DISTRIBUTED BAYESIAN ESTIMATION WITH LOW-RANK DATA: APPLICATION TO SOLAR ARRAY PROCESSING

Raksha Ramakrishna, Anna Scaglione, Andreas Spanias, Cihan Tepedelenlioglu

School of ECEE, Arizona State University, Tempe, AZ, USA, 85281

ABSTRACT

In this paper, we present a distributed array processing algorithm to analyze the power output of solar photo-voltaic (PV) installations, leveraging the low-rank structure inherent in the data to estimate possible faults. Our multi-agent algorithm requires near-neighbor communications only and is also capable of jointly estimating the common low rank cloud profile and local shading of panels. To illustrate the workings of our algorithm, we perform experiments to detect shading faults in solar PV installations within a single ZIP code. Additionally, we also derive a Bayesian lower bound on the shading parameter's mean squared estimation error. The results are promising and show that we can successfully estimate the fraction of partial shading in solar installations that can usually go unnoticed.

Index Terms— Distributed array processing, Bayesian estimation, solar panel monitoring, partial shading

1. INTRODUCTION

Partial shading type of faults in solar panels reduce the power output from solar panels to a value below their operating point since panels are shaded due to dirt, soiling, buildings or tree shadows [1, 2]. As a consequence, regular maintenance is of utmost importance in solar PV panel arrays. However, there could be thousands of panels and human inspection to determine what panels are soiled or shaded is an impossible task. Besides, such soiling errors can go unnoticed for a very long time and slowly degrade the performance of the system.

In this paper, we propose a distributed algorithm that can be applied both in the context of a utility-scale array of solar panels or multiple residential installations of solar panels within a certain geographical region. The underlying principle is simple: all the panels in a small enough geographical area see the same levels of irradiance and cloudiness (c.f. Section 4 for corroborating evidence). This lends to the problem a low-rank structure which we utilize to detect the fraction of *partial-shading* or soiling of solar panels in multiple installations. Since it is assumed that very few panels are shaded, the infrequent nature of shading can be modeled by assuming a prior distribution that promotes sparsity, such as that of independent truncated exponentially distributed shading coefficients. Near-neighbor communication based methods are a natural extension to the existing state of the art which use just local information for fault detection. Additionally, such communications could be multipurpose and be used not just for fault detection but for general monitoring and participation in demand-response programs as an aggregate load. This serves as motivation for a decentralized algorithm that is used to find the common feature of cloud-induced shading among the nodes with fewer number of data exchanges whereas the specific partial-shading parameter is computed locally. Under these assumptions, the problem of partial shading estimation can be cast as a decentralized matrix decomposition into a rank-1 matrix, where the left singular vector represents the common attenuation pattern due to the clouds, and it is estimated in a decentralized way via average consensus among agents, while the right singular vector (fractions of partial shading) is *sparse* and it is estimated locally, alternating between the two estimates. We note that existing methods, such as the decentralized power iteration methods [3] could be applied, but they would not leverage fully the structure of the problem. We can think of our problem as an instance of decentralized dictionary learning [4] where the sparse coding step is replaced with a maximum aposteriori (MAP) estimation of shading parameter. The Bayesian framework makes it also possible to provide performance benchmarks. For this purpose we derive an extension of Bayesian Cr̂amer Rao lower bounds (BCRB) for the partial shading parameter mean squared estimation error that, unlike the ones used in the context of Bayesian learning [5, 6], applies to parameters with a finite support prior distribution that may not satisfy certain regularity conditions [7].

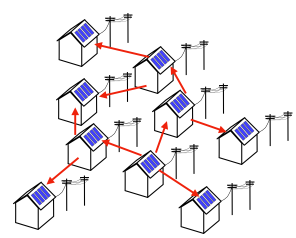

Fig. 1. Schematic showing the communication links between different solar panel installations for distributed estimation.

Related works: Many papers address the fault detection problem. They range from utilizing panel specific electrical characteristics such as their operating points that depend on solar irradiance levels [8] to applying machine learning algorithms [9, 10, 11] including k-means clustering [2] and minimum covariance determinant [1, 12] methods and general statistical methods like multi-resolution signal analysis [13, 14]. They focus mainly on a single installation with an array of solar PV panels. However, in this paper, we take advantage of structure in solar data from different installations to estimate the fraction of partial shading instead of using additional data like irradiance or temperature.

The paper first introduces the distributed algorithm more generally in Section 2 and then presents the application of the introduced framework for partial shading estimation in solar PV panels in Section 3. Solar data from multiple residential rooftop PV panel installations within a ZIP code is used to test the efficacy of the proposed algorithm and results are presented in Section 4.

