# Topology-independent GEVD-based distributed adaptive node-specific signal estimation in ad-hoc wireless acoustic sensor networks

Paul Didier, Toon van Waterschoot, Marc Moonen
*KU Leuven, Department of Electrical Engineering (ESAT),*
*STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics, Belgium*
Email: paul.didier, toon.vanwaterschoot, marc.moonen@esat.kuleuven.be

***Abstract*—A low-rank approximation-based version of the topology-independent distributed adaptive node-specific signal estimation (TI-DANSE) algorithm is introduced, using a generalized eigenvalue decomposition (GEVD) for application in ad-hoc wireless acoustic sensor networks. This TI-GEVD-DANSE algorithm as well as the original TI-DANSE algorithm exhibit a non-strict convergence, which can lead to numerical instability over time, particularly in scenarios where the estimation of accurate spatial covariance matrices is challenging. An adaptive filter coefficient normalization strategy is proposed to mitigate this issue and enable the stable performance of TI-(GEVD-)DANSE. The method is validated in numerical simulations including dynamic acoustic scenarios, demonstrating the importance of the additional normalization.**

***Index Terms*—wireless acoustic sensor networks, distributed signal estimation, topology-independent, low-rank approximation**

## I. Introduction

In recent years, the ever-increasing ubiquity of multi-microphone devices capable of exchanging and processing acoustic signals has motivated the development of distributed audio signal processing algorithms. As opposed to traditional localized microphone arrays, distributed systems do not rely on a fusion center; they instead leverage the computing capacities of each device (i.e., each node) in wireless acoustic sensor networks (WASNs). Distributed systems are typically able to use signals spanning a large acoustic area while maintaining a high degree of flexibility in their physical design [1], [2].

The tasks of distributed algorithms may be categorized depending on the estimated quantity [3]. Here, the focus is set on signal estimation for applications that require the retrieval of entire (possibly multichannel) signals of interest which may be node-specific and non-stationary, e.g., for noise reduction in speech enhancement tasks. In a fully connected WASN, one may let all nodes transmit all their local sensor signals to all other nodes, effectively corresponding to a centralized case. For obvious reasons, this strategy suffers from an inefficient usage of communication bandwidth. Instead, it has been shown that the nodes can exchange fused versions of their local sensor signals while retaining the same performance as if they were transmitting all of their sensor signals. This is a core idea in the distributed adaptive node-specific signal estimation (DANSE) algorithm [4], which is considered in this paper.

The DANSE algorithm is iterative and converges towards the centralized linear minimum mean square error (LMMSE) optimum. It operates in fully connected WASNs, where every node can communicate with every other node. However, in many practical applications, there is no guarantee that the network topology will be fully connected or be static over time (e.g., due to link failures). A solution to this is provided by the topology-independent (TI) DANSE algorithm (TI-DANSE) [5], which allows a new tree to be pruned from the ad-hoc topology at any algorithm iteration while retaining convergence, thus being robust to dynamic topologies.

The DANSE and TI-DANSE algorithms rely at their core on the computation of a multichannel Wiener filter (MWF). A significant performance improvement can be obtained when the number of latent target sources, or an estimate thereof, is known a priori – a reasonable assumption in many applications. The rank of the desired signal spatial covariance matrix (SCM) can be set equal to that number via a generalized eigenvalue decomposition (GEVD). It has been shown that such GEVD-based low-rank approximation of the MWF (GEVD-MWF) indeed outperforms the unconstrained MWF, particularly in challenging signal-to-noise ratio conditions [6]. Although the concept has been applied to DANSE in [7], leading to the GEVD-DANSE algorithm, its potential for TI-DANSE has remained unexplored.

The contribution of this paper is two-fold. First, we introduce the TI-GEVD-DANSE algorithm, which can operate in any topology and incorporates the advantages of a GEVD-MWF. Second, we address the observed non-strict convergence of the TI-GEVD-DANSE algorithm, also observed in the original TI-DANSE algorithm, which allows the LMMSE optimum to be reached even though the filters themselves diverge. To this end, we propose a filter coefficient normalization strategy which stabilizes the behavior of TI-(GEVD-)DANSE even when signal statistics are estimated on the fly.

This research was carried out at the ESAT Laboratory of KU Leuven, in the frame of Research Council KU Leuven C14-21-0075 "A holistic approach to the design of integrated and distributed digital signal processing algorithms for audio and speech communication devices", and was supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 956369: 'Service-Oriented Ubiquitous Network-Driven Sound — SOUNDS'. The scientific responsibility is assumed by its authors. This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information.

This paper is organized as follows. The problem statement is given in Section II, defining the signal model and the centralized MWF solution. The TI-DANSE algorithm is reviewed in Section III. The TI-GEVD-DANSE algorithm is presented in Section IV. The proposed normalization strategy addressing the stability of TI-(GEVD-)DANSE is presented in Section V. Numerical experiments are provided in Section VI. Finally, conclusions are given in Section VII.

