# DISTRIBUTED SIGNAL SUBSPACE ESTIMATION BASED ON LOCAL GENERALIZED EIGENVECTOR MATRIX INVERSION

*Amin Hassani*, Alexander Bertrand*, Marc Moonen**

* KU Leuven, Dept. of Electrical Engineering-ESAT,
Stadius Center for Dynamical Systems, Signal Processing and Data Analytics,
Address: Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
E-mail: {amin.hassani,alexander.bertrand,marc.moonen}@esat.kuleuven.be

## ABSTRACT

Many array-processing algorithms or applications require the estimation of a target signal subspace, e.g., for source localization or for signal enhancement. In wireless sensor networks, the straightforward estimation of a network-wide signal subspace would require a centralization of all the sensor signals to compute network-wide covariance matrices. In this paper, we present a distributed algorithm for network-wide signal subspace estimation in which such data centralization is avoided. The algorithm relies on a generalized eigenvalue decomposition (GEVD), which allows to estimate a target signal subspace in spatially correlated noise. We show that the network-wide signal subspace can be found from the inversion of the matrices containing the generalized eigenvectors of a pair of reduced-dimension sensor signal covariance matrices at each node. The resulting distributed algorithm reduces the per-node communication and computational cost, while converging to the centralized solution. Numerical simulations reveal a faster convergence speed compared to a previously proposed algorithm.

***Index Terms***— Wireless sensor network (WSN), distributed estimation, signal subspace estimation, generalized eigenvalue decomposition (GEVD)

## 1. INTRODUCTION

Many array-processing algorithms or applications require the estimation of a target signal subspace, e.g., for source localization [1, 2] or for signal enhancement [3], where the performance heavily depends upon how accurately the signal subspace is estimated. We consider the problem of network-wide signal subspace estimation in a fully-connected[1] wireless sensor network (WSN), with multi-sensor nodes and where the noise is possibly spatially correlated. Although the per-node signal subspace can be estimated locally without any signal exchange between nodes, the network-wide signal subspace provides better estimates, since more correlation structure can be exploited (as demonstrated in [5, 6]). Furthermore in some applications such as WSN positioning, the network-wide relative geometry between the nodes has to be captured in the network-wide signal subspace. In a WSN, the straightforward estimation of a network-wide signal subspace would require a centralization of all the sensor signals to compute network-wide covariance matrices. In this paper, we present a distributed algorithm for network-wide signal subspace estimation in which such data centralization is avoided.

The network-wide signal subspace can be estimated using an eigenvalue decomposition (EVD) of the network-wide sensor signal covariance matrix, where part of the eigenvectors directly corresponds to the underlying signal subspace. However, a generalized EVD- (GEVD-) based counterpart delivers a better estimation performance for scenarios with spatially correlated noise, assuming that the noise covariance is either known a-priori or can be estimated, e.g., based on 'noise-only' signal segments [6, 7]. Furthermore as discussed in [5, 6], the GEVD is immune to arbitrary sensor gains at different nodes. Hence we consider a GEVD-based method for the estimation of the network-wide signal subspace.

When a GEVD is employed, the actual network-wide signal subspace can be extracted by the inversion of a matrix containing all the network-wide generalized eigenvectors (GEVCs) [6]. An attempt to estimate this network-wide signal subspace in a distributed fashion was presented earlier in [6], relying only on part of the local GEVCs, i.e., without

This work was carried out at the ESAT Laboratory of KU Leuven, in the frame of KU Leuven Research Council CoE PFV/10/002 (OPTEC), BOF/STG-14-005, the Interuniversity Attractive Poles Programme initiated by the Belgian Science Policy Office IUAP P7/23 'Belgian network on stochastic modeling analysis design and optimization of communication systems' (BESTCOM) 2012-2017, Project FWO nr. G.0763.12 'Wireless Acoustic Sensor Networks for Extended Auditory Communication', Project FWO nr. G.0931.14 'Design of distributed signal processing algorithms and scalable hardware platforms for energy-vs-performance adaptive wireless acoustic sensor networks', and project HANDiCAMS. The project HANDiCAMS acknowledges the financial support of the Future and Emerging Technologies (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 323944. The scientific responsibility is assumed by its authors.

[1] For the sake of an easy exposition, we only consider the case of a fully-connected WSN, but it is noted that all results in this paper can be extended to partially-connected WSNs, using similar techniques as in [4].

the need to compute all local GEVCs at each node. In this paper we propose an alternative distributed algorithm in which each node first estimates the matrix containing *all* the local GEVCs in each iteration, and then applies a local matrix inversion to estimate part of the network-wide signal subspace. Remarkably, while these inverted matrices are based on the *local* GEVCs of per-node *reduced-dimension* covariance matrices, we show that a concatenation of part of these matrices converges to the signal subspace that can be obtained from the inversion of the matrix containing all *network-wide* GEVCs. Moreover, it will be shown via numerical simulations that the proposed method delivers a faster convergence speed, compared to the method presented in [6].