This project was funded in part by the ASU SenSIP Center and the NSF I/UCRC award 1540040 and by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0000696. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

2. SYSTEM MODEL

Let a discrete set of data at time indices $n = 1, 2, \dots N$ from locations $\ell = 1, 2, \dots L$ be available. Stacking them column-wise we get a matrix, $\boldsymbol{Y} \in \mathbb{R}^{N\times L}$. It is postulated that $\boldsymbol{Y}$ has a specific low-rank structure such that each column of the matrix is a scaled version of the other by a certain attenuation coefficient, thereby giving the following model:

$$\boldsymbol{Y} = \boldsymbol{x}(\boldsymbol{1}-\boldsymbol{\alpha})^T + \begin{bmatrix}\boldsymbol{\eta}_1 & \boldsymbol{\eta}_2 & \cdots & \boldsymbol{\eta}_L\end{bmatrix}, \boldsymbol{\eta}_\ell \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}) \quad (1)$$

where $\boldsymbol{x} \in \mathbb{R}^N$ and belongs to a convex constraint set $\mathcal{C}$, $\boldsymbol{1}$ is a vector of ones with dimension L and $\boldsymbol{\alpha} \in \mathbb{R}^{L\times 1}$. It is assumed that noise covariance matrix $\boldsymbol{\Sigma}$ is diagonal and the scaling vector $\boldsymbol{\alpha}$ is sparse with the support of non-zero entries being finite. The goal is to estimate $\boldsymbol{x}, \boldsymbol{\Sigma}$ and $\boldsymbol{\alpha}$ using $\boldsymbol{Y}$. The problem statement is,

$$\begin{aligned}\min_{\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\Sigma}} \quad & \tfrac{1}{2}\textstyle\sum_{\ell=1}^{L} \left\| \boldsymbol{\Sigma}^{-\frac{1}{2}} \left[\boldsymbol{y}_\ell - (1-\alpha_\ell)\boldsymbol{x}\right]\right\|_2^2 + \lambda \left\|\boldsymbol{\alpha}\right\|_1 \\ \text{subject to} \quad & \boldsymbol{x} \in \mathcal{C},\ \underline{\alpha} \le \alpha_\ell \le \overline{\alpha},\ \ell = 1, 2, \dots L \end{aligned} \quad (2)$$

In our setup, there is no central entity to estimate $\boldsymbol{x}$ and $\boldsymbol{\Sigma}$ so the agents have to do it in a decentralized fashion. From (2), the problem can be treated as a variant of *decentralized matrix factorization* problem where the rank of the matrix is exactly 1.

2.1. Solution approach

We use an alternating minimization method since the cost is non-convex. Firstly, given $\boldsymbol{\alpha}$ and $\boldsymbol{\Sigma}$, maximum likelihood (ML) estimate $\hat{\boldsymbol{x}}_{ML}$, is computed in a distributed manner. Then, given $\hat{\boldsymbol{x}}_{ML}$ and $\boldsymbol{\Sigma}$ we update α_ℓ at location ℓ with the maximum aposteriori (MAP) estimate of α_ℓ by imposing as a prior a truncated exponential distribution, with support between $\overline{\alpha}$ and $\underline{\alpha}$:

$$p(\alpha_\ell) = C^{-1}\lambda e^{-\lambda\alpha_\ell},\ C = e^{-\lambda\underline{\alpha}} - e^{-\lambda\overline{\alpha}},\ \underline{\alpha} \le \alpha_\ell \le \overline{\alpha}. \quad (3)$$

Lastly, given $\boldsymbol{x}$ and $\boldsymbol{\alpha}$, we update covariance matrix $\boldsymbol{\Sigma}$ using residual error and average consensus based method.

Decentralized ML estimate : Let the network of sensors be described by the graph $G = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of vertices and $\mathcal{E}$ the set of edges/connections. Then, cost for each agent at ℓ is,

$$f_\ell(\boldsymbol{x}, \alpha_\ell) = \frac{1}{2}\left(\boldsymbol{y}_\ell - (1-\alpha_\ell)\boldsymbol{x}\right)^T \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{y}_\ell - (1-\alpha_\ell)\boldsymbol{x}\right)$$

and total cost is $F(\boldsymbol{x}, \boldsymbol{\alpha}) = \frac{1}{2}\sum_{\ell=1}^{L} f_\ell(\boldsymbol{x}, \alpha_\ell)$. We use the decentralized Frank-Wolfe [15] method (note that any distributed optimization algorithm is viable) to estimate $\boldsymbol{x}$. Assuming that the values of α_ℓ and $\boldsymbol{\Sigma}$ are fixed for this step, average consensus (AC) is employed to update the estimate of vector $\boldsymbol{x}$ as well as the gradient, $\sum_{\ell=1}^{L} \nabla_{\boldsymbol{x}} f_\ell(\boldsymbol{x}, \alpha_\ell)$. For AC, we use the Metropolis-Hasting update which provides the following weight matrix:

