSO-VAR model, the 1-st agent can recover $\mathbf{M}\mathbf{X}\in\mathbb{R}^{T\times ns}$ and $\mathbf{M}\mathbf{G}\in\mathbb{R}^{T\times ng}$, as shown in [10]. That said, the 1-st agent receives $2nTr+nTr'+nTs+nTg$, and a confidentiality breach occurs if $T(2nr+nr'+ns+ng)\geq T^2+(n-1)[u+v+T(r-s)+r^2+T(r'-g)+r'^2]$.

After a little algebra, it is possible to verify that taking (26) and (27), the previous inequality has no solution in $\mathbb{R}_0^+$.

B. Collusion between agents

A set of agents $\mathcal{C}$ can come together to recover the data of the remaining competitors. This collusion assumes that such agents are willing to share their private data. Let c be the number of agents colluding. In this scenario, the objective

is to determine $\mathbf{M} \in \mathbb{R}^{T\times T}$, knowing $\mathbf{MW}_{A_i} \in \mathbb{R}^{T\times r}$, $\mathbf{W}_{A_i}^{\top}\mathbf{M}^{-1} \in \mathbb{R}^{r\times T}$, $\mathbf{MW'}_{A_i} \in \mathbb{R}^{T\times r'}$, $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i} \in \mathbb{R}^{T\times p}$, and $\mathbf{MY}_{A_i} \in \mathbb{R}^{T\times 1}$, $i \in \mathcal{C}$.

Mathematically, it means that colluders can recover T^2 values by solving $cT(r + r + r' + p + 1)$ equations, which is only possible for $c \geq \lceil \frac{T}{2r+r'+p+1} \rceil$.

REFERENCES

[1] C. Sweeney, R. J. Bessa, J. Browell, and P. Pinson, "The future of forecasting for renewable energy," *Wiley Inter. Rev.: Energ. and Env.*, vol. 9, no. 2, p. e365, Mar. 2020.

[2] J. Tastu, P. Pinson, P.-J. Trombe, and H. Madsen, "Probabilistic forecasts of wind power generation accounting for geographically dispersed information," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 480–489, Jan. 2014.

[3] R. Bessa, A. Trindade, and V. Miranda, "Spatial-temporal solar power forecasting for smart grids," *IEEE Trans. Ind. Informat.*, vol. 11, no. 1, pp. 232–241, Feb. 2015.

[4] Q. Zhu, J. Chen, D. Shi, L. Zhu, X. Bai, X. Duan, and Y. Liu, "Learning temporal and spatial correlations jointly: A unified framework for wind speed prediction," *IEEE Trans. Sustain. Energy*, vol. 11, no. 1, pp. 509–523, Jan. 2020.

[5] Y. Zhao, L. Ye, P. Pinson, Y. Tang, and P. Lu, "Correlation-constrained and sparsity-controlled vector autoregressive model for spatio-temporal wind power forecasting," *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 5029–5040, Sep. 2018.

[6] J. Dowell and P. Pinson, "Very-short-term probabilistic wind power forecasts by sparse vector autoregression," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 63–770, Mar. 2016.

[7] L. Cavalcante, R. J. Bessa, M. Reis, and J. Browell, "LASSO vector autoregression structures for very short-term wind power forecasting," *Wind Energy*, vol. 20, no. 4, pp. 657–675, Apr. 2017.

[8] J. W. Messner and P. Pinson, "Online adaptive lasso estimation in vector autoregressive models for high dimensional wind power forecasting," *Int. Journal of Forecasting*, vol. 35, no. 4, pp. 1485–1498, Oct. 2019.

[9] Y. Zhang and J. Wang, "A distributed approach for wind power probabilistic forecasting considering spatio-temporal correlation without direct access to off-site information," *IEEE Trans. on Power Systems*, vol. 33, no. 5, pp. 5714–5726, Sep. 2018.

[10] C. Gonçalves, R. J. Bessa, and P. Pinson, "A critical overview of privacy-preserving approaches for collaborative forecasting," *Int. Journal of Forecasting*, vol. 37, no. 1, pp. 322–342, Jan. 2021.

[11] B. Sommer, P. Pinson, J. Messner, and D. Obst, "Online distributed learning in wind power forecasting," *Int. Journal of Forecasting*, vol. 37, no. 1, pp. 205–223, Jan. 2021.

[12] V. Berdugo, C. Chaussin, L. Dubus, G. Hebrail, and V. Leboucher, "Analog method for collaborative very-short-term forecasting of power generation from photovoltaic systems," in *Proc. Next Gener. Data Min. Summit*, Greece, Sep. 2011, pp. 1–5.

[13] W. Li, H. Li, and C. Deng, "Privacy-preserving horizontally partitioned linear programs with inequality constraints," *Opt. Letters*, pp. 137–144, 2013.

[14] C. Dwork, K. Talwar, K. Talwar, A. Thakurta, and L. Zhang, "Analyze gauss: optimal bounds for privacy-preserving principal component analysis," in *Forty-sixth Annual ACM Symposium on Theory of Computing*, May 2014, pp. 11–20.