## 2. DATA MODEL AND PROBLEM STATEMENT

We consider a fully-connected WSN with $K$ multi-sensor nodes in which each node $k \in \mathcal{K} = \{1, \ldots, K\}$ collects observations of a complex-valued $M_k$-channel sensor signal $\mathbf{y}_k$, which is modeled as

$$\mathbf{y}_k = \mathbf{A}_k \mathbf{s} + \mathbf{n}_k \tag{1}$$

where $\mathbf{s}$ is an $R$-channel signal containing $R$ target source signals, $\mathbf{A}_k = [\mathbf{a}_{k1} \cdots \mathbf{a}_{kR}]$ is a static (or slowly varying) $M_k \times R$ steering matrix with $\mathbf{a}_{kr}$ $(r = 1, \cdots, R)$ the so-called steering vector (SV) from source $r$ to the sensors of node $k$, and $\mathbf{n}_k$ is the additive noise which can be spatially correlated between nodes. By stacking all $\mathbf{y}_k$'s and $\mathbf{n}_k$'s, we obtain the network-wide $M$-channel ($M = \sum_{k=1}^{K} M_k$) signals $\mathbf{y}$ and $\mathbf{n}$, respectively. Likewise, we define the $M \times R$ matrix $\mathbf{A} = [\mathbf{a}_1 \cdots \mathbf{a}_R]$ as the stacked version of all $\mathbf{A}_k$'s such that

$$\mathbf{y} = \mathbf{A}\mathbf{s} + \mathbf{n}. \tag{2}$$

In this paper we consider the problem of estimating an $R$-dimensional basis of the network-wide signal subspace, i.e., the *range* or the *column space* of the network-wide steering matrix $\mathbf{A}$. We aim to do this in a distributed fashion, i.e., without explicitly constructing network-wide covariance matrices, as this would require centralization of all the sensor signal observations. Instead the nodes will only exchange $R$-channel sensor signal observations, which results in a compression factor of $M_k/R$ at node $k$ (assuming $M_k \geq R$). Here we assume that $R$ is either known or estimated a-priori (e.g., as in [1, 8]). It is noted that, if $R = 1$, the problem reduces to an SV estimation problem, where we estimate $\mathbf{a}_1$ up to a scaling ambiguity.

## 3. CENTRALIZED GEVD-BASED SIGNAL SUBSPACE ESTIMATION

In this section we first explain how the network-wide signal subspace can be estimated by means of a centralized GEVD, i.e., in the case where all the sensor observations are collected in a fusion center. The network-wide sensor signal and noise correlation matrices are defined as $\mathbf{R}_{yy} = E\{\mathbf{y}\mathbf{y}^H\}$ and $\mathbf{R}_{nn} = E\{\mathbf{n}\mathbf{n}^H\}$, respectively, where $E\{\cdot\}$ denotes the expected value operator, and the superscript $H$ denotes the conjugate transpose operator.

Since $\mathbf{y}$ is observable, $\mathbf{R}_{yy}$ can be estimated using sample averaging, and we assume that $\mathbf{R}_{nn}$ is either known a-priori or can be estimated as well via sample averaging from 'noise-only' segments in the sensor signal observations (as explained in [5, 6]).

A GEVD of the ordered matrix pair $(\mathbf{R}_{yy}, \mathbf{R}_{nn})$ is defined as [9]

$$\mathbf{R}_{yy}\hat{\mathbf{X}} = \mathbf{R}_{nn}\hat{\mathbf{X}}\hat{\mathbf{\Lambda}} \tag{3}$$

where $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1 \ldots \hat{\mathbf{x}}_M]$, with $\hat{\mathbf{x}}_m$ the $m$-th GEVC, and $\hat{\mathbf{\Lambda}} = diag\{\hat{\lambda}_1 \ldots \hat{\lambda}_M\}$, with $\hat{\lambda}_m$ the $m$-th largest generalized eigenvalue (GEVL), and where the hat ($\hat{}$) refers to the fact that the centralized estimation is considered. Note that when $\mathbf{R}_{nn}$ is invertible, (3) can be written as a non-symmetric EVD such that

$$\mathbf{R}_{nn}^{-1}\mathbf{R}_{yy} = \hat{\mathbf{X}}\hat{\mathbf{\Lambda}}\hat{\mathbf{X}}^{-1}. \tag{4}$$