$$\boldsymbol{W} = \mathbb{I} - \boldsymbol{M}\boldsymbol{1} + \boldsymbol{M}, [\boldsymbol{M}]_{ij} = \begin{cases}\left(\max\{d_i, d_j\}\right)^{-1}, (i,j) \in \mathcal{E} \\ 0, (i,j) \notin \mathcal{E}\end{cases}$$

where d_i, d_j are degrees of nodes i and j in the graph G and $\mathbb{I}$ is the identity matrix of size L. The steps of the decentralized Frank-Wolfe method are as follows:

1. Begin with an estimate of $\boldsymbol{x}_\ell(0)$ and $\alpha_\ell(0)$ at each ℓ . Let $\boldsymbol{X}(0) \triangleq \begin{bmatrix}\boldsymbol{x}_1(0) & \boldsymbol{x}_2(0) & \dots & \boldsymbol{x}_L(0)\end{bmatrix} \in \mathbb{R}^{N\times L}$.
2. A step of AC is carried out, $\boldsymbol{X}(t) = \boldsymbol{X}(t-1)\boldsymbol{W}$.
3. At sensor ℓ, the gradient is calculated:

$$\nabla_{\boldsymbol{x}} f_\ell(\boldsymbol{x}_\ell(t), \alpha_\ell) = (1-\alpha_\ell)\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{y}_\ell - (1-\alpha_\ell)\boldsymbol{x}_\ell(t)\right).$$

The full gradient at time t, $\nabla_{\boldsymbol{x}} F(\boldsymbol{x}(t), \boldsymbol{\alpha})$, is unavailable. Hence the local estimate of the gradient at ℓ is given by

$$\begin{aligned}\nabla_{\boldsymbol{x}} F_\ell^t(\boldsymbol{x}_\ell(t)) = \nabla_{\boldsymbol{x}} F_\ell^{t-1}(\boldsymbol{x}_\ell(t-1)) + \nabla_{\boldsymbol{x}} f_\ell(\boldsymbol{x}_\ell(t), \alpha_\ell) \\ - \nabla_{\boldsymbol{x}} f_\ell(\boldsymbol{x}_\ell(t-1), \alpha_\ell).\end{aligned} \quad (4)$$

Then, the AC step is employed,

$$\nabla_{\boldsymbol{x}} F_\ell^{t+1}(\boldsymbol{x}_\ell(t)) = \textstyle\sum_j [\boldsymbol{W}]_{\ell,j} \nabla_{\boldsymbol{x}} F_j^t(\boldsymbol{x}_j(t)) \quad (5)$$

4. To perform the constrained optimization, Frank-Wolfe (FW) update [15] is used which involves projection to the constraint space as a linear optimization problem. General FW step is:

$$\boldsymbol{x}_\ell(t+1) \leftarrow (1-\tau_t)\boldsymbol{x}_\ell(t) + \tau_t \boldsymbol{b}_t^\ell \quad (6)$$

$$\text{where } \boldsymbol{b}_t^\ell := \arg\min_{\boldsymbol{b}\in\mathcal{C}} \langle \boldsymbol{b}, \nabla_{\boldsymbol{x}} F_\ell^{t+1}(\boldsymbol{x}_\ell(t))\rangle \quad (7)$$

where $\tau_t = 2/(t+1)$ is the step size. From [15], it is known that the decentralized optimization problem to estimate $\boldsymbol{x}$ converges to a stationary point since the original problem is non-convex (bi-convex) with respect to both $\boldsymbol{x}$ and $\boldsymbol{\alpha}$.

Updating α: Consider the second step of alternating minimization, i.e. that of estimating α_ℓ. From (1) (dropping time index t),

$$\boldsymbol{y}_\ell = (1-\alpha_\ell)\boldsymbol{x}_\ell + \boldsymbol{\eta}_\ell,\ \boldsymbol{\eta}_\ell \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}). \quad (8)$$

