ADAPTIVE DISTRIBUTED SPARSITY-AWARE MATRIX DECOMPOSITION

Ioannis D. Schizas

Department of EE, Univ. of Texas at Arlington, 416 Yates Street, Arlington, TX 76011, USA

ABSTRACT

Data covariance matrices that consist of sparse factors arise in settings where the field sensed by a network of sensors is formed by localized sources. It is established that the task of identifying source-informative sensors boils down to estimating the support of the underlying sparse covariance factors. Relying on norm-one regularization a distributed sparsity-aware framework is developed. The associated minimization problems are solved using computationally efficient coordinate descent iterations that are combined with matrix deflation mechanisms. A simple scheme is also developed to set appropriately the sparsity-adjusting coefficients which can provably recover the support of a covariance matrix factor. Adaptive implementations that account for time-varying settings are also considered. The novel utilization of covariance sparsity does not require knowledge of the data model parameters, while numerical tests demonstrate that the novel schemes outperform existing alternatives.

***Index Terms*—** Distributed processing, sparsity, adaptive algorithms

1. INTRODUCTION

Sensor networks are well fitted for applications including surveillance and health monitoring of large structures. The major task of sensors is to collect and process information about a sensed field where sources of interest may be present. Such sources, e.g., moving person/object, thermal sources and so on, in practical settings are localized and affect a small number of sensors. Thus, only sensors acquiring information about a source have to stay active and perform sensing and processing. Different approaches have been developed to perform sensor selection for estimation/tracking [1–4] and detection applications [5]. These works focus on optimizing a pre-specified estimation/detection performance metric while respecting constraints either in power, or, the number of active sensors that collect information. However, existing approaches either rely on the availability of the underlying data and source models parameters [1–4], or are not amenable to distributed processing [1–3, 5]. The goal in the present work is to design *distributed* schemes that have the potential to identify *all* source-informative sensors without requiring knowledge of the data model parameters.

Sensors that are located close to a source acquire data measurements that tend to be *correlated*. It turns out that the covariance matrix of the sensor measurements can be analyzed into sparse factors. Sparsity has been exploited in a broad range of applications including sparse regression and sub-Nyquist sampling [6, 7]. Interestingly, the problem of determining the source-informative sensors boils down to the task of decomposing the data covariance matrix into sparse factors and determining their support (position of nonzero entries). Some existing sparse matrix decomposition techniques assume that the unknown sparse factors are orthogonal [8–12]. Other matrix factorization techniques decompose matrices into factors with nonnegative entries [13–16]. Sparsity inducing mechanisms are incorporated in [13], though no guidelines are provided on how to determine the sparsity level of the unknown factors. Further, the latter matrix decomposition techniques are not distributed.

The novel sparsity-aware matrix decomposition framework is built by augmenting a least-squares matrix factorization cost with pertinent ℓ_1- regularization terms that exploit the sparsity present in the, not necessarily orthogonal, covariance factors. A distributed sparsity-aware matrix decomposition (SMD) framework is put forth, while coordinate descent iterations are employed to estimate the underlying covariance factors and determine their support (Sec. 3). Deflation is utilized to further simplify the decomposition schemes, and facilitate the development of a distributed scheme that sets appropriately the sparsity-adjusting coefficients (Secs. 3.1 and 3.2). The latter selection technique can provably recover the support of an underlying covariance factor. Adaptive implementations are also considered in Sec. 4 that can handle time-varying settings. Numerical tests show that the novel factorization schemes outperform existing alternatives (Sec. 5).

2. PROBLEM FORMULATION AND PRELIMINARIES

Consider a network comprising p sensors that monitor a field which is formed by r zero-mean uncorrelated stationary sources represented through the signals $s_\rho(t)$, with $\rho = 1, \ldots, r$. Each sensor $\{S_j\}_{j=1}^p$ acquires local scalar measurements $\{x_j(t)\}_{j=1}^p$ for the sensed field at discrete time instances $t = 0, 1, 2, \ldots$. For example, in Fig. 1 there are $p = 12$ sensors and $r = 3$ sources of interest, namely $s_1(t)$, $s_2(t)$ and $s_3(t)$. Source $s_1(t)$ affects the measurements of sensors S_4, S_5, S_6, S_7, source $s_2(t)$ is sensed by sensors S_8, S_9 and source $s_3(t)$ by S_1, S_3. The probabilistic distribution and position of the sources are unknown. Utilizing the sensor data $\{x_j(t)\}_{j=1}^p$, acquired over a time-horizon $[0, N-1]$, this work develops techniques that focus on i) identifying the informative sensors that sense field source(s); and on ii) identifying groups of sensors that observe the same source. Every sensor is connected with neighboring sensors that can be reached via single-hop communications. Let $\mathcal{N}_j$ denote the single-hop neighborhood of S_j.