In general, the corresponding joint diagonalization of $\mathbf{R}_{yy}$ and $\mathbf{R}_{nn}$ can be written as $\mathbf{R}_{yy} = \hat{\mathbf{Q}}\hat{\mathbf{\Sigma}}\hat{\mathbf{Q}}^H$ and $\mathbf{R}_{nn} = \hat{\mathbf{Q}}\hat{\mathbf{\Gamma}}\hat{\mathbf{Q}}^H$, where $\hat{\mathbf{\Sigma}}$ and $\hat{\mathbf{\Gamma}}$ are diagonal matrices. With this and using (4), it follows that $\hat{\mathbf{Q}} = \hat{\mathbf{X}}^{-H}$, with $\hat{\mathbf{Q}}$ a full-rank $M \times M$ matrix (not necessarily orthogonal). It can be then verified that $\hat{\mathbf{\Sigma}} = \hat{\mathbf{X}}^H\mathbf{R}_{yy}\hat{\mathbf{X}}$ and $\hat{\mathbf{\Gamma}} = \hat{\mathbf{X}}^H\mathbf{R}_{nn}\hat{\mathbf{X}}$ and that $\hat{\mathbf{\Lambda}} = \hat{\mathbf{\Sigma}}(\hat{\mathbf{\Gamma}})^{-1}$. Since the GEVCs are defined up to a scaling, here we assume without loss of generality (w.l.o.g.) that all $\hat{\mathbf{x}}_m$'s are scaled such that $\hat{\mathbf{X}}^H\mathbf{R}_{nn}\hat{\mathbf{X}} = \mathbf{I}_M$. With this we have $\hat{\mathbf{\Gamma}} = \mathbf{I}_M$ and $\hat{\mathbf{\Sigma}} = \hat{\mathbf{\Lambda}}$, and hence the joint diagonalization becomes

$$\mathbf{R}_{yy} = \hat{\mathbf{Q}}\hat{\mathbf{\Lambda}}\hat{\mathbf{Q}}^H, \ \mathbf{R}_{nn} = \hat{\mathbf{Q}}\hat{\mathbf{Q}}^H. \tag{5}$$

Defining $\mathbf{\Pi} = diag\{P_1, ..., P_R\}$ with $P_r$ the power of target source signal $r$, equation (2) implies that $\mathbf{A}\mathbf{\Pi}\mathbf{A}^H = \mathbf{R}_{yy} - \mathbf{R}_{nn}$, where further incorporating (5) yields

$$\mathbf{A}\mathbf{\Pi}\mathbf{A}^H = \hat{\mathbf{Q}}\big(\hat{\mathbf{\Lambda}} - \mathbf{I}_M\big)\hat{\mathbf{Q}}^H. \tag{6}$$

Note that $\hat{\mathbf{Q}}$ is full rank, and that the left-hand side of (6) consists of a positive semi-definite matrix with rank $R$. This requires that $\hat{\mathbf{\Lambda}} - \mathbf{I}_M$ contains only $R$ non-zero diagonal entries. Therefore, the first $R$ GEVLs are larger than one, and all others are equal to one. The first $R$ columns of $\hat{\mathbf{Q}}$ must then span the same $R$-dimensional subspace as the columns of $\mathbf{A}$ and hence fully define the network-wide signal subspace.

We define $\hat{\mathbb{X}}$ and $\hat{\mathbb{Q}}$ as an $M \times R$ matrix containing the first $R$ columns of $\hat{\mathbf{X}}$ and $\hat{\mathbf{Q}}$, respectively. Hence $\hat{\mathbb{Q}}$ spans the network-wide signal subspace. We further define the partitioning

$$\hat{\mathbb{X}} \triangleq \begin{bmatrix} \hat{\mathbb{X}}_1 \\ \vdots \\ \hat{\mathbb{X}}_K \end{bmatrix}, \ \hat{\mathbb{Q}} \triangleq \begin{bmatrix} \hat{\mathbb{Q}}_1 \\ \vdots \\ \hat{\mathbb{Q}}_K \end{bmatrix} \tag{7}$$

where $(.)_k$ is the $M_k \times R$ submatrix that corresponds to node $k$. Moreover we define the $R \times R$ diagonal matrix $\hat{\Lambda} = diag\{\hat{\lambda}_1 \dots \hat{\lambda}_R\}$.

## 4. DISTRIBUTED GEVD-BASED SIGNAL SUBSPACE ESTIMATION

So far we have considered the GEVD-based estimation of $\hat{\mathbb{Q}}$ via data centralization. However in a typical WSN, a node $k$ only observes its own $M_k$-channel sensor signal $\mathbf{y}_k$ and hence can only estimate an $M_k \times M_k$ submatrix of $\mathbf{R}_{yy}$ and $\mathbf{R}_{nn}$. In this section we explain how the nodes can efficiently cooperate to estimate the columns of the network-wide $\hat{\mathbb{Q}}$, without constructing the network-wide correlation matrices $\mathbf{R}_{yy}$ and $\mathbf{R}_{nn}$. The algorithm derivation starts from a distributed GEVD algorithm, referred to as the distributed adaptive covariance-matrix generalized eigenvector estimation (DACGEE) algorithm [7]. We will then show that the network-wide $\hat{\mathbb{Q}}$ can be inferred from the inversion of compressed GEVC matrices that are computed at the individual nodes in the DACGEE algorithm. This is remarkable, since DACGEE only computes $\hat{\mathbb{X}}$, whereas the inversion of the full matrix $\hat{\mathbf{X}}$ is at first sight required to find $\hat{\mathbb{Q}}$.