Introduce a modified scalar measurement, $\tilde{y}_\ell$, by multiplying $\boldsymbol{x}_\ell^T\boldsymbol{\Sigma}^{-1}$ on both sides of (8) as $\tilde{y}_\ell \triangleq 1 - \left(\boldsymbol{x}_\ell^T\boldsymbol{\Sigma}^{-1}\boldsymbol{y}_\ell / \boldsymbol{x}_\ell^T\boldsymbol{\Sigma}^{-1}\boldsymbol{x}_\ell\right)$. Then,

$$\tilde{y}_\ell = \alpha_\ell + \tilde{\eta}_\ell,\ \tilde{\eta}_\ell \sim \mathcal{N}\left(0, \gamma_\ell^2\right),\ \gamma_\ell^2 = \left(\boldsymbol{x}_\ell^T\boldsymbol{\Sigma}^{-1}\boldsymbol{x}_\ell\right)^{-1}. \quad (9)$$

With prior on α_ℓ as in (3), MAP estimate of α_ℓ at iteration t is

$$\hat{\alpha}_\ell(t) = \begin{cases}\underline{\alpha}, & \tilde{y}_\ell \le \underline{\alpha} + \lambda\gamma_\ell^2 \\ \tilde{y}_\ell - \lambda\gamma_\ell^2, & \underline{\alpha} + \lambda\gamma_\ell^2 \le \tilde{y}_\ell \le \overline{\alpha} + \lambda\gamma_\ell^2 \\ \overline{\alpha}, & \tilde{y}_\ell \ge \overline{\alpha} + \lambda\gamma_\ell^2\end{cases} \quad (10)$$

Estimation of $\boldsymbol{\Sigma}$: We assume that covariance matrix $\boldsymbol{\Sigma}$ is diagonal:

$$\boldsymbol{\Sigma} = \mathrm{diag}(\boldsymbol{\sigma})\mathbb{I},\ i = 1, 2, \dots, N \quad (11)$$

Then, ML estimate of $[\sigma]_i$ at time t is,

$$[\sigma]_i(t) = L^{-1}\textstyle\sum_{\ell=1}^{L}\left([\boldsymbol{y}_\ell]_i - \alpha_\ell [\boldsymbol{x}_\ell(t)]_i\right)^2. \quad (12)$$

AC can be employed on the N variances by letting

$$[\boldsymbol{\sigma}_\ell]_i(t) = \left([\boldsymbol{y}_\ell]_i - \alpha_\ell[\boldsymbol{x}_\ell(t)]_i\right)^2,\ \ell = 1, 2, \dots L \quad (13)$$

One iteration of AC is $\boldsymbol{Q}(t+1) = \boldsymbol{Q}(t)\boldsymbol{W}$ where

$$\boldsymbol{Q}(t) \triangleq \begin{bmatrix}\boldsymbol{\sigma}_1(t) & \boldsymbol{\sigma}_2(t) & \dots & \boldsymbol{\sigma}_L(t)\end{bmatrix}. \quad (14)$$

Algorithm 1 describes the steps of the proposed distributed Bayesian estimation method. Next, we present a lower bound for estimation error of α_ℓ given $\boldsymbol{x}$ and $\boldsymbol{\Sigma}$.

Theorem 1 (Bayesian CRB for finite support prior). *A lower bound for the mean squared error of the MAP estimator in* (10) *given* $\boldsymbol{x}$ *and* $\boldsymbol{\Sigma}$ *is*

$$\mathbb{E}_{\alpha_\ell, \tilde{y}_\ell}\left[(\hat{\alpha}_\ell - \alpha_\ell)^2 \middle| \boldsymbol{x}, \boldsymbol{\Sigma}\right] \geq \frac{\gamma_\ell^2(1-\nu)^2}{\lambda^2\gamma_\ell^2+1} \tag{15}$$

where
$$\nu = (p(\overline{\alpha}) + p(\underline{\alpha}))\left[\gamma_\ell\phi(\lambda\gamma_\ell) + \lambda\gamma_\ell^2\Phi(\lambda\gamma_\ell)\right] + Rp(\underline{\alpha})$$
$$- p(\overline{\alpha})\left[\gamma_\ell\phi\left(\lambda\gamma_\ell - \frac{R}{\gamma_\ell}\right) + (\lambda\gamma_\ell^2 - R)\Phi\left(\lambda\gamma_\ell - \frac{R}{\gamma_\ell}\right)\right]$$
$$- p(\underline{\alpha})\left[\gamma_\ell\phi\left(\lambda\gamma_\ell + \frac{R}{\gamma_\ell}\right) + (\lambda\gamma_\ell^2 + R)\Phi\left(\lambda\gamma_\ell + \frac{R}{\gamma_\ell}\right)\right],$$
$$\Phi(x) = \int_{-\infty}^{x} e^{-\frac{z^2}{2}}\, dz,\ \phi(x) = (\sqrt{2\pi})^{-1}e^{-x^2/2},\ R = \overline{\alpha} - \underline{\alpha}.$$

Proof is omitted due to lack of space. Since the bound applies to a case when $\boldsymbol{x}$ and $\boldsymbol{\Sigma}$ are known exactly, it is not likely to be tight when the estimation error in $\boldsymbol{x}$ and $\boldsymbol{\Sigma}$ are significant. This is seen in the numerical results.