The sensor observations $\mathbf{x}_t := [x_1(t), \ldots, x_p(t)]^T$ relate to the field sources $\{s_\rho(t)\}_{\rho=1}^r$ via the linear model

$$\mathbf{x}_t = \textstyle\sum_{\rho=1}^r \mathbf{b}_\rho s_\rho(t) + \mathbf{w}_t = \mathbf{B}\mathbf{s}_t + \mathbf{w}_t, \qquad (1)$$

where the columns of $\mathbf{B} := [\mathbf{b}_1, \ldots, \mathbf{b}_r]$ correspond to the *unknown* regressors, $\mathbf{s}_t := [s_1(t), \ldots, s_r(t)]^T$ contains the source signals, while $\mathbf{w}_t := [w_1(t), \ldots, w_p(t)]^T$ is the zero-mean white sensing noise. Regressor $\mathbf{b}_\rho \in \mathbb{R}^{p\times 1}$ has nonzero entries at these positions whose indices correspond to sensors whose measurements are affected by source $s_\rho(t)$ for $\rho = 1, \ldots, r$. For instance in Fig. 1, $\mathbf{b}_1$ will have nonzero entries in positions $4, 5, 6, 7$, $\mathbf{b}_2$ in positions $8, 9$ and $\mathbf{b}_3$ in positions $1, 3$, while the rest of their entries are zero

This work is supported by the NSF grant CCF 1218079 and UTA.

since the corresponding sensors are just sensing noise. Note that the support of $\mathbf{b}_\rho$ indicates which sensors sense $s_\rho(t)$.
Based on the data model in (1) the covariance $\mathbf{\Sigma}_x$ is written as

$$\mathbf{\Sigma}_x = \mathbf{B}\mathbf{D}_s\mathbf{B}^T + \sigma_w^2\mathbf{I}_{p\times p} = \bar{\mathbf{H}}\bar{\mathbf{H}}^T + \sigma_w^2\mathbf{I}_{p\times p}, \qquad (2)$$

where $\mathbf{D}_s$ corresponds to the diagonal covariance matrix of the source vector $\mathbf{s}_t$, σ_w^2 indicates the noise variance, $\mathbf{I}_{p\times p}$ denotes the $p \times p$ identity matrix and $\bar{\mathbf{H}} := \mathbf{B}\mathbf{D}_s^{1/2}$. Further, let $\mathbf{M}_x := \mathbf{\Sigma}_x - \sigma_w^2\mathbf{I}_{p\times p} = \bar{\mathbf{H}}\bar{\mathbf{H}}^T$, correspond to the noiseless signal covariance matrix. Notice that the support of $\mathbf{B}$ and $\bar{\mathbf{H}}$ are identical since $\mathbf{D}_s$ is a diagonal matrix. The columns of $\bar{\mathbf{H}}$ are not necessarily orthogonal while the support may contain negative entries. It turns out that identifying the set of source-informative sensors in the WSN boils down to locating where the nonzero entries are in $\bar{\mathbf{H}}$. Oftentimes the ensemble covariance $\mathbf{\Sigma}_x$ is not known. Thus, in what follows sample-average based estimates for the covariance entries will be formed, namely $\hat{\mathbf{\Sigma}}_x(j,j') = N^{-1}\sum_{t=0}^{N-1} x_j(t)x_{j'}(t)$, using sensor data $\{\mathbf{x}_t\}_{t=0}^{N-1}$. Further, an estimate for $\mathbf{M}_x$ will be obtained as $\hat{\mathbf{M}}_x := \hat{\mathbf{\Sigma}}_x - \hat{\mathbf{D}}_w$, where $\hat{\mathbf{D}}_w := \text{diag}(\hat{\sigma}_{w,1}^2, \ldots, \hat{\sigma}_{w,p}^2)$ while $\hat{\sigma}_{w,j}^2$ denotes the noise variance estimate at sensor S_j. Such an estimate can be formed when sources are not present in the field (i.e., $x_j(t) = w_j(t)$) via e.g., sample-averaging $N_w^{-1}\sum_{\tau=0}^{N_w-1} x_j^2(\tau)$, using N_w noisy samples. A distributed sparsity-aware matrix factorization framework is developed next.

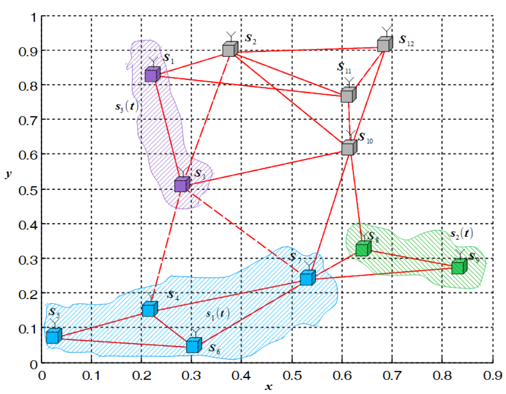

Fig. 1. A sensor network sensing three field sources.

3. DISTRIBUTED ℓ_1-AWARE MATRIX DECOMPOSITION

Motivated by $\mathbf{M}_x = \bar{\mathbf{H}}\bar{\mathbf{H}}^T$ one reasonable way to determine factors for the signal covariance $\hat{\mathbf{M}}_x$ involves minimization of the cost $\|\hat{\mathbf{M}}_x - \mathbf{H}\mathbf{H}^T\|_F^2$ with respect to (wrt) $\mathbf{H} \in \mathbb{R}^{p\times r}$, while $\|\cdot\|_F$ denotes the Frobenius norm. However, such a minimization task is not guaranteed to provide sparse factors, challenging the determination of source-informative sensors, let alone associating sources with sensors. It is of interest to enhance the latter 'least-squares' like matrix decomposition cost with mechanisms that induce sparsity.