### 4.1. The fully-connected DACGEE algorithm

The DACGEE algorithm [7] estimates and updates (in a distributed and block-adaptive fashion) the matrix $\hat{\mathbb{X}}$, where the communication cost of node $k$ is reduced by a factor $M_k/R$ (compared to the centralized case and assuming $M_k \geq R$). In DACGEE, each node $k$ only updates the $M_k \times R$ submatrix $\mathbb{X}_k^i$, which is the estimate of $\hat{\mathbb{X}}_k$ at iteration $i$. We define the $M \times R$ matrix $\mathbb{X}^i$ as the estimate of $\hat{\mathbb{X}}$, which is constructed by concatenating all submatrices $\mathbb{X}_k^i, \forall k \in \mathcal{K}$, i.e., $\mathbb{X}^i \triangleq [\mathbb{X}_1^{i\,T} \dots \mathbb{X}_K^{i\,T}]^T$. Hence the objective of DACGEE is to obtain $\lim_{i\to\infty} \mathbb{X}^i = \hat{\mathbb{X}}$.

Each node $k$, at iteration $i$, uses $\mathbb{X}_k^i$ to compress $N$ observations of its $M_k$-channel sensor signal into observations of the $R$-channel signal

$$\mathbf{z}_k^i = \mathbb{X}_k^{i\,H} \mathbf{y}_k \tag{8}$$

and then broadcasts these $N$ observations of $\mathbf{z}_k^i$ to all other nodes. Therefore an updating node $q$ observes the following $P_q$-channel sensor signal ($P_q \triangleq M_q + R(K-1)$) and estimates the corresponding covariance matrix:

$$\widetilde{\mathbf{y}}_q^i = \begin{bmatrix} \mathbf{y}_q \\ \mathbf{z}_{-q}^i \end{bmatrix} \Rightarrow \mathbf{R}_{\tilde{y}_q\tilde{y}_q}^i = E\{\widetilde{\mathbf{y}}_q^i \widetilde{\mathbf{y}}_q^{i\,H}\} \tag{9}$$

where $\mathbf{z}_{-q}^i = [\mathbf{z}_1^{i\,T} \dots \mathbf{z}_{(q-1)}^{i\,T} \mathbf{z}_{(q+1)}^{i\,T} \dots \mathbf{z}_K^{i\,T}]^T$. In a similar way, we can define $\mathbf{R}_{\tilde{n}_q\tilde{n}_q}^i$, which can be estimated from $\widetilde{\mathbf{y}}_q^i$ during 'noise-only' segments[2]. The updating node $q$ can then compute the GEVD of the reduced-dimension GEVD of $(\mathbf{R}_{\tilde{y}_q\tilde{y}_q}^i, \mathbf{R}_{\tilde{n}_q\tilde{n}_q}^i)$ as (compare with (3))

$$\mathbf{R}_{\tilde{y}_q\tilde{y}_q}^i \widetilde{\mathbf{X}}_q^i = \mathbf{R}_{\tilde{n}_q\tilde{n}_q}^i \widetilde{\mathbf{X}}_q^i \widetilde{\mathbf{\Lambda}}_q^i \;\text{ s.t. }\; \widetilde{\mathbf{X}}_q^{i\,H} \mathbf{R}_{\tilde{n}_q\tilde{n}_q}^i \widetilde{\mathbf{X}}_q^i = \mathbf{I}_{P_q} \tag{10}$$

where the $P_q \times P_q$ matrix $\widetilde{\mathbf{X}}_q^i$ and the $P_q \times P_q$ diagonal matrix $\widetilde{\mathbf{\Lambda}}_q^i$ contain all $P_q$ local GEVCs and GEVLs of $(\mathbf{R}_{\tilde{y}_q\tilde{y}_q}^i, \mathbf{R}_{\tilde{n}_q\tilde{n}_q}^i)$, respectively. We now define the $P_q \times R$ matrix $\widetilde{\mathbb{X}}_q^i$ containing the first $R$ columns of $\widetilde{\mathbf{X}}_q^i$. Moreover we define the $R \times R$ diagonal matrix $\widetilde{\mathbb{\Lambda}}_q^i$ as the $R \times R$ submatrix of $\widetilde{\mathbf{\Lambda}}_q^i$ containing its first $R$ diagonal entries. In DACGEE, node $q$ then uses the first $M_q$ rows of $\widetilde{\mathbb{X}}_q^i$ as $\mathbb{X}_q^{i+1}$. In the next iteration, the updating node $q$ is changed, and a new block of $N$ sensor signal observations is used. Note that the latter means that the iterations are spread out over different signal segments in a block-recursive fashion.

Using the above updating procedure, it can be shown that (up to estimation errors in the covariance matrices) [7, 10].

$$\lim_{i\to\infty} \mathbb{X}^i = \hat{\mathbb{X}} \;\text{ and }\; \lim_{i\to\infty} \widetilde{\mathbb{\Lambda}}_k^i = \hat{\mathbb{\Lambda}}, \forall k \in \mathcal{K} \tag{11}$$

### 4.2. Extracting the signal subspace

Similar to $\mathbb{X}^i$ we define $\mathbb{Q}^i \triangleq [\mathbb{Q}_1^{i\,T} \dots \mathbb{Q}_K^{i\,T}]^T$ as the estimation of $\hat{\mathbb{Q}}$ at iteration $i$. Then the objective is to estimate $\mathbb{Q}^i$ in a distributed fashion such that it converges to the network-wide $\hat{\mathbb{Q}}$. We first briefly review the correlation-based method presented in [6].