Algorithm 1 Distributed Bayesian estimation

Input: $\boldsymbol{x}_\ell(0), \alpha_\ell(0), \boldsymbol{\sigma}_\ell(0)\ \forall \ell, T, M$ and $\boldsymbol{W}$.
while iter $\leq M$ **do**
  for $t = 0 : T-1$ **do**
    $\boldsymbol{X}(t+1) \leftarrow \boldsymbol{X}(t)\boldsymbol{W}$
    $\boldsymbol{x}_\ell(t+1) \leftarrow (1-\tau_t)\boldsymbol{x}_\ell(t+1) + \tau_t \boldsymbol{b}_t^\ell\ \forall \ell$ (7)
    Update $\hat{\alpha}_\ell(t)\ \forall \ell$ from (10)
  end for
  for $k = 1 : K$ **do**
    $[\boldsymbol{\sigma}_\ell]_i(1) = \left([\boldsymbol{y}_\ell]_i - \alpha_\ell(T)\left[\boldsymbol{x}_\ell(T)\right]_i\right)^2,\ \ell \in \mathcal{L}_i$
    $\boldsymbol{Q}(k+1) = \boldsymbol{Q}(k)\boldsymbol{W}$
  end for
  $\boldsymbol{x}_\ell(0) \leftarrow \boldsymbol{x}_\ell(T), \alpha_\ell(0) \leftarrow \alpha_\ell(T), \boldsymbol{\sigma}_\ell(0) \leftarrow \boldsymbol{\sigma}_\ell(K),\ \forall \ell$
  iter = iter+1
Output: $\boldsymbol{x}_\ell, \alpha_\ell, \boldsymbol{\sigma}_\ell\ \forall \ell$

3. PARTIAL SHADING ESTIMATION IN SOLAR PANELS

The system model described in Section 2 can be applied to detect partial shading type of faults.

Model for small utility-scale farm: If the area covered by the panels in a farm is relatively small, we can assume that the cloud cover induced attenuation of solar power is the same across each panel j at a given time instant n on day d. Assuming that all the panels have uniform characteristics, the solar power data matrix $\boldsymbol{Y}_d \in \mathbb{R}^{N\times P}$ constructed using power measurements from panels $j = 1, \ldots, P$ for times $n = 1, 2, \ldots N$ on a day d is

$$\boldsymbol{Y}_d = \boldsymbol{w}_d \times (\boldsymbol{1} - \boldsymbol{\alpha}_d)^T,\ \ \boldsymbol{w}_d \in \mathbb{R}^N,\ \boldsymbol{\alpha}_d \in \mathbb{R}^P \tag{16}$$

where $\boldsymbol{w}_d$ is the vector of power measurements on day d. If the panels are operating normally (without shading faults) then $\boldsymbol{\alpha} = \boldsymbol{0}$. If not, the support of $\boldsymbol{\alpha}$ corresponds to panels which are affected by shading/soiling.

Model for installations in a ZIP code: To detect faults among multiple installations within a ZIP code using solar power data $w_{\ell,d}[n]$ at location $\ell = 1, 2, \ldots L$, time instant $n = 1, 2, \ldots N$ and day d, normalization by the sunny day solar power pattern, $s_{\ell,d}[n]$, is necessary to assume the low-rank structure. In general this pattern changes slowly with the season and can be estimated locally for day

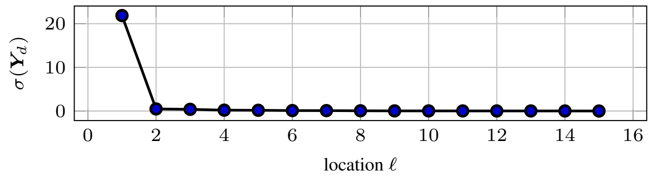

Fig. 2. Singular values of measurement matrix $\boldsymbol{Y}_d$: low rank with one dominant singular value. Solar power data from PV panels at different locations within a ZIP code.