Motivated by the sparsity-inducing mechanisms used in Lasso-based regression [7], as well as in sparse principal component analysis techniques [8–12], an ℓ_1 regularization term is added in $\|\hat{\mathbf{M}}_x - \mathbf{H}\mathbf{H}^T\|_F^2$ to effect sparsity in the unknown factors $\mathbf{H}$. A pertinent formulation for estimating the sparse columns (factors) of $\bar{\mathbf{H}}$ while complying with the single-hop communication topology was considered in [17]. Sparse factor estimates $\hat{\mathbf{H}}$ can be obtained by solving:

$$\hat{\mathbf{H}} = \arg\min_{\mathbf{H}} \left\| \mathbf{E} \odot (\hat{\mathbf{M}}_x - \mathbf{H}\mathbf{H}^T) \right\|_F^2 + \textstyle\sum_{\rho=1}^r \lambda_\rho \|\mathbf{h}_\rho\|_1, \qquad (3)$$

where $\hat{\mathbf{H}} := [\hat{\mathbf{h}}_1 \ldots \hat{\mathbf{h}}_r]$, while $\mathbf{E}$ denotes the adjacency matrix of the WSN communication graph and $\odot$ indicates entry-wise matrix product. The Frobenius term in (3) can be rewritten as $\sum_{j=1}^p \sum_{j'\in\mathcal{N}_j\cup\{j\}} \left(\hat{\mathbf{M}}_x(j,j') - \sum_{\ell=1}^r \mathbf{H}(j,\ell)\mathbf{H}(j',\ell)\right)^2$ which involves the entries $\hat{\mathbf{M}}_x(j,j')$ for $j \in \mathcal{N}_j$ that can be found by sensor S_j after communicating with its neighbors in $\mathcal{N}_j$.

Relying on block coordinate descent techniques [18, pg. 160] an iterative minimization algorithm was derived in [17], where the cost in (3) is recursively minimized wrt an entry of $\mathbf{H}$, while keeping the remaining elements in $\mathbf{H}$ fixed. During one coordinate descent cycle all the entries of matrix $\mathbf{H}$ are updated. In order to update the (j,ρ)-th entry of matrix $\mathbf{H}$ during cycle k, namely $\hat{\mathbf{H}}^k(j,\rho)$, all the entries of $\mathbf{H}$, but $\mathbf{H}(j,\rho)$, in (3) are set equal to their most up-to-date value, denoted as $\check{\mathbf{H}}^k(m,n)$. During cycle k if the (m,n)-th entry of $\mathbf{H}$ is updated, before $\mathbf{H}(j,\rho)$ then $\check{\mathbf{H}}^k(m,n) = \hat{\mathbf{H}}^k(m,n)$ (the most recent updated value for $\mathbf{H}(m,n)$). Otherwise, if $\mathbf{H}(m,n)$ is updated after $\mathbf{H}(j,\rho)$, then $\check{\mathbf{H}}^k(m,n) = \hat{\mathbf{H}}^{k-1}(m,n)$.

Applying the Karush-Kuhn-Tucker optimality conditions [18, pg. 316] (details in [17,19]) $\hat{\mathbf{H}}^k(j,\rho)$ is found as the value that results the minimum possible cost in (3) among the candidate values: i) $h = 0$; ii) the real positive roots of the third-degree polynomial

$$\begin{aligned} 4\cdot h^3 + 4[\textstyle\sum_{\mu\in\mathcal{N}_j} [\check{\mathbf{H}}^k(\mu,\rho)]^2 - \delta_M^k(j,j,\rho)]\cdot h \qquad (4)\\ + \lambda_\rho - [4\textstyle\sum_{\mu\in\mathcal{N}_j} \delta_M^k(j,\mu,\rho)\check{\mathbf{H}}^k(\mu,\rho)] = 0; \end{aligned}$$

and iii) the real negative roots of the third-degree polynomial

$$\begin{aligned} 4\cdot h^3 + 4[\textstyle\sum_{\mu\in\mathcal{N}_j} [\check{\mathbf{H}}^k(\mu,\rho)]^2 - \delta_M^k(j,j,\rho)]\cdot h \qquad (5)\\ - \lambda_\rho - [4\textstyle\sum_{\mu\in\mathcal{N}_j} \delta_M^k(j,\mu,\rho)\check{\mathbf{H}}^k(\mu,\rho)] = 0, \end{aligned}$$