## II. Problem Statement

We consider a WASN consisting of $K$ nodes, where node $k$ has $M_k$ sensors ($k \in \mathcal{K} := \{1, ..., K\}$) such that the total number of sensors is $M = \sum_{k\in\mathcal{K}} M_k$. The signals are assumed to be complex-valued to allow representation of, e.g., processing in a particular bin of a filter bank. The acoustic scenario includes $S$ desired sources whose latent signals at time $t$ are grouped in $\hat{\mathbf{s}}[t] \in \mathbb{C}^{S\times 1}$, and $S_n$ noise sources with latent signals $\hat{\mathbf{n}}[t] \in \mathbb{C}^{S_n\times 1}$. The sensor signals at all nodes are stacked in $\mathbf{y}[t] \in \mathbb{C}^{M\times 1}$, which is modeled as:

$$\mathbf{y}[t] = \mathbf{s}[t] + \mathbf{n}[t] = \mathbf{A}\hat{\mathbf{s}}[t] + \mathbf{B}\hat{\mathbf{n}}[t] + \mathbf{q}[t], \tag{1}$$

where $\mathbf{n}[t]$ is the noise component, $\mathbf{A} \in \mathbb{C}^{M\times S}$ and $\mathbf{B} \in \mathbb{C}^{M\times S_n}$ are the steering matrices of the desired signal and the noise, respectively, and $\mathbf{q}[t]$ is uncorrelated (thermal) noise. Time indices $[t]$ are omitted from here on for conciseness. Each vector in Eq.(1) can be partitioned as $\mathbf{y} = [\mathbf{y}_1^T \dots \mathbf{y}_K^T]^T$, where $\cdot^T$ denotes the transpose operator and $\mathbf{y}_k \in \mathbb{C}^{M_k\times 1}$ are the sensor signals of node $k$. Each node strives to estimate its own $J$-channel desired signal $\mathbf{d}_k = \mathbf{E}_{kk}^T\mathbf{s}_k$ where $\mathbf{E}_{kk}$ is an $M_k \times J$ selection matrix. A node-specific LMMSE estimation problem is considered, where the optimal filter matrix $\mathbf{W}_k \in \mathbb{C}^{M\times J}$ used to estimate $\mathbf{d}_k$ from $\mathbf{y}$ is defined as:

$$\mathbf{W}_k = \arg\min_{\mathbf{W}} E\left\{\left\|\mathbf{d}_k - \mathbf{W}^H\mathbf{y}\right\|_2^2\right\}, \tag{2}$$

where $E\{\cdot\}$, $\|\cdot\|_2$, and $\cdot^H$ denote the expected value operator, the Euclidean norm, and the Hermitian operator, respectively. The solution of Eq.(2) is the well-known MWF:

$$\mathbf{W}_k = (\mathbf{R}_{\mathbf{yy}})^{-1}\mathbf{R}_{\mathbf{ss}}\mathbf{E}_k, \tag{3}$$

with the SCMs $\mathbf{R}_{\mathbf{yy}} = E\{\mathbf{y}\mathbf{y}^H\}$ and $\mathbf{R}_{\mathbf{ss}} = E\{\mathbf{s}\mathbf{s}^H\}$ and $\mathbf{E}_k$ the $M \times J$ selection matrix extracting $\mathbf{d}_k$ from $\mathbf{s}$. In practice, under the assumption that noise and desired signal components are uncorrelated, $\mathbf{R}_{\mathbf{ss}}$ may be estimated as $\mathbf{R}_{\mathbf{yy}} - \mathbf{R}_{\mathbf{nn}}$, where $\mathbf{R}_{\mathbf{nn}} = E\{\mathbf{n}\mathbf{n}^H\}$. In real-world applications, SCMs must be estimated in an online fashion. This can be achieved via, e.g., exponential averaging with a forgetting factor $0 < \beta < 1$ as:

$$\begin{aligned}\mathbf{R}_{\mathbf{yy}}[t] &= \beta\mathbf{R}_{\mathbf{yy}}[t-1] + (1-\beta)\mathbf{y}[t]\mathbf{y}^H[t],\\ \mathbf{R}_{\mathbf{nn}}[t] &= \beta\mathbf{R}_{\mathbf{nn}}[t-1] + (1-\beta)\mathbf{n}[t]\mathbf{n}^H[t],\end{aligned} \tag{4}$$

where the same symbols are used for estimated and true quantities for simplicity of notation. In practice, $\mathbf{n}[t]$ may not be directly available and can be extracted via, e.g., an activity detector exploiting the ON-OFF structure of speech-like target signals [4], [8]. The desired signal estimate is finally obtained as $\hat{\mathbf{d}}_k = \mathbf{W}_k^H\mathbf{y}$.