d [16]. In the absence of faults the data matrix $\boldsymbol{Y}_d \in \mathbb{R}^{N\times L}$ has entries $[\boldsymbol{Y}_d]_{n,\ell} = w_{\ell,d}[n]/s_{\ell,d}[n]$. where $x_{\ell,d}[n] \triangleq w_{\ell,d}[n]/s_{\ell,d}[n]$ is interpreted as the attenuation due to cloud cover. This attenuation is approximately the same within a ZIP code. Further evidence is provided in Fig. 2 that shows the singular values of the matrix $\boldsymbol{Y}_d$. Due to partial shading fault in the system, further attenuation of power is observed. It then becomes important to distinguish the cause of attenuation which could be due to cloud cover and/or partial shading. Let the common attenuation vector for a given day be denoted as $\boldsymbol{x}_d \in \mathbb{R}^N$ and fraction of shading at each location be $\alpha_{\ell,d}$. Then, the problem of partial shading detection can be modeled similar to (1) where the measured attenuation at location ℓ, $\boldsymbol{y}_{\ell,d} \in \mathbb{R}^N$ is the product of cloud cover induced attenuation and partial shading fault,

$$\boldsymbol{y}_{\ell,d} = (1-\alpha_{\ell,d})\boldsymbol{x}_d + \boldsymbol{\eta}_{\ell,d},\ \ \boldsymbol{Y}_d = \boldsymbol{x}_d \times (\boldsymbol{1} - \boldsymbol{\alpha}_d) \tag{17}$$

Thus, the problem of fault detection is to estimate the common attenuation vector $\boldsymbol{x}_d$ in order to determine $\boldsymbol{\alpha}_d \in \mathbb{R}^L$. Note that the model for $\boldsymbol{Y}_d$ is same as in (1). We assume that there are fewer locations with faulty panels than the total number of locations. This promotes sparsity in $\boldsymbol{\alpha}_d$. Also, attenuation by definition is less than 1. Therefore, $\boldsymbol{0} \preceq \boldsymbol{\alpha}_d \preceq \boldsymbol{1}$ and $\boldsymbol{0} \preceq \boldsymbol{x}_d \preceq \boldsymbol{1}$.

The estimation algorithm starts with random initialization for all parameters. The FW update for $\boldsymbol{x}_d$ is from (6) and $\boldsymbol{b}_t^\ell$ is

$$\left[\boldsymbol{b}_t^\ell\right]_i = \begin{cases} 1, & \left[\nabla_{\boldsymbol{x}} F_\ell^{t+1}(\boldsymbol{x}_{\ell,d}(t))\right]_i < 0 \\ 0, & \left[\nabla_{\boldsymbol{x}} F_\ell^{t+1}(\boldsymbol{x}_{\ell,d}(t))\right]_i \geq 0 \end{cases}. \tag{18}$$

The update to $\alpha_{\ell,d}$ is from (10) where $\underline{\alpha} = 0,\ \overline{\alpha} = 1$.

4. NUMERICAL RESULTS

Simulations on synthetic data: Synthetic data is generated according (1). We generate parameter α_ℓ using (3) with $\underline{\alpha} = 0, \overline{\alpha} = 1$. The noise covariance matrix is simulated as $\boldsymbol{\Sigma} = \sigma^2\mathbb{I}$. Parameter $\boldsymbol{x}$ has each entry drawn at random, $[\boldsymbol{x}]_i \sim \mathcal{U}(0,1)$ where $\mathcal{U}(.)$ is

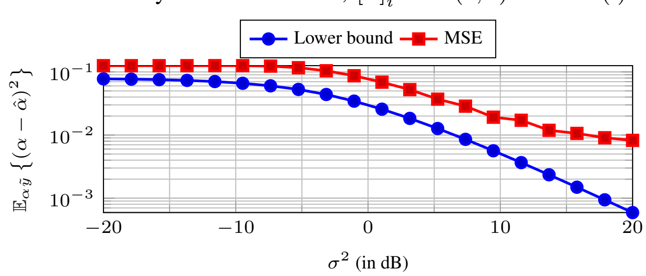

Fig. 3. Comparison of lower bound with MSE for synthetic data when the prior of α_ℓ is constrained to be between 0 and 1 which is what we require in the real-world example of fault detection in solar PV panels.