where $\delta_M^k(j,\mu,\rho) := \hat{\mathbf{M}}_x(j,\mu) - \sum_{\ell=1,\ell\neq\rho}^r \check{\mathbf{H}}^k(j,\ell)\check{\mathbf{H}}(\mu,\ell)$. Sensor S_j is responsible for forming updates/estimates for the jth row of matrix $\bar{\mathbf{H}}$, namely $\{\hat{\mathbf{H}}^k(j,1),\ldots,\hat{\mathbf{H}}^k(j,r)\}$. The roots can be found using standard techniques, e.g, companion matrices [17,20]. Note that S_j can evaluate the coefficients of the polynomials in (4) and (5) by exchanging information only with its neighbors in $\mathcal{N}_j$. Specifically, sensor S_j receives $\{\hat{\mathbf{H}}^{k-1}(\mu,1),\ldots,\hat{\mathbf{H}}^{k-1}(\mu,r)\}$ from its single-hop neighbors $\mu \in \mathcal{N}_j$ and forms $\delta_M^k(j,\mu,\rho)$. Similarly, S_j transmits to its neighbors the r scalar local updates for the jth row of $\mathbf{H}$, namely $\{\hat{\mathbf{H}}^{k-1}(j,1),\ldots,\hat{\mathbf{H}}^{k-1}(j,r)\}$. In [17,19] is established that the novel distributed (D-) SMD scheme converges at least to a stationary point of the cost in (3).

3.1. Deflation-Based D-SMD

The D-SMD formulation developed in [17] involves the estimation of multiple factors $\{\mathbf{h}_\rho\}_{\rho=1}^r$. Notice that different λ_ρ's are used to weigh the factors $\{\mathbf{h}_\rho\}_{\rho=1}^r$. The reason is that each of $\bar{\mathbf{h}}_\rho$ may have a different number of nonzero entries since every source $s_\rho(t)$ affects a different number of sensors. One challenging and instrumental step in implementing D-SMD, not sufficiently addressed in [17], is the systematic selection of pertinent λ_ρ's that ensure recoverability of the factors' support. Cross-validation techniques could be used, see e.g., [21], though the associated complexity of searching on a grid of candidate λ-values increases exponentially with r.

The notion of deflation will be used, see e.g., [9,22], to facilitate the selection of the sparsity-controlling coefficients and reduce its computational complexity. D-SMD is utilized to numerically solve (3) for a single factor ($r = 1$). Let $\hat{\mathbf{h}}_1$ denote the estimate of a single factor obtained via D-SMD. The next step is to remove the impact of

$\hat{\mathbf{h}}_1$ from the sample covariance matrix $\breve{\mathbf{M}}_{x,d}^0 := \hat{\mathbf{M}}_x$ by forming the 'deflated' covariance matrix $\breve{\mathbf{M}}_{x,d}^1 = \breve{\mathbf{M}}_{x,d}^0 - \hat{\mathbf{h}}_1\hat{\mathbf{h}}_1^T$. Then, D-SMD can be employed again ($r = 1$) using the deflated covariance matrix $\breve{\mathbf{M}}_{x,d}^1$ to obtain one more sparse factor $\hat{\mathbf{h}}_2$. Since $\breve{\mathbf{M}}_x$ is multiplied entry-wise with $\mathbf{E}$ in (3), during deflation we can also account for the missing entries by forming $\hat{\mathbf{M}}_{x,d}^1 = \mathbf{E} \odot \hat{\mathbf{M}}_{x,d}^0 - \mathbf{E} \odot \hat{\mathbf{h}}_1\hat{\mathbf{h}}_1^T$. Notice that $\mathbf{E}\odot\breve{\mathbf{M}}_{x,d}^1 = \mathbf{E}\odot\hat{\mathbf{M}}_{x,d}^1$, though $\hat{\mathbf{M}}_{x,d}^1$ is preferable since it contains only the entries that correspond to neighboring sensors.

In general, after deflation step ρ the estimated sparse factor $\hat{\mathbf{h}}_\rho$ is used to evaluate the deflated covariance $\hat{\mathbf{M}}_{x,d}^\rho = \mathbf{E} \odot \hat{\mathbf{M}}_{x,d}^{\rho-1} - \mathbf{E}\odot\hat{\mathbf{h}}_\rho\hat{\mathbf{h}}_\rho^T$. Sensor S_j is updating the entries $\hat{\mathbf{M}}_{x,d}^\rho(j,\mu) = \hat{\mathbf{M}}_{x,d}^{\rho-1}(j,\mu) - \hat{\mathbf{h}}_\rho(j)\hat{\mathbf{h}}_\rho(\mu)$ which requires communication only with the single-hop neighbors in $\mu \in \mathcal{N}_j$. Then, D-SMD for $r = 1$ and covariance $\mathbf{E} \odot \hat{\mathbf{M}}_{x,d}^\rho$ is applied to evaluate the estimate $\hat{\mathbf{h}}_{\rho+1}$. During deflation step $\rho + 1$ a coordinate descent recursion at sensor S_j entails: i) Evaluation of $\{\delta_M^k(j,\mu,\rho+1) = \hat{\mathbf{M}}_{x,d}^\rho(j,\mu)\}_{\mu\in\mathcal{N}_j\cup\{j\}}$ with a computational complexity of the order of $\mathcal{O}(|N_j|)$; ii) forming the coefficients of the polynomials at (4) and (5) with a complexity of $\mathcal{O}(4|N_j|)$; and iii) determining the roots of the third-order polynomials in (4) and (5) whose complexity is fixed and does not depend on the network topology. The complexity at sensor S_j per coordinate cycle is linearly dependent on $|\mathcal{N}_j|$. The previous steps are applied recursively during deflation step $\rho+1$ until the reduction of the cost in (3) between two consecutive cycles drops below a threshold ϵ. This is ensured since D-SMD always converges at least to a stationary point of the cost in (3) (details in [19]). Deflation can be continued until the diagonal entries of $\hat{\mathbf{M}}_{x,d}^{r_d}$ drop below a desired tolerance value, for a given r_d. The deflation-based D-SMD algorithm is tabulated as Algorithm 1.