In practice, making $\mathbf{y}$ available requires either a fusion center or the exchange of $M_k$-dimensional signals between the nodes of a fully connected WASN.

## III. TI-DANSE

The DANSE algorithm [4] may be used to reach the centralized solution of Eq.(2) while substantially reducing the communication bandwidth requirements. However, it requires a static and fully connected WASN topology, which is rarely the case in practice. In an ad-hoc, possibly time-varying WASN topology, one may instead employ the TI-DANSE algorithm [5] which is reviewed in this section.

The TI-DANSE algorithm is iterative with iteration index $i$, i.e., it asymptotically converges towards the centralized LMMSE solution. At any iteration, the WASN topology can be pruned to a new tree using, e.g., Prim's algorithm. One node is chosen as the updating node, a natural choice being the root of the tree. The index of the updating node is set to cycle through all $k \in \mathcal{K}$ in a round-robin fashion, such that all nodes have updated once after $K$ iterations. Each node $k$ defines its so-called fusion matrix $\mathbf{P}_k^i \in \mathbb{C}^{M_k\times J}$ and uses it to compute a $J$-channel fused version of its local sensor signals $\mathbf{z}_k^i = \mathbf{P}_k^{iH}\mathbf{y}_k$. All fused signals in the WASN are summed up via a sequence of partial in-network signal summations from the leaf nodes towards the root node. Once the full $J$-dimensional in-network sum $\boldsymbol{\eta}^i = \sum_{k\in\mathcal{K}} \mathbf{z}_k^i$ is built at the root node, it is flooded back towards the leaf nodes. Through this sequence of operations, each node has access to $D_k = M_k + J$ signals grouped in an observation vector:

$$\tilde{\mathbf{y}}_k^i = [\mathbf{y}_k^T \mid \boldsymbol{\eta}_{-k}^{iT}]^T \quad \text{where} \quad \boldsymbol{\eta}_{-k}^i = \boldsymbol{\eta}^i - \mathbf{z}_k^i = \sum_{q\in\mathcal{K}\setminus\{k\}} \mathbf{z}_q^i. \tag{5}$$

The objective of node $k$ is then to solve its own node-specific LMMSE problem to estimate $\mathbf{d}_k$ from $\tilde{\mathbf{y}}_k^i$:

$$\tilde{\mathbf{W}}_k^{i+1} = \arg\min_{\mathbf{W}} E\left\{\left\|\mathbf{d}_k - \mathbf{W}^H\tilde{\mathbf{y}}_k^i\right\|_2^2\right\}, \tag{6}$$

which, as in the centralized case, is solved by an MWF as:

$$\tilde{\mathbf{W}}_k^{i+1} = (\tilde{\mathbf{R}}_{\mathbf{y}_k\mathbf{y}_k}^i)^{-1}\tilde{\mathbf{R}}_{\mathbf{s}_k\mathbf{s}_k}^i\tilde{\mathbf{E}}_k, \tag{7}$$

with the SCMs $\tilde{\mathbf{R}}_{\mathbf{y}_k\mathbf{y}_k}^i = E\{\tilde{\mathbf{y}}_k^i\tilde{\mathbf{y}}_k^{iH}\}$ and $\tilde{\mathbf{R}}_{\mathbf{s}_k\mathbf{s}_k}^i = E\{\tilde{\mathbf{s}}_k^i\tilde{\mathbf{s}}_k^{iH}\}$, $\tilde{\mathbf{y}}_k^i = \tilde{\mathbf{s}}_k^i + \tilde{\mathbf{n}}_k^i$, and $\tilde{\mathbf{E}}_k = [\mathbf{E}_{kk}^T \mid \mathbf{0}_{J\times J}]^T$. A partitioning of $\tilde{\mathbf{W}}_k^i$ is defined here as $[\mathbf{W}_{kk}^{iT} \mid \mathbf{G}_k^{iT}]^T$, where $\mathbf{W}_{kk}^i \in \mathbb{C}^{M_k\times J}$ is applied to the $\mathbf{y}_k$ while $\mathbf{G}_k^i \in \mathbb{C}^{J\times J}$ is applied to $\boldsymbol{\eta}_{-k}^i$.

In [5], the convergence and optimality of TI-DANSE is proved when the fusion rule is $\mathbf{P}_k^i = \mathbf{W}_{kk}^i(\mathbf{G}_k^i)^{-1}$, which completes the algorithm definition. It can be noted that the $(\mathbf{G}_k^i)^{-1}$ term serves to decouple the contributions of each individual node in $\boldsymbol{\eta}^i$. The TI-DANSE definition of $\mathbf{P}_k^i$ is a fundamental difference with that used in the DANSE algorithm, where the $(\mathbf{G}_k^i)^{-1}$ term is omitted [4]. As $i \to \infty$, the TI-DANSE algorithm converges to the centralized solution

to Eq.(2) while reducing the amount of information exchanged between nodes with respect to the centralized case, as well as to DANSE. The algorithm assumes that the nodes update sequentially, in a round-robin fashion. Simultaneous or asynchronous node-updating strategies are not considered here.

## IV. TI-GEVD-DANSE

The MWF solving Eq.(2) depends on $\mathbf{R}_{\mathbf{ss}}$, which is a rank-$S$ matrix since $\mathbf{R}_{\mathbf{ss}} = \mathbf{A}E\{\hat{\mathbf{s}}\hat{\mathbf{s}}^H\}\mathbf{A}^H$ (cf. Eq.(1)). However, in practice, $\mathbf{R}_{\mathbf{ss}}$ is not directly available and must be estimated, for instance as $\mathbf{R}_{\mathbf{ss}} = \mathbf{R}_{\mathbf{yy}} - \mathbf{R}_{\mathbf{nn}}$, often resulting in a rank greater than $S$. In many practical applications such as speech enhancement, $S$ is known a priori or can be well-estimated. Constraining the rank of $\mathbf{R}_{\mathbf{ss}}$ can then be done via a GEVD of the pencil $\{\mathbf{R}_{\mathbf{yy}}, \mathbf{R}_{\mathbf{nn}}\}$. This low-rank approximation has been shown to yield more robust performance in scenarios where accurate estimation of $\mathbf{R}_{\mathbf{ss}}$ is challenging.

Since the TI-DANSE algorithm is also based on the signal model from Eq.(1), $\tilde{\mathbf{R}}^i_{\mathbf{s}_k\mathbf{s}_k}$ is also rank-$S$. This can be seen from the definitions below Eq.(7), rewriting:

$$\tilde{\mathbf{R}}^i_{\mathbf{s}_k\mathbf{s}_k} = E\left\{\tilde{\mathbf{s}}^i_k \tilde{\mathbf{s}}^{iH}_k\right\} = E\left\{\mathbf{C}^{iH}_k \mathbf{s}\mathbf{s}^H \mathbf{C}^i_k\right\} = \mathbf{C}^{iH}_k \mathbf{R}_{\mathbf{ss}} \mathbf{C}^i_k, \tag{8}$$

$$\text{where: } \mathbf{C}^{iH}_k = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I}_{M_k} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{P}^{iH}_1 & \cdots & \mathbf{P}^{iH}_{k-1} & \mathbf{0} & \mathbf{P}^{iH}_{k+1} & \cdots & \mathbf{P}^{iH}_K \end{bmatrix}. \tag{9}$$

Since $\mathbf{R}_{\mathbf{ss}}$ is rank-$S$, so is $\tilde{\mathbf{R}}^i_{\mathbf{s}_k\mathbf{s}_k}$. This low-rank property can be guaranteed by first making use of a GEVD on the pencil $\{\tilde{\mathbf{R}}^i_{\mathbf{y}_k\mathbf{y}_k}, \tilde{\mathbf{R}}^i_{\mathbf{n}_k\mathbf{n}_k}\}$ to rewrite the SCMs as:

$$\tilde{\mathbf{R}}^i_{\mathbf{y}_k\mathbf{y}_k} = \tilde{\mathbf{Q}}^{-H}_k \mathbf{\Sigma}^i_k \tilde{\mathbf{Q}}^{iH}_k \text{ and } \tilde{\mathbf{R}}^i_{\mathbf{n}_k\mathbf{n}_k} = \tilde{\mathbf{Q}}^{-H}_k \tilde{\mathbf{Q}}^{iH}_k, \tag{10}$$

where $\mathbf{\Sigma}^i_k = \text{diag}\{\tilde{\sigma}^i_{k1}, ..., \tilde{\sigma}^i_{kD_k}\}$ contains the generalized eigenvalues ordered from largest to smallest and $\tilde{\mathbf{Q}}^i_k$ contains the corresponding generalized eigenvectors. The rank can be constrained to $R$ (ideally equal to $S$, when $S$ is known a priori) by setting the $D_k - R$ smallest eigenvalues in $\tilde{\mathbf{R}}^i_{\mathbf{y}_k\mathbf{y}_k}$ to zero, obtaining an estimate of $\tilde{\mathbf{R}}_{\mathbf{s}_k\mathbf{s}_k}$ as:

$$\hat{\tilde{\mathbf{R}}}_{\mathbf{s}_k\mathbf{s}_k} = \hat{\tilde{\mathbf{R}}}^i_{\mathbf{y}_k\mathbf{y}_k} - \tilde{\mathbf{R}}^i_{\mathbf{n}_k\mathbf{n}_k} = \tilde{\mathbf{Q}}^{-H}_k \mathbf{\Delta}^i_k \tilde{\mathbf{Q}}^{iH}_k, \tag{11}$$

where $\mathbf{\Delta}^i_k = \text{diag}\{1-\tilde{\sigma}^i_{k1}, \ldots, 1-\tilde{\sigma}^i_{kR}, 0, \ldots, 0\}$. This results in the TI-GEVD-DANSE algorithm, which solution to Eq.(6) is obtained by substituting Eq.(11) into Eq.(7):

$$\tilde{\mathbf{W}}^{i+1}_k = (\tilde{\mathbf{Q}}^i_k)^{-H} \tilde{\mathbf{\Lambda}}^i_k \tilde{\mathbf{Q}}^{iH}_k \tilde{\mathbf{E}}_k, \tag{12}$$

where $\tilde{\mathbf{\Lambda}}^i_k = \text{diag}\{1-1/\tilde{\sigma}^i_{k1} \ldots 1-1/\tilde{\sigma}^i_{kR}, 0 \ldots 0\}$. The rest of the algorithm remains unchanged with respect to Section III. In addition to enabling enhanced and more robust signal estimation performance, it is observed that this rank-$R$ approximation preserves convergence of TI-GEVD-DANSE even in cases where $J$ or $R$ underestimates the number of latent desired sources $S$. This is not the case of TI-DANSE which rely on the assumption $S \leq J$. Although a convergence proof is not provided in this paper, the observed property is illustrated via the experimental results showcased in Section VI.

## V. Improved Robustness

### A. Non-Strict Convergence of TI-(GEVD-)DANSE

It is observed that the TI-(GEVD-)DANSE algorithm exhibits a non-strict convergence, which is due to the formulation of its fusion matrices. As derived in Section III, $\tilde{\mathbf{y}}^i_k = \mathbf{C}^{iH}_k \mathbf{y}$, implying that the desired signal estimate can be written as $\hat{\mathbf{d}}^{i+1}_k = \tilde{\mathbf{W}}^{i+1,H}_k \tilde{\mathbf{y}}^i_k = \tilde{\mathbf{W}}^{i+1,H}_k \mathbf{C}^{iH}_k \mathbf{y}$. Therefore, the $M \times J$ matrix $\mathbf{W}^{i+1}_k = \mathbf{C}^i_k \tilde{\mathbf{W}}^{i+1}_k$ is the network-wide version of the TI-(GEVD-)DANSE filter matrix, with structure:

$$\mathbf{W}^{i+1}_k = \begin{bmatrix} \left[(\mathbf{P}^i_1 \mathbf{G}^{i+1}_k)^T \mid \cdots \mid (\mathbf{P}^i_{k-1} \mathbf{G}^{i+1}_k)^T\right]^T \\ \mathbf{W}^{i+1}_{kk} \\ \left[(\mathbf{P}^i_{k+1} \mathbf{G}^{i+1}_k)^T \mid \cdots \mid (\mathbf{P}^i_K \mathbf{G}^{i+1}_k)^T\right]^T \end{bmatrix}. \tag{13}$$

Inspecting the terms $\mathbf{P}^i_q \mathbf{G}^{i+1}_k = \mathbf{W}^i_{qq} (\mathbf{G}^i_q)^{-1} \mathbf{G}^{i+1}_k$ in Eq.(13) reveals that convergence of $\mathbf{W}^i_k$ (and thus $\tilde{\mathbf{W}}^i_k$) does not require the filter coefficients in the $\{\mathbf{G}^i_k\}_{k\in\mathcal{K}}$ matrices to strictly converge. In fact, only the strict convergence of $(\mathbf{G}^i_q)^{-1}\mathbf{G}^{i+1}_k \forall\ (k,q) \in \mathcal{K} \times \mathcal{K}\backslash\{k\}$ is necessary. In this state, the TI-(GEVD-)DANSE algorithm allows the elements of the $\{\mathbf{G}^i_k\}_{k\in\mathcal{K}}$ matrices to grow infinitely large or small as $i$ increases. Such behavior is indeed observable in practice and can lead to numerical overflow or significant precision errors.

### B. Normalization strategy

In this section, we address the non-strict convergence of TI-(GEVD-)DANSE. Let us define a normalization factor $\gamma^i \in \mathbb{C}$ and suppose that every node simultaneously start normalizing their $\mathbf{G}^i_k$ matrix at iteration $i$. The normalized fused signal $\bar{\mathbf{z}}^i_k$ can be related to its non-normalized counterpart via:

$$\bar{\mathbf{z}}^i_k = (\mathbf{W}^i_{kk}(\mathbf{G}^i_k)^{-1}\gamma^i)^H \mathbf{y}_k = \gamma^{i,*} \mathbf{P}^{iH}_k \mathbf{y}_k = \gamma^{i,*} \mathbf{z}^i_k. \tag{14}$$

where $\cdot^*$ denotes the complex conjugate. Following the same notation logic, $\bar{\boldsymbol{\eta}}^i = \sum_{k\in\mathcal{K}} \bar{\mathbf{z}}^i_k = \gamma^{i,*}\boldsymbol{\eta}^i$, meaning that the normalized version of $\tilde{\mathbf{y}}^i_k$ can be expressed as $\bar{\tilde{\mathbf{y}}}^i_k = \mathbf{N}^i_k \tilde{\mathbf{y}}^i_k$ where $\mathbf{N}^i_k = \text{blkdiag}\{\mathbf{I}_{M_k}, \gamma^i \mathbf{I}_J\}$. It follows that:

$$\bar{\tilde{\mathbf{R}}}^i_{\mathbf{y}_k\mathbf{y}_k} = E\left\{\bar{\tilde{\mathbf{y}}}^i_k \bar{\tilde{\mathbf{y}}}^{iH}_k\right\} = \mathbf{N}^i_k \tilde{\mathbf{R}}^i_{\mathbf{y}_k\mathbf{y}_k} \mathbf{N}^{iH}_k, \tag{15}$$

and likewise for $\bar{\tilde{\mathbf{R}}}^i_{\mathbf{n}_k\mathbf{n}_k}$. Substituting in Eq.(10) gives:

$$\bar{\tilde{\mathbf{R}}}^i_{\mathbf{y}_k\mathbf{y}_k} = \bar{\tilde{\mathbf{Q}}}^{-H}_k \mathbf{\Sigma}^i_k \bar{\tilde{\mathbf{Q}}}^{iH}_k \text{ and } \bar{\tilde{\mathbf{R}}}^i_{\mathbf{n}_k\mathbf{n}_k} = \bar{\tilde{\mathbf{Q}}}^{-H}_k \bar{\tilde{\mathbf{Q}}}^{iH}_k, \tag{16}$$

where $\bar{\tilde{\mathbf{Q}}}^i_k = \mathbf{N}^i_k \tilde{\mathbf{Q}}^i_k$. Consequently, Eq.(12) gives:

$$\bar{\tilde{\mathbf{W}}}^{i+1}_k = (\bar{\tilde{\mathbf{Q}}}^i_k)^{-H} \tilde{\mathbf{\Lambda}}^i_k \bar{\tilde{\mathbf{Q}}}^{iH}_k \tilde{\mathbf{E}}_k = (\mathbf{N}^i_k)^{-H} \tilde{\mathbf{W}}^{i+1}_k, \tag{17}$$

where use is made of the fact that the desired signal only includes contributions from local sensors, i.e., $\mathbf{N}^{iH}_k \tilde{\mathbf{E}}_k = \tilde{\mathbf{E}}_k$. It can be shown that the normalization does not alter the network-wide filters of Eq.(13). This can be seen from:

$$\bar{\hat{\mathbf{d}}}^{i+1}_k = \bar{\tilde{\mathbf{W}}}^{i+1,H}_k \bar{\tilde{\mathbf{y}}}^i_k = \tilde{\mathbf{W}}^{i+1,H}_k (\mathbf{N}^i_k)^{-1} \mathbf{N}^i_k \tilde{\mathbf{y}}^i_k = \hat{\mathbf{d}}^{i+1}_k. \tag{18}$$

This normalization procedure can be incorporated in TI-GEVD-DANSE, resulting in the following algorithm:

1: Initialize $u = 0$, $\gamma^1 = 1$, $r \in \mathcal{K}$.
2: **for** $i = 1, 2, 3, \ldots$ **do**
3: Form tree topology rooted at node $u$.
4: At all $k \in \mathcal{K}$, compute $\bar{\mathbf{z}}_k^i = \gamma^{i,*} \bar{\mathbf{P}}_k^{iH} \mathbf{y}_k$.
5: At node $u$, compute the in-network sum $\boldsymbol{\eta}^i$ and flood it back through the WASN.
6: **for** $k \in \mathcal{K}$ **do**
7: Build observation vector $\bar{\tilde{\mathbf{y}}}_k^i$ as in Eq.(5).
8: Compute SCMs $\bar{\tilde{\mathbf{R}}}_{\mathbf{y}_k\mathbf{y}_k}^i$ and $\bar{\tilde{\mathbf{R}}}_{\mathbf{n}_k\mathbf{n}_k}^i$.
9: **if** $k = u$ **then**
10: Compute $\bar{\tilde{\mathbf{W}}}_k^{i+1}$ via Eq.(12).
11: **else if** $k \neq u$ **then**
12: Compute $\bar{\tilde{\mathbf{W}}}_k^{i+1} = \left(\mathbf{N}_k^i\right)^{-H} \bar{\tilde{\mathbf{W}}}_k^i$
13: **end if**
14: **end for**
15: At reference node $r$, compute $\gamma^{i+1} = \|\bar{\mathbf{G}}_r^{i+1}\|_F$ and flood it through the WASN.
16: At all $k \in \mathcal{K}$, compute $\hat{\mathbf{d}}_k^{i+1} = \bar{\tilde{\mathbf{W}}}_k^{i+1,H} \bar{\tilde{\mathbf{y}}}_k^{i+1}$.
17: $u \leftarrow (u+1) \mod K$.
18: **end for**

**Remark 1:** The choice for $\gamma^i$ is to use $\|\bar{\mathbf{G}}_r^{i+1}\|_F$ for a fixed reference node $r$ (step 15). It should be noted that the algorithm can in principle select any node $r$ as reference, as long as the normalization is equal for all nodes.