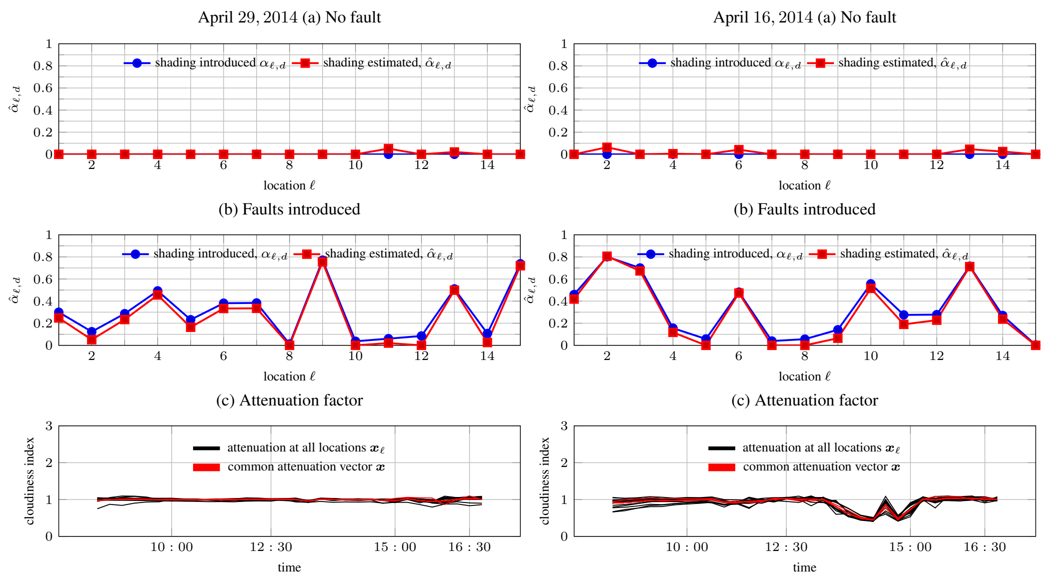

Fig. 4. Results for April 29 and April 16, 2014 when (a) *no fault* introduced and (b) when faults are introduced according to (3) . In (c), attenuation vector for all locations, $\boldsymbol{x}_\ell$ is plotted prior to introduction of faults. The estimated $\boldsymbol{x}$ is plotted in red after the introduction of faults.

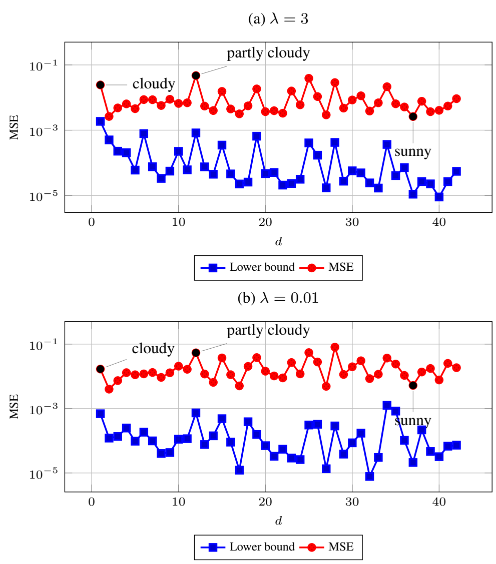

Fig. 5. Comparison of lower bound with mean squared error for every day d for (a) $\lambda = 3$ and (b) $\lambda = 0.01$. Examples of cloudy, partly cloudy and sunny days are marked on both plots.

uniform distribution. Graph $G = (\mathcal{V}, \mathcal{E})$ is constructed as a Erdos-Renyi graph with connectivity $p = 0.3$. Fig.3 shows lower bound on the estimation error. As seen, the bound is not tight. This is due to the non-convexity of the problem due to which it only converges to a stationary point [15] which compounds the error in estimating α_ℓ.

Fault detection for solar panels: We have solar power data from SolarCity ordered by ZIP code. Faults (amount of partial shading) are generated using the prior distribution on $\alpha_{\ell,d}$ from (10) and introduced artificially in the data. We apply the proposed estimation technique to identify installations with partial shading faults within a ZIP code. The algorithm returns a value of $\alpha_{\ell,d}$ which is the fraction of soiling/ partial shading. Firstly, the results of estimation of α_ℓ are shown for two days-one sunny with little attenuation and other with varying cloud cover in Fig.4. As expected, the value of $\alpha_{\ell,d} \approx 0,\ \forall \ell$ for both the days when no fault is introduced. Then, the estimated partial shading $\tilde{\alpha}_\ell$ and the introduced shading fault are plotted. The estimated attenuation vector for the day is also plotted.

Lower bound on partial shading estimation error: We run simulations to test the Bayesian lower bound in (15) while estimating the partial shading parameter $\alpha_{\ell,d}$. To compute the lower bound for each day, the common attenuation vector, $\boldsymbol{x}$ is the average: $\boldsymbol{x} = L^{-1}\boldsymbol{Y}\boldsymbol{1}^T$ assuming there is *no fault* in any panel. This only serves as a benchmark to compute the lower bound on mean squared error (MSE) of $\alpha_{\ell,d}, \ell = 1, 2, \ldots, L$ and not used in the process of estimating $\alpha_{\ell,d}$. In Fig. 5 different amount of faults are introduced with $\lambda = 0.01$ and $\lambda = 3$ for all days to compare the proximity of the lower-bound to MSE with changing weather conditions. As examples, cloudy, partly cloudy and sunny days are marked on the plot to show that the error on partly cloudy and cloudy days is higher than on sunny days in general. On partly cloudy days, the low rank structure is not exact due to the presence of uncoordinated spikes in power at different locations due to attenuation from local clouds.