#### *4.2.1. Correlation-based method [6]*

Let $\hat{\mathbf{z}} = \hat{\mathbb{X}}^H \mathbf{y} = \sum_{k\in\mathcal{K}} \hat{\mathbb{X}}_k^H \mathbf{y}_k$. Note that, due to the convergence of the DACGEE algorithm, $\hat{\mathbf{z}}$ will be equal to $\hat{\mathbf{z}} = \sum_{k\in\mathcal{K}} \mathbf{z}_k^i$ for $i \to \infty$ (see (8)). It has been shown in [6] that $\hat{\mathbf{R}}_{y\hat{z}} = E\{\mathbf{y}\hat{\mathbf{z}}^H\} = \hat{\mathbb{Q}}\hat{\mathbb{\Lambda}}$. Since $\hat{\mathbb{\Lambda}}$ only scales the columns of $\hat{\mathbb{Q}}$, $\hat{\mathbf{R}}_{y\hat{z}}$ defines the network-wide signal subspace. In iteration $i$, each node computes $\mathbf{R}_{y_k z}^i = E\{\mathbf{y}_k \mathbf{z}^{i\,H}\}$ where $\mathbf{z}^i = \sum_{k\in\mathcal{K}} \mathbf{z}_k^i$. The stacked matrix of all $\mathbf{R}_{y_k z}^i$'s, i.e., $\mathbf{R}_{yz}^i$, has been shown to converge to $\hat{\mathbf{R}}_{y\hat{z}}$, i.e., $\lim_{i\to\infty} \mathbf{R}_{yz}^i = \hat{\mathbf{R}}_{y\hat{z}}$.

#### *4.2.2. Inversion-based method*

In this section we propose a new method to estimate the network-wide signal subspace $\hat{\mathbb{Q}}$, based on the inversion of the matrix $\widetilde{\mathbf{X}}_k^i$ at each node $k$. $\widetilde{\mathbf{X}}_k^i$ is computed by the DACGEE algorithm, although DACGEE for its own convergence only needs to compute the matrix $\widetilde{\mathbb{X}}_q^i$ at the updating node $q$. Therefore, similar to (5), we can write

$$\mathbf{R}_{\tilde{y}_k\tilde{y}_k}^i = \widetilde{\mathbf{Q}}_k^i \widetilde{\mathbf{\Lambda}}_k^i \widetilde{\mathbf{Q}}_k^{i\,H},\; \mathbf{R}_{\tilde{n}_k\tilde{n}_k}^i = \widetilde{\mathbf{Q}}_k^i \widetilde{\mathbf{Q}}_k^{i\,H} \tag{12}$$

where $\widetilde{\mathbf{Q}}_k^i = \widetilde{\mathbf{X}}_k^{i\,-H}$. Node $k$ then extracts $\mathbb{Q}_k^i$ as

$$\mathbb{Q}_k^i = [\mathbf{I}_{M_k}\; \mathbf{0}]\, \widetilde{\mathbf{Q}}_k^i \begin{bmatrix} \mathbf{I}_R \\ \mathbf{0} \end{bmatrix}. \tag{13}$$

[2] If the network-wide $\mathbf{R}_{nn}$ is known a-priori , one can compute $\mathbf{R}_{\tilde{n}_q\tilde{n}_q}^i$ directly by means of the compression matrices from the other nodes, as in (24).