**Remark 2:** Step 12 ensures consistency across iterations for non-updating nodes $k \neq u$. Indeed, the $\gamma^{i,*}$ factor from step 4 is carried along in the SCMs and thus impacts the filter of the updating node. Conversely, a non-updating node will keep its filters from the previous iteration, meaning that the normalization effect must be accounted for via $(\mathbf{N}_k^i)^{-H}$.

**Remark 3:** The communication bandwidth increase generated by the broadcasting of $\gamma^{i+1}$ (step 15) is negligible in comparison to the unnormalized TI-GEVD-DANSE algorithm, as it represents the exchange of a single scalar at most at every iteration. Depending on the severity of the non-strictly convergent behavior of the $\{\bar{\mathbf{G}}_k^i\}_{k\in\mathcal{K}}$ coefficients, it may even be sufficient to update (and thus broadcast) $\gamma$ less frequently.

When estimating SCMs in an online fashion based on Eq.(4), the normalization must be accounted for by setting:

$$\begin{aligned} \bar{\tilde{\mathbf{R}}}_{\mathbf{y}_k\mathbf{y}_k}^i &= \beta \mathbf{N}_k^i \bar{\tilde{\mathbf{R}}}_{\mathbf{y}_k\mathbf{y}_k}^{i-1} \mathbf{N}_k^{iH} + (1-\beta) \bar{\tilde{\mathbf{y}}}_k^i \bar{\tilde{\mathbf{y}}}_k^{iH}, \\ \bar{\tilde{\mathbf{R}}}_{\mathbf{n}_k\mathbf{n}_k}^i &= \beta \mathbf{N}_k^i \bar{\tilde{\mathbf{R}}}_{\mathbf{n}_k\mathbf{n}_k}^{i-1} \mathbf{N}_k^{iH} + (1-\beta) \bar{\tilde{\mathbf{y}}}_k^i \bar{\tilde{\mathbf{y}}}_k^{iH}. \end{aligned} \tag{19}$$

where the index $i$ here represents both the iteration index and the time index, for simplicity.

## VI. Numerical Experiments

The performance of TI-GEVD-DANSE with and without normalization is assessed via simulations in an acoustic environment composed of an ad-hoc non-fully connected WASN and 6 localized sources, $S$=3 of which are considered as targets and the $S_n$=3 others as noise. The number of nodes is fixed but the specific WASN topology does not have to be, as it is simply assumed that $\boldsymbol{\eta}^i$ is available at all nodes at any $i$. Although the GEVD-rank $R$ can be chosen independently from the number of exchanged channels $J$, we here set $R$=$J$ for simplicity (other cases are discussed in, e.g., [7]). The uncorrelated (thermal) noise at node $k$ is set to have a power equal to 10% of the power of the combined target source signals as observed by the first sensor of the node. Signal samples and steering matrices entries are drawn from the uniform distribution over [-0.5, 0.5].

### A. Batch-mode simulations

Batch-mode simulations without normalization ($\gamma^i = 1 \,\forall\, i$) are first performed with $K$=5 and $M_k$=4 for all $k$ to demonstrate the convergence of TI-GEVD-DANSE towards the centralized GEVD-MWF. The number of observations is set to $N$=15000 samples. The SCMs as estimated as sample means. The results are averaged over 3 runs with different random steering matrices. In order to isolate SCM estimation errors, $\mathbf{R}_{\mathbf{nn}}$ and $\bar{\tilde{\mathbf{R}}}_{\mathbf{n}_k\mathbf{n}_k}^i$ are estimated using oracle knowledge of the noise-only signals. The performance of the TI-GEVD-DANSE algorithm is shown for different values of $R$=$J$ in Fig. 1, where two mean square error (MSE) metrics are used. The first, $\mathrm{MSE}_{W_k}^i$, is defined between the network-wide expansion of the TI-GEVD-DANSE solution of Eq.(13) and the centralized GEVD-MWF solution of Eq.(2). The second, $\mathrm{MSE}_{d_k}^i$, is defined between the true desired signal and its estimate $\hat{\mathbf{d}}_k^i$:

$$\mathrm{MSE}_{W_k}^i = \frac{1}{MJ} \|\bar{\mathbf{W}}_k^i - \mathbf{W}_k\|_F^2, \tag{20}$$

$$\mathrm{MSE}_{d_k}^i = \frac{1}{JN} \sum_{n=0}^{N-1} \|\hat{\mathbf{d}}_k^i[n] - \mathbf{d}_k[n]\|_F^2. \tag{21}$$