5. CONCLUSIONS

This paper introduced a distributed Bayesian estimation technique and utilized it to estimate partial shading fraction in rooftop solar PV installations. A lower bound on the mean squared error incurred in estimation of shading fraction was also derived. The proposed fault detection method was tested on data from multiple solar panel installations within a ZIP code in California and the results were satisfactory. Timely action to alleviate shading can lead to potential reduction in inverter transients. Future work involves developing an online version of the algorithm and extending the application to utility scale installations spread over larger geographical regions where rank 1 assumption needs to be relaxed and cloud motion dynamics must be taken into consideration.

6. REFERENCES

[1] H. Braun, S. T. Buddha, V. Krishnan, A. Spanias, C. Tepedelenlioglu, T. Yeider, and T. Takehara, "Signal Processing for fault detection in photovoltaic arrays," in *Proc of IEEE ICASSP, Kyoto*, March 2012, pp. 1681–1684.

[2] S. Rao, D. Ramirez, H. Braun, J. Lee, C. Tepedelenlioglu, E. Kyriakides, D. Srinivasan, J. Frye, S. Koizumi, Y. Morimoto, and A. Spanias, "An 18 kW solar array research facility for fault detection experiments," in *Mediterranean Electrotechnical Conference (MELECON), Limassol*, April 2016.

[3] A. Scaglione, R. Pagliari, and H. Krim, "The Decentralized Estimation of the Sample Covariance," in *Conference Record of the 42nd Asilomar Conference on Signals, Systems and Computers*, 2008.

[4] H.-T. Wai, T.-H. Chang, and A. Scaglione, "A consensus-based decentralized algorithm for non-convex optimization with application to dictionary learning," in *Proceedings of the IEEE ICASSP*, 2015.

[5] M. E. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," *Journal of Machine Learning Research*, 2000.

[6] R.Prasad and C.R.Murthy, "Cramér-Rao-Type Bounds for Sparse Bayesian Learning," *IEEE Transactions on Signal Processing*, 2013.

[7] R. D. Gill and B. Y. Levit, "Applications of the van Trees Inequality: A Bayesian Cramér-Rao Bound," *Bernoulli Society for Mathematical Statistics and Probability*, vol. 1, no. 1/2, pp. 59–79, Jun 1995.

[8] K. A. Kim, G.-S. Seo, B.-H. Cho, and P. T. Krein, "Photovoltaic Hot-Spot Detection for Solar Panel Substrings Using AC Parameter Characterization," *IEEE Transcations on Power Electronics*, 2016.

[9] Y. Zhao, R. Ball, J. Mosesian, J.-F. de Palma, and B. Lehman, "Graph-Based Semi-supervised Learning for Fault Detection and Classification in Solar Photovoltaic Arrays," *IEEE Transcations on Power Electronics*, 2015.

[10] M. N. Akram and S. Lotfifard, "Modeling and Health Monitoring of DC Side of Photovoltaic Array," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 4, 2015.

[11] S. Katoch and et.al, "Shading Prediction, Fault Detection and Consensus Estimation for Solar Array Control," in *IEEE International Conference on Industrial Cyber-Physical Systems (ICPS)*, 2018.

[12] H. Braun, C. Tepedelenlioglu, A. Spanias, and M. Banavar, *Signal Processing for Solar Array Monitoring, Fault Detection, and Optimization, Synthesis Lectures on Power Electronics*, ser. ISBN 978-1608459483. Morgan & Claypool, 2012.

[13] Z. Yi and A. H. Etemadi, "Fault Detection for Photovoltaic Systems Based on Multi-Resolution Signal Decomposition and Fuzzy Inference Systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 3, 2017.

[14] Radu Platon and Jacques Martel and Norris Woodruff and Tak Y. Chau, "Online Fault Detection in PV Systems," *IEEE Transactions on Sustainable Energy*, 2015.

[15] H.-T. Wai, J. Lafond, A. Scaglione, and E. Moulines, "Decentralized Frank–Wolfe Algorithm for Convex and Nonconvex Problems," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, Nov 2017.

[16] R. Ramakrishna and A. Scaglione, "A Compressive Sensing Framework for the analysis of Solar Photo-Voltaic Power," in *Conference Record of the Fiftieth Asilomar Conference on Signals, Systems and Computers*, 2016, pp. 308–312.