**Table 1**. Distributed signal subspace estimation based on local GEVC matrix inversion in a fully-connected WSN

1. Set $i \leftarrow 0$, $q \leftarrow 1$, and initialize all $\mathbb{X}_k^0, \forall\, k \in \mathcal{K}$, with random entries.
2. Each node $k \in \mathcal{K}\backslash q$ broadcasts $N$ new compressed observations of
$$\mathbf{z}_k^i[j] = \mathbb{X}_k^{i\,H}\mathbf{y}_k[iN+j], \quad j = 1 \ldots N \tag{14}$$
where [.] denotes a sample index and where $N$ is sufficiently large such that it includes both 'signal+noise' and 'noise-only' samples.
3. At node $q$:
   (a) Estimate $\mathbf{R}^i_{\tilde{y}_q\tilde{y}_q}$ and $\mathbf{R}^i_{\tilde{n}_q\tilde{n}_q}$ via sample averaging.
   (b) Compute local GEVCs $\widetilde{\mathbf{X}}_q^{i+1}$ from the GEVD of $(\mathbf{R}^i_{\tilde{y}_q\tilde{y}_q}, \mathbf{R}^i_{\tilde{n}_q\tilde{n}_q})$ and then $\widetilde{\mathbf{Q}}_q^{i+1} = \big(\widetilde{\mathbf{X}}_q^{i+1}\big)^{-H}$.
   (c) Partition $\widetilde{\mathbf{Q}}_q^{i+1}$ as
$$\mathbb{Q}_q^{i+1} = \big[\mathbf{I}_{M_k}\ \mathbf{0}\big]\,\widetilde{\mathbf{Q}}_q^{i+1}\begin{bmatrix}\mathbf{I}_R\\ \mathbf{0}\end{bmatrix}. \tag{15}$$
   (d) Partition $\widetilde{\mathbf{X}}_q^{i+1}$ as
$$\mathbb{X}_q^{i+1} = \big[\mathbf{I}_{M_k}\ \mathbf{0}\big]\,\widetilde{\mathbf{X}}_q^{i+1}\begin{bmatrix}\mathbf{I}_R\\ \mathbf{0}\end{bmatrix} \tag{16}$$
$$\mathbf{G}_{-q} = \big[\mathbf{0}\ \mathbf{I}_{R(K-1)}\big]\,\widetilde{\mathbf{X}}_q^{i+1}\begin{bmatrix}\mathbf{I}_R\\ \mathbf{0}\end{bmatrix}. \tag{17}$$
   (e) Broadcast $\mathbf{G}_{-q} \triangleq \Big[\mathbf{G}_1^T\ \ldots\ \mathbf{G}_{q-1}^T\,\mathbf{G}_{q+1}^T\ \ldots\ \mathbf{G}_K^T\Big]^T$ and $\mathbf{z}_q^i[j] = \mathbb{X}_q^{i+1\,H}\mathbf{y}_q[iN+j]$ to all other nodes.
4. Each node $k \in \mathcal{K}\backslash\{q\}$ updates its compressor as $\mathbb{X}_k^{i+1} = \mathbb{X}_k^i\mathbf{G}_k$.
5. Each node $k \in \mathcal{K}\backslash\{q\}$ updates its $\mathbb{Q}_k^{i+1}$ similar to steps 3a-3c.
6. $i \leftarrow i+1$ and $q \leftarrow (q \mod K)+1$ and return to step 2.

The stacked matrix $\mathbb{Q}^i$ will be shown below to converge to $\hat{\mathbb{Q}}$, which is remarkable since $\hat{\mathbb{Q}}$ is part of the inverse of the full matrix $\hat{\mathbf{X}}$, whereas the $\mathbb{Q}_k^i$'s are extracted from the inverses of the compressed matrices $\widetilde{\mathbf{X}}_k^i$. The resulting distributed algorithm is provided in Table 1. It will be shown in Section 5 that the new method significantly outperforms the correlation-based method described in Section 4.2.1 in terms of convergence speed. This improvement is indeed achieved at the cost of more complex computations, since each node $k \in \mathcal{K}$ at iteration $i$ must compute all the $P_k$ GEVCs in $\widetilde{\mathbf{X}}_k^i$ from the GEVD of the reduced-dimension correlation matrices $(\mathbf{R}^i_{\tilde{y}_k\tilde{y}_k}, \mathbf{R}^i_{\tilde{n}_k\tilde{n}_k})$ as well as a matrix inverse.

**Theorem I:** *In the algorithm defined in Table 1, the network-wide signal subspace $\mathbb{Q}^i$ converges to the centralized network-wide signal subspace $\hat{\mathbb{Q}}$, i.e.,* $\lim_{i\to\infty}\mathbb{Q}^i = \hat{\mathbb{Q}}$ *or in particular,* $\lim_{i\to\infty}\mathbb{Q}_k^i = \hat{\mathbb{Q}}_k, \forall k \in \mathcal{K}$.