[15] W. Du, Y. S. Han, and S. Chen, "Privacy-preserving multivariate statistical analysis: Linear regression and classification," in *2004 SIAM Int. Conf. on Data Mining*, 2004, pp. 222–233.

[16] Y. Liu, W. Guo, C.-I. Fan, L. Chang, and C. Cheng, "A practical privacy-preserving data aggregation (3PDA) scheme for smart grid," *IEEE Trans. on Ind. Info.*, vol. 15, no. 3, pp. 1767–1774, 2018.

[17] G. Mateos, J. A. Bazerque, and G. B. Giannakis, "Distributed sparse-linear regression," *IEEE Trans. on Signal Proc.*, vol. 58, no. 10, pp. 5262–5276, 2010.

[18] Y. Li, X. Jiang, S. Wang, H. Xiong, and L. Ohno-Machado, "VERTIcal Grid lOgistic regression (VERTIGO)," *J. of the Amer. Medical Inf. Ass.*, vol. 23, no. 3, pp. 570–579, 2015.

[19] B. Elsinga and W. G. van Sark, "Short-term peer-to-peer solar forecasting in a network of photovoltaic systems," *Applied Energy*, vol. 206, pp. 1464–1483, Nov. 2017.

[20] A. Tascikaraoglu, "Evaluation of spatio-temporal forecasting methods in various smart city applications," *Renewable and Sustainable Energy Reviews*, vol. 82, no. 1, pp. 424–435, Feb. 2018.

[21] A. Pourhabib, J. Z. Huang, and Y. Ding, "Short-term wind speed forecast using measurements from multiple turbines in a wind farm," *Technometrics*, vol. 58, no. 1, pp. 138–147, 2016.

[22] C. Dwork and A. Smith, "Differential privacy for statistics: What we know and what we want to learn," *J. of Privacy and Conf.*, vol. 1, no. 2, pp. 135–154, Apr. 2010.

[23] T. Zhang and Q. Zhu, "Dynamic differential privacy for ADMM-based distributed classification learning," *IEEE Trans. on Information Forensics and Security*, vol. 12, no. 1, pp. 172–187, Jan. 2017.

[24] K. Chen and L. Liu, "A survey of multiplicative perturbation for privacy-preserving data mining," in *Privacy-Preserving Data Mining*. Springer, 2008, pp. 157–181.

[25] L. Cavalcante and R. J. Bessa, "Solar power forecasting with sparse vector autoregression structures," in *2017 IEEE Manchester PowerTech*. IEEE, Jun. 2017, pp. 1–6.

[26] P. Bacher, H. Madsen, and H. A. Nielsen, "Online short-term solar power forecasting," *Solar Energy*, vol. 83, no. 10, pp. 1772–1783, Oct. 2009.

[27] D. Harvey, S. Leybourne, and P. Newbold, "Testing the equality of prediction mean squared errors," *Int. Journal of forecasting*, vol. 13, no. 2, pp. 281–291, 1997.

[28] J. Tastu, P. Pinson, E. Kotwa, H. Madsen, and H. A. Nielsen, "Spatio-temporal analysis and modeling of short-term wind power forecast errors," *Wind Energy*, vol. 14, no. 1, pp. 43–60, Jan. 2011.

[29] C. Gonçalves, P. Pinson, and R. J. Bessa, "Towards data markets in renewable energy forecasting," *IEEE Trans. Sustain. Energy*, vol. 12, no. 1, pp. 533–542, Jan. 2021, available online.

[30] J. B. de Souza, V. A. Reisen, G. C. Franco, M. Ispany, P. Bondon, and J. M. Santos, "Generalized additive models with principal component analysis: an application to time series of respiratory disease and air pollution data," *J. of the Royal Stat. Soc. Series C,*, vol. 67, no. 2, pp. 453–480, Feb. 2018.

[31] J. Jiang, "Multivariate functional-coefficient regression models for nonlinear vector time series data," *Biometrika*, vol. 101, no. 3, pp. 689–702, Sep. 2014.

Carla Gonçalves is a Ph.D. candidate in Applied Mathematics from the Faculty of Sciences of the University of Porto (FCUP), and a researcher at INESC TEC, Portugal. She received the M.Sc. in Applied Mathematics from FCUP, in 2015. Her research focuses on probabilistic and collaborative forecasting methods, with a special emphasis on renewable energies.

Ricardo Bessa (M'18–SM'19) received the *Licenciado* (5-years) degree in electrical and computer engineering, the M.Sc. degree in data analysis and decision support systems, and the Ph.D. degree in Sustainable Energy Systems (MIT Portugal) from the University of Porto. He is coordinator of the Center for Power and Energy Systems at INESC TEC. His main research interests include renewable energy, energy analytics, smart grids, and electricity markets. He serves as an Editor for the *IEEE Transactions on Sustainable Energy*.

Pierre Pinson (SM'13, F'20) received the M.Sc. degree in applied mathematics from the National Institute for Applied Sciences, Toulouse, France, and the Ph.D. degree in energetics from Ecole des Mines de Paris, Paris, France. He is a Professor with the Technical University of Denmark, Department of Technology, Management and Economics. His research interests include forecasting, uncertainty estimation, optimization under uncertainty, decision sciences, and renewable energies. He is the Editor-in-Chief for the International Journal of Forecasting.