The convergence of the TI-GEVD-DANSE algorithm towards the centralized GEVD-MWF is clearly visible from the average $\mathrm{MSE}_{W_k}^i$, which would decrease to machine precision for $i \to \infty$. Larger values of $R$=$J$ show a faster convergence since more distinct signals are available at each nodes. It is important to notice that convergence is preserved even in cases where $R$=$J$<$S$, which is not the case for the original TI-DANSE [5]. The average $\mathrm{MSE}_{d_k}^i$ shows that TI-GEVD-DANSE also matches the target signal estimation performance of the GEVD-MWF with corresponding rank, and that a rank $R$ closer (or equal) to $S$ yields better performance.

### B. Online-mode simulations

The behavior of TI-GEVD-DANSE is assessed in an online processing scenario with and without normalization, with $K$=10 and $M_k$=15 for all $k$. The SCM estimation strategy defined in Eq.(19) is used with $\beta$=0.7 and all SCM entries randomly initialized. At every iteration, a new frame of $B = 500$ samples is drawn from the uniform distribution over [-0.5, 0.5] for each sound source. A dynamic scenario is created where the entries of steering matrices $\mathbf{A}$ and $\mathbf{B}$ defined in Eq.(1)

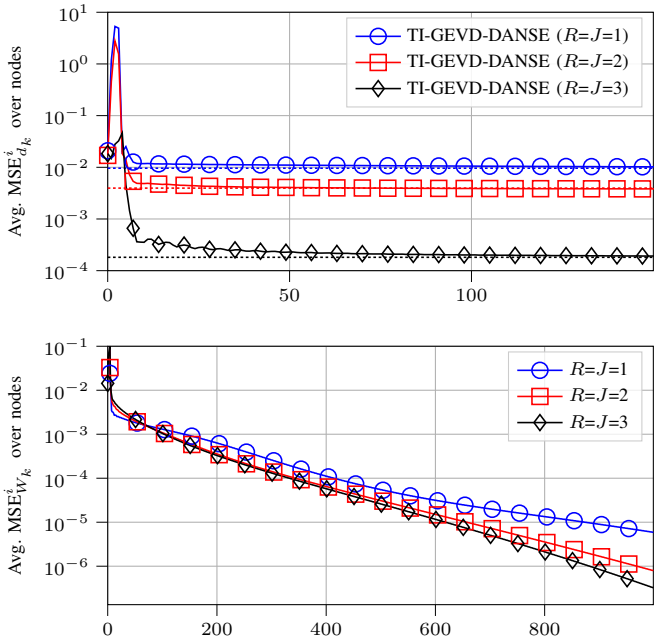



Fig. 1. Batch processing results. The $x$-axis represents the iteration index $i$. Top: $\text{MSE}^i_{d_k}$ averaged over nodes, where the horizontal dotted lines represent the corresponding centralized values. Bottom: $\text{MSE}^i_{W_k}$ averaged over nodes.

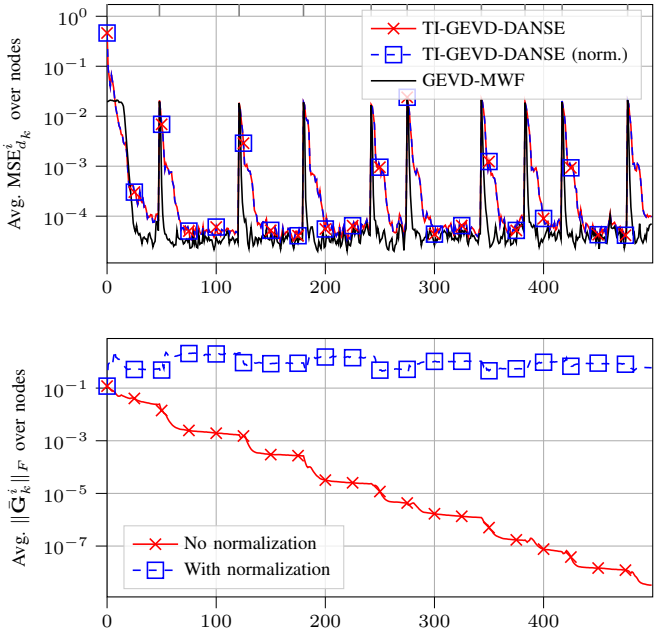



Fig. 2. Online processing results. The $x$-axis represents the index $i$. Top: $\text{MSE}^i_{d_k}$ averaged over nodes. Bottom: $\|\bar{\mathbf{G}}^i_k\|_F$ averaged over nodes.

have, at each time frame, a 0.01 probability to be changed by drawing entries from the same uniform distribution over [-0.5, 0.5], thus changing the relative positioning of sensors and sources