Algorithm 1 Deflation-Based D-SMD

Sensor S_j initializes $\hat{\mathbf{M}}_{x,d}^0(j,:) = \mathbf{E}(j,:) \odot \hat{\mathbf{M}}_x(j,:)$.
for $\rho = 1,\ldots,r_d$ **do**
  S_j sets $\hat{\mathbf{H}}^0(j,\rho) = \hat{\mathbf{H}}_{\text{ls}}(j,\rho)$, where $\hat{\mathbf{H}}_{\text{ls}}(j,\rho)$ is obtained via D-SMD using $\lambda = 0$ and $\hat{\mathbf{M}}_x \equiv \hat{\mathbf{M}}_{x,d}^{\rho-1}$.
  for $k = 1, 2, \ldots$ **do**
    Each S_j**:** Transmits $\hat{\mathbf{H}}^{k-1}(j,\rho)$ to its neighbors in $\mathcal{N}_j$, and receives $\hat{\mathbf{H}}^{k-1}(\mu,\rho)$ from $\mu \in \mathcal{N}_j$.
    Evaluates $\{\delta_M^k(j,\mu,\rho)\}_{\mu\in\mathcal{N}_j\cup\{j\}}$ and $\{\hat{\mathbf{H}}^k(j,\rho)\}_{\rho=1}^r$.
    If $|\text{Cost}(k) - \text{Cost}(k-1)| \le \epsilon$ then stop.
  end for
  S_j forms deflated entries $\hat{\mathbf{M}}_{x,d}^\rho(j,\mu)$, $\mu \in \mathcal{N}_j \cup \{j\}$.
end for

3.2. Selection of λ

Proper selection of the sparsity-controlling coefficients in D-SMD is critical to determine the support of $\bar{\mathbf{H}}$. A selection scheme is developed here which works well when the nonzero entries of the underlying factors $\{\bar{\mathbf{h}}_\rho\}_{\rho=1}^r$ do not differ significantly in amplitude. This is reasonable to assume since the sensors that sense source $s_\rho(t)$ will be spatially close given the locality of the source, thus the magnitude of the nonzero entries of $\bar{\mathbf{h}}_\rho$ is not expected to vary significantly. The following result, established in [19], is in order.

Proposition 1 *Assume that i) the ensemble* $\mathbf{M}_x$ *is available and used in the D-SMD formulation in* (3) *with* $r = 1$*, while the nonzero entries of each factor have the same magnitude, i.e.,* $\{|\bar{\mathbf{h}}_\rho(j)| = \eta_\rho\}_{\rho=1}^r$ *for* $j \in$ *support*$(\bar{\mathbf{h}}_\rho)$*; and ii) the subset of sensors sensing* $s_\rho(t)$*, form a connected communication subgraph. Further, assume that the underlying factors have non-overlapping supports, while any of the square submatrices of* $\{\mathbf{E} \odot (\bar{\mathbf{h}}_\rho\bar{\mathbf{h}}_\rho^T)\}_{\rho=1}^r$ *have different spectral radius for different* ρ*. Then, the minimizer of* (3) *, say* $\mathbf{h}_o$*, either is an all-zeroes vector if* $\lambda \ge \lambda_{max}$*, or the support of* $\mathbf{h}_o$ *coincides with the support of one of the underlying factors in* $\bar{\mathbf{H}}$ *if* $\lambda < \lambda_{max}$ *and* λ *is sufficiently large, with* λ_{max} *denoting the minimum value of* λ *that results* $\mathbf{h}_o = \mathbf{0}$.

In Prop. 1 it is assumed that the nonzero entries of an underlying factor have the same magnitude. Numerical tests show that Prop. 1 holds even when the nonzero entries of a factor do not have the same amplitude as long as they do not differ significantly. Prop. 1 suggests that an appropriate value for λ can be obtained after starting at $\lambda_{\max}$ (or an upper bound), which gives an all-zeroes solution, and keep decreasing λ until when (3) gives a nonzero solution. Then, the support of the nonzero solution will correspond to the support of one of the factors in $\bar{\mathbf{H}}$. The result of Prop. 1 requires knowledge of the ensemble covariance $\mathbf{M}_x$, though the same result will also hold approximately when a sufficiently large but finite number of data is utilized to form $\hat{\mathbf{M}}_x$. Thus, when λ is appropriately set and there is a sufficiently large number of sensor data the deflation-based D-SMD can recover the support of an underlying sparse covariance factor, or equivalently determine the sensors sensing a specific source.

Note that $\lambda_{\max}$ does not have to be known and an upper bound can be used instead. As shown in [19]: $\lambda_{\max} \le \lambda_{\text{u}} = 1.54 \cdot [\text{tr}(\hat{\mathbf{M}}_x)]^{3/2}$. The *upper bound* λ_u can be computed in a distributed fashion using, e.g., consensus-based techniques (e.g., see [23]) to evaluate $\text{tr}(\hat{\mathbf{M}}_x) = \sum_{j=1}^p \hat{\mathbf{M}}_x(j,j)$. The latter is possible since each sensor S_j has available $\hat{\mathbf{M}}_x(j,j)$. A pertinent λ, using Prop. 1, can be obtained in a distributed fashion as follows. Once sensors evaluate the upper bound λ_{u}, then they create the grid of values $\mathcal{G}_\lambda := \{\lambda_{g,1},\ldots,\lambda_{g,J}\}$ with $\lambda_{g,J} = \lambda_{\text{u}}$ and $\lambda_{g,1} = \epsilon\lambda_{\text{u}}$, with ϵ small, e.g., $\epsilon = 10^{-3}$. Sensors execute the deflation-based D-SMD for decreasing values of λ, starting from λ_{u}. If $\lambda_{g,j}$ corresponds to a grid value for which D-SMD returns an all-zeroes vector, while for the grid value $\lambda_{g,j-1} < \lambda_{g,j}$ there are some sensors that get nonzero values through D-SMD, then using the property of Prop. 1 the λ-selection process can stop at $\lambda_{g,j-1}$. The same λ-selection process can be applied per deflation step.

4. ADAPTIVE IMPLEMENTATION

The deflation-based D-SMD scheme developed in Sec. 3 is a batch scheme in the sense that D-SMD takes place after the acquisition of data and evaluation of $\mathbf{E} \odot \hat{\mathbf{M}}_x$. Such an approach is pertinent when sensors are deployed for exploratory purposes and estimation performance is more important than real-time processing. When it comes to applications such as threat detection, the need for adaptive algorithms that process data online is more prevalent. We build on the deflation D-SMD framework to derive adaptive implementations by properly updating covariance $\hat{\mathbf{M}}_{x,t}$.

Stationary setting: The data covariance $\mathbf{\Sigma}_x$, and the signal covariance $\mathbf{M}_x$, are time-invariant. The sample-average estimate $\hat{\mathbf{\Sigma}}_{x,t} = (t+1)^{-1}\sum_{\tau=0}^t \mathbf{x}_\tau\mathbf{x}_\tau^T$, can be updated recursively as

$$\hat{\mathbf{\Sigma}}_{x,t} = t(t+1)^{-1}\hat{\mathbf{\Sigma}}_{x,t-1} + (t+1)^{-1}\mathbf{x}_t\mathbf{x}_t^T, \tag{6}$$

and $\hat{\mathbf{M}}_{x,t} = \hat{\mathbf{\Sigma}}_{x,t} - \hat{\mathbf{D}}_w$. At time instant t sensor S_j utilizes the updating formula in (6) to refine the corresponding single-hop neighboring entries $\hat{\mathbf{M}}_{x,t}(j,\mu)$ for $\mu \in \mathcal{N}_j \cup \{j\}$. Then, the updated covariance entries are plugged in (3) and the deflation-based D-SMD is applied to estimate the support of $\bar{\mathbf{H}}$ at time instant t. Notice that during deflation step ρ in Alg. 1, an indeterminate number of coordinate descent cycles (index k) are applied until convergence. In an online implementation a small fixed number, say K, of coordinate descent cycles is applied to facilitate online processing. Numerical simulations will show that even for $K = 1$ the adaptive deflation-based D-SMD can identify the informative sensors after a sufficient number of data has been acquired.

Non-stationary setting: In a non-stationary environment, e.g., when the sources are time-varying, covariance $\mathbf{M}_{x,t}$ as well as the support of $\bar{\mathbf{H}}$ will also change with time. The data covariance matrix should be updated in such a way that puts more emphasis on the recent data, while gradually forgets the old ones. One pertinent way to do that is using exponential weighing, i.e.,

$$\hat{\mathbf{\Sigma}}_{x,t} = \sum_{\tau=0}^{t} \beta^{t-\tau}\mathbf{x}_\tau\mathbf{x}_\tau^T = \beta\hat{\mathbf{\Sigma}}_{x,t-1} + \mathbf{x}_t\mathbf{x}_t^T, \qquad (7)$$

and $\hat{\mathbf{M}}_{x,t} = (1-\beta)(1-\beta^{t+1})^{-1}\hat{\mathbf{\Sigma}}_{x,t} - \hat{\mathbf{D}}_w$. The coefficient $\beta \in [0,1]$ denotes the forgetting factor that controls the 'memory' duration. The scaling performed in $\hat{\mathbf{\Sigma}}_{x,t}$ when forming $\hat{\mathbf{M}}_{x,t}$ is done such that $(1-\beta)(1-\beta^{t+1})^{-1}\hat{\mathbf{\Sigma}}_{x,t+1}$ is an unbiased estimate of $\mathbf{\Sigma}_x$ in a stationary setting, i.e., $\mathbb{E}\left[(1-\beta)(1-\beta^{t+1})^{-1} \ \hat{\mathbf{\Sigma}}_{x,t}\right] = \mathbf{\Sigma}_x$ when $\mathbb{E}[\mathbf{x}_\tau\mathbf{x}_\tau^T] = \mathbf{\Sigma}_x$ is time-invariant.

As in the batch processing case, the sparsity-controlling coefficients have to be set appropriately. An initialization step can be incorporated in the adaptive schemes where sensors after acquiring some data $\mathbf{x}_0, \ldots, \mathbf{x}_{L-1}$ they apply the λ-selection scheme outlined in Sec. 3.2. During the operational stage of the adaptive algorithms, the signal covariance estimate $\hat{\mathbf{M}}_{x,t}$ [see eqs. (6) and (7)] changes with time. Such changes in the covariance imply that λ should also be updated using the scheme in Sec. 3.2. The faster the covariance changes, the more frequently λ should be updated. In a stationary setting the updating of λ can stop after a sufficient number of sensor measurements have been acquired and $\hat{\mathbf{M}}_{x,t}$ is converging to $\mathbf{M}_x$.

5. NUMERICAL TESTS

Here the performance of batch and adaptive deflation-based D-SMD is compared with the performance of the batch centralized nonnegative factorization schemes in [16] and [13] abbreviated as NMF-L and NMF-H respectively. The probability of correctly identifying the support of $\bar{\mathbf{H}}$, namely P_D, will be used as a performance metric. Recall that P_D is equal to the probability of identifying the source-informative sensors as well as correctly determining which sensors observe a certain source. P_D was estimated using 500 Monte Carlo runs. We consider the $r = 3$-source and $p = 12$-sensors setting depicted in Fig. 1. The nonzero entries of $\bar{\mathbf{H}} \in \mathbb{R}^{12\times 3}$ are extracted from a Gaussian distribution with mean one and variance 10^{-3}. The λ_ρ coefficients are selected as outlined in Sec. 3.2. Fig. 2 (top) depicts the P_D achieved by the batch and online implementation of D-SMD [using (6)], as well as NMF-H and NMF-L versus time t for a stationary setting. Note that the batch algorithms have to run 'from scratch' at every time instant t using an amount of data $N = t$, whereas the adaptive D-SMD processes recursively only the new data using one coordinate descent cycle ($K = 1$) per t. After a sufficiently large number of data $N = t$, batch D-SMD reaches $P_D = 1$, thus corroborating Prop. 1. This is not the case if sparsity is not exploited ($\lambda = 0$), or when NMF-L and NMF-H are employed since they cannot handle the presence of negative entries in $\bar{\mathbf{H}}$. Online D-SMD is less computationally demanding than the batch D-SMD, though is still capable to achieve $P_D = 1$ at a slower

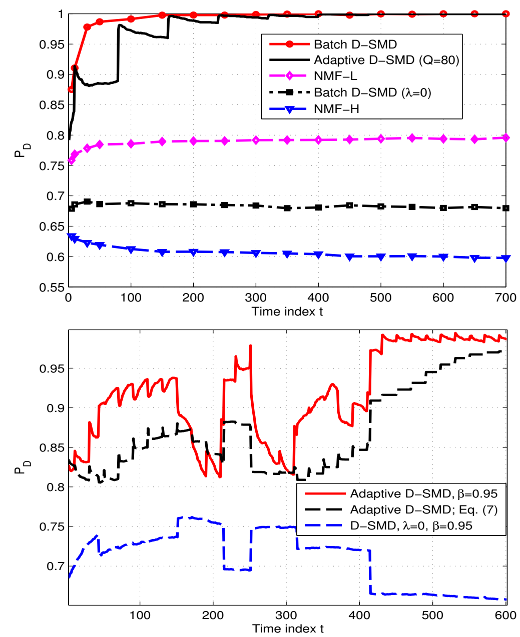

Fig. 2. Probability of detecting the informative sensors vs. time t for a stationary setting (top); and a non-stationary setting (bottom).

rate. The jumps in the P_D curve of online D-SMD correspond to instances during which λ is updated. The λ-readjusting period is set to $Q = 80$. As $t \to \infty$ then $\hat{\mathbf{M}}_{x,t} \to \mathbf{M}_x$, and λ is fixed for $t \geq 400$.

A nonstationary setting is considered next, where the support and the nonzero entries of $\bar{\mathbf{H}}_t \in \mathbb{R}^{12\times 3}$ are time-varying. Let $\bar{\mathbf{H}}_t$ denote the time-varying factor matrix. For $t \leq 20$, $\bar{\mathbf{H}}_t$ remains fixed and set as described earlier. The variations taking place in $\bar{\mathbf{H}}_t$ are as follows: For $t \in [21,200] \cup [301,400]$, $\bar{\mathbf{H}}_t(3,3) = 0.96\cdot\bar{\mathbf{H}}_{t-1}(3,3)$, $\bar{\mathbf{H}}_t(3,1) = \bar{\mathbf{H}}_{t-1}(3,1) + 0.4^{t-T}$ and $\bar{\mathbf{H}}_t(2,3) = \bar{\mathbf{H}}_{t-1}(2,3) + 0.4^{t-T}$, where $T = 20$ when $t \in [21,200]$, and $T = 300$ when $t \in [301,400]$. Then, for $t \in [201,300] \cup [401,600]$ it holds that $\bar{\mathbf{H}}_t(3,3) = \bar{\mathbf{H}}_{t-1}(3,3) + 0.4^{t-T}$, $\bar{\mathbf{H}}_t(3,1) = 0.96\cdot\bar{\mathbf{H}}_{t-1}(3,1)$ and $\bar{\mathbf{H}}_t(2,3) = 0.96\cdot\bar{\mathbf{H}}_{t-1}(2,3)$, where $T = 200$ when $t \in [201,300]$, or $T = 400$ when $t \in [401,600]$. Fig. 2 (bottom) depicts the P_D achieved by i) online D-SMD where the covariance is tracked via (7) with $\beta = 0.95$; ii) same as in i) but no sparsity exploited ($\lambda = 0$); and iii) online D-SMD using (6). As expected since the covariance $\hat{\mathbf{M}}_{x,t}$ is time-varying, the online D-SMD utilizing (7) performs better than the scheme that utilizes (6) since the latter is pertinent for stationary settings. Despite the fact that $\bar{\mathbf{H}}_t$ is keep changing P_D stays above 0.9 most of the time. This is to be contrasted with the performance achieved for $\lambda = 0$ (no sparsity).

6. CONCLUSIONS

The problem of determining sensors that acquire source-informative measurements was formulated as the task of identifying the support of sparse covariance factors. To this end, a sparsity-aware matrix decomposition framework that relies on norm-one regularization was put forth. Block coordinate descent were employed to minimize the associated non-convex costs. A deflation-based mechanism combined with a novel way to select the sparsity-controlling coefficients and adaptive updating of the covariance structure were introduced to reduce complexity and account for time-varying settings.

7. REFERENCES

[1] J. A. Fuemmeler and V. V. Veeravalli, "Energy efficient multi-object tracking in sensor networks," *IEEE Trans. on Signal Processing*, vol. 58, no. 7, pp. 3742–3750, 2010.

[2] V. Gupta, T. Chung, B. Hassibi, and R. Murray, "On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage," *Automatica*, vol. 42, no. 2, pp. 251–260, 2006.

[3] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. on Signal Proc.*, vol. 57, no. 2, pp. 451–462, 2009.

[4] V. Krishnamurthy, M. Maskery, and G. Yin, "Decentralized adaptive filtering algorithms for sensor activation in an unattended ground sensor network," *IEEE Trans. on Signal Proc.*, vol. 56, no. 12, pp. 6086–6101, 2008.

[5] R. S. Blum, Z. Xu, and B. M. Sadler, "Decentralized sensor selection based on sensor observations for energy efficient hypothesis testing," in *Proc. of 43rd Annual Conf. on Info. Sciences and Systems*, Baltimore, MD, 2009, pp. 618–622.

[6] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Info. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[7] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[8] I. Jolliffe, N. Tendafilov, and M. Uddin, "A modified principal component technique based on the Lasso," *Journal of Computational and Graphical Statistics*, vol. 12, no. 3, pp. 531–547, 2003.

[9] A. D'Aspremont, L. E. Ghaoui, M. Jordan, and G. Lanckriet, "A direct formulation for sparse PCA using semidefinite programming," *SIAM Review*, vol. 49, no. 3, pp. 434–448, 207.

[10] I. M. Johnstone and A. Lu, "On consistency and sparsity for principal components analysis in high dimensions," *Journal of the American Statistical Association*, vol. 104, no. 486, pp. 682–693, 2011.

[11] I. D. Schizas and G. B. Giannakis, "Covariance eigenvector sparsity for compression and denoising," *IEEE Trans. on Sig. Processing*, vol. 60, no. 5, pp. 2408–2421, 2012.

[12] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *Journ. of Comp. and Graphical Stat.*, vol. 15, no. 2, pp. 265–286, 2006.

[13] P. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[14] D. D. Lee and H. S. Seung, "Learning the parts of objects by nonnegative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[15] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. of 13th Conference on Advanced Neural Information Processing Systems (NIPS)*, Denver, CO, 2000, pp. 556–562.

[16] C. J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[17] I. D. Schizas, "Distributed informative-sensor determination via sparsity-cognizant matrix decomposition," in *Proc. of the IEEE Workshop on Statistical Signal Processing*, Ann Arbor, MI, 2012, pp. 41–44.

[18] D. P. Bertsekas, *Nonlinear Programming*, Second Edition, Athena Scientific, 2003.

[19] I. D. Schizas, "Distributed informative-sensor identification using sparsity-aware matrix factorization," *IEEE Trans. on Sig. Processing*, submitted Oct. 2012.

[20] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, 1985.

[21] T. Hastie, R. Tibshirani, and D. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Second Edition, Springer, 2009.

[22] B. Yang, "Projection approximation subspace tracking," *IEEE Trans. on Sig. Processing*, vol. 43, no. 1, pp. 95–107, 1995.

[23] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc wsns with noisy links part I: Distributed estimation of deterministic signals," *IEEE Trans. on Sig. Processing*, vol. 56, no. 1, pp. 350–364, 2008.