*Proof:* Replacing $\mathbf{R}^i_{\tilde{y}_k\tilde{y}_k}$ in (10) by (12), and considering the fact that $\widetilde{\mathbf{Q}}_k^{i\,H}\widetilde{\mathbf{X}}_k^i = \mathbf{I}_{P_k}$, equation (10) can be rewritten as
$$\widetilde{\mathbf{Q}}_k^i\widetilde{\mathbf{\Lambda}}_k^i = \mathbf{R}^i_{\tilde{n}_k\tilde{n}_k}\widetilde{\mathbf{X}}_k^i\widetilde{\mathbf{\Lambda}}_k^i \tag{18}$$
where taking the first $R$ columns yields
$$\widetilde{\mathbb{Q}}_k^i = \mathbf{R}^i_{\tilde{n}_k\tilde{n}_k}\widetilde{\mathbb{X}}_k^i. \tag{19}$$
We define the $M \times P_k$ compressor matrix $\mathbf{C}_k^i$ as
$$\mathbf{C}_k^i = \left[\begin{array}{c|c|c}\mathbf{0} & \mathbf{B}^i_{<k} & \mathbf{0}\\ \hline \mathbf{I}_{M_k} & \mathbf{0} & \mathbf{0}\\ \hline \mathbf{0} & \mathbf{0} & \mathbf{B}^i_{>k}\end{array}\right] \tag{20}$$
where $\mathbf{0}$ is an all-zero matrix of proper dimension, and where[3]
$$\mathbf{B}^i_{<k} = \text{Blkdiag}(\mathbb{X}_1^i, \ldots, \mathbb{X}_{(k-1)}^i) \tag{21}$$
$$\mathbf{B}^i_{>k} = \text{Blkdiag}(\mathbb{X}_{(k+1)}^i, \ldots, \mathbb{X}_K^i). \tag{22}$$
With this, and using (8)-(9), we can link the reduced $P_k$-dimensional correlation matrices $(\mathbf{R}_{\tilde{y}_k\tilde{y}_k}, \mathbf{R}_{\tilde{n}_k\tilde{n}_k})$ with the full $M$-dimensional correlation matrices $(\mathbf{R}_{yy}, \mathbf{R}_{nn})$, i.e.,
$$\mathbf{R}^i_{\tilde{y}_k\tilde{y}_k} = \mathbf{C}_k^{i\,H}\mathbf{R}_{yy}\mathbf{C}_k^i \tag{23}$$
$$\mathbf{R}^i_{\tilde{n}_k\tilde{n}_k} = \mathbf{C}_k^{i\,H}\mathbf{R}_{nn}\mathbf{C}_k^i. \tag{24}$$
We now assume that $i \to \infty$, i.e., when DACGEE has converged and hence at all nodes $k \in \mathcal{K}$ we have that $\mathbb{X}_k^i = \hat{\mathbb{X}}_k$ and that $\widetilde{\mathbb{\Lambda}}_k^i = \hat{\mathbb{\Lambda}}$ (see (11)). Moreover since after the convergence at all nodes $k \in \mathcal{K}$ we have that $\mathbb{X}_k^{i+1} = \mathbb{X}_k^i$, this requires that [10]
$$\widetilde{\mathbb{X}}_k^{i+1} = [\mathbb{X}_k^{i\,T}\,\mathbf{I}_R \ldots \mathbf{I}_R]^T. \tag{25}$$
Let $\hat{\mathbf{C}}_k$ be the compressor matrix $\mathbf{C}_k^i$ after the convergence of the DACGEE algorithm, i.e., when in (20)-(22) $\mathbb{X}_k^i = \hat{\mathbb{X}}_k, \forall k \in \mathcal{K}$. Considering this together with (24), we can rewrite (19) as
$$\lim_{i\to\infty}\widetilde{\mathbb{Q}}_k^i = \lim_{i\to\infty}\hat{\mathbf{C}}_k^H\mathbf{R}_{nn}\hat{\mathbf{C}}_k\widetilde{\mathbb{X}}_k^i \tag{26}$$
Based on (25), after convergence we have that $\hat{\mathbb{X}} = \hat{\mathbf{C}}_k\widetilde{\mathbb{X}}_k^i$. Now plugging (5) into (26), and considering the fact that $\hat{\mathbb{Q}}^H\hat{\mathbb{X}} = \mathbf{I}_R$, it follows that
$$\lim_{i\to\infty}\widetilde{\mathbb{Q}}_k^i = \lim_{i\to\infty}\hat{\mathbf{C}}_k^H\hat{\mathbf{Q}}\hat{\mathbf{Q}}^H\hat{\mathbb{X}} \tag{27}$$
$$= \lim_{i\to\infty}\hat{\mathbf{C}}_k^H\hat{\mathbf{Q}}\begin{bmatrix}\mathbf{I}_R\\ \mathbf{0}\end{bmatrix} \tag{28}$$
$$= \lim_{i\to\infty}\hat{\mathbf{C}}_k^H\hat{\mathbb{Q}}. \tag{29}$$
Selecting the first $M_k$ rows of (28), we obtain
$$\lim_{i\to\infty}\mathbb{Q}_k^i = \hat{\mathbb{Q}}_k \tag{30}$$
which proves the theorem.

[3] It is noted that the diagonal blocks are not square here, i.e., in this case Blkdiag(.) is not truly a block-diagonal matrix in the strict sense.

## 5. SIMULATION RESULTS

In this section, we compare the performance of the proposed inversion-based method with the correlation-based method previously proposed in [6]. The simulations compare the two methods with the 'centralized' and the 'isolated' cases. The latter corresponds to the case where each node only observes its own $M_k$-channel sensor signal and hence does not cooperate with other nodes. In all Monte-Carlo (MC) runs, $K = 10$ and $M_k = 15, \forall k \in \mathcal{K}$. Out of 10 localized sources in each MC scenario, $R$ of them are considered as the target sources (with an on-off behavior) and the remaining $(10-R)$ sources are treated as noise sources (continuously active). The network-wide noise signal $\mathbf{n}$ (see (2)) can be described as $\mathbf{n} = \mathbf{J}\mathbf{t} + \mathbf{v}$ where $\mathbf{J}$ is the steering matrix corresponding to the noise sources, $\mathbf{t}$ contains the $(10-R)$ noise source signals, and $\mathbf{v}$ models the spatially uncorrelated noise signals. The network-wide steering matrices $\mathbf{A}$ and $\mathbf{J}$ are static matrices with dimensions $150 \times R$ and $150 \times (10-R)$, respectively, in which the entries are drawn from a uniform distribution over the interval $[-0.5; 0.5]$. $\mathbf{s}$ and $\mathbf{t}$ are $R$-channel and $(10-R)$-channel stochastic source signals from which the observations are independently drawn from a uniform distribution over the interval $[-0.5; 0.5]$. Moreover, $\mathbf{v}$ is a 150-channel stochastic signal from which the observations are independently drawn from a uniform distribution over the interval $[-\sqrt{0.1}/2; \sqrt{0.1}/2]$. In each MC run, a different simulation scenario is created and an average of the largest canonical angle (principal angle) between the true steering matrix $\mathbf{A}_k$ and its corresponding signal subspace estimate over all nodes is considered as a performance measure.

Fig. 1 illustrates the results for the cases where $R = 2$ and $R = 4$, averaged over 200 MC runs. This figure clearly shows that: 1) The cooperative estimation (either centralized or distributed) significantly outperforms the one achieved by the isolated estimation in terms of the estimation accuracy; 2) The estimate obtained with the proposed distributed algorithm converges to the estimate obtained with the centralized estimation; 3) The distributed signal subspace estimation obtained with the proposed method converges significantly faster than the correlation-based method presented in [6]; 4) While the value of $R$ remarkably affects the convergence speed of the correlation-based method, it has almost no effect on the convergence speed of the the inversion-based method.

## 6. CONCLUSION

In this paper, we have proposed a distributed algorithm for network-wide signal subspace estimation in a fully-connected WSN. We have applied a GEVD-based method, which allows to better estimate a target signal subspace in spatially correlated noise. The algorithm first applies the DACGEE algorithm to estimate the matrix containing all local GEVCs at each node, and then inverts this matrix to obtain a basis of the corresponding part of the network-wide signal subspace. It has been demonstrated that the network-wide signal subspace estimate obtained with the proposed distributed algorithm converges to the estimate obtained with the centralized network-wide signal subspace, with a faster convergence speed compared to the correlation-based method.

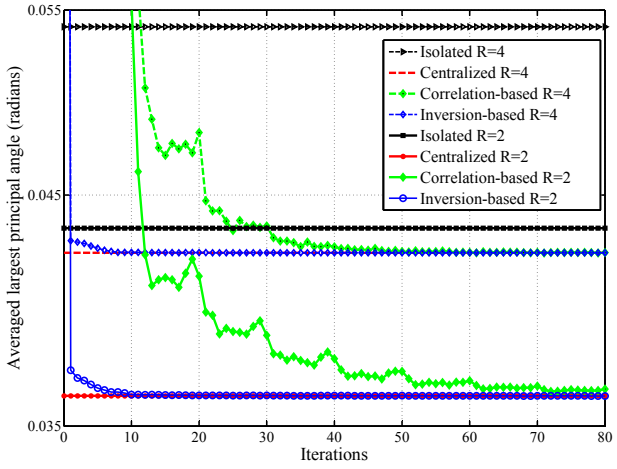


**Fig. 1**. Multiple target sources (signal subspace estimation)

## REFERENCES


[1] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *Signal Processing Magazine, IEEE*, vol. 13, no. 4, pp. 67–94, 1996.

[2] S. Haykin and K.J Liu, *Handbook on array processing and sensor networks*, IEEE, Hoboken, N.J. Wiley, Piscataway, NJ, 2009.

[3] M. Brandstein and D. Ward, *Microphone Arrays: Signal Processing Techniques and Applications*, Digital Signal Processing - Springer-Verlag. Springer, 2001.

[4] A. Bertrand and M. Moonen, "Distributed adaptive estimation of covariance matrix eigenvectors in wireless sensor networks with application to distributed PCA," *Signal Processing*, vol. 104, pp. 120–135, 2014.

[5] A. Hassani, A. Bertrand, and M. Moonen, "Cooperative integrated noise reduction and node-specific direction-of-arrival estimation in a fully connected wireless acoustic sensor network," *Signal Processing*, vol. 107, pp. 68–81, Feb. 2015.

[6] A. Hassani, A. Bertrand, and M. Moonen, "Distributed GEVD-based signal subspace estimation in a fully-connected wireless sensor network," in *Proc. European Signal Processing Conference (EUSIPCO)*, Lisbon, Portugal, Sept. 2014.

[7] A. Bertrand and M. Moonen, "Distributed adaptive generalized eigenvector estimation of a sensor signal covariance matrix pair in a fully-connected sensor network," *Signal Processing*, vol. 106, pp. 209–214, Jan. 2015.

[8] A. Scaglione, R. Pagliari, and H. Krim, "The decentralized estimation of the sample covariance," in *Signals, Systems and Computers, 2008 42nd Asilomar Conference on*, Oct 2008, pp. 1722–1726.

[9] C. F. Van Loan G. H. Golub, *Matrix Computations, 3rd ed.*, Baltimore, MD: John Hopkins Univ. Press, 1996.

[10] A. Hassani, A. Bertrand, and M. Moonen, "GEVD-based low-rank approximation for distributed adaptive node-specific signal estimation in a fully-connected wireless sensor network," in *Internal Report KU Leuven ESAT-STADIUS*, 2015, available online at http://homes.esat.kuleuven.be/ sistawww/cgi-bin/pub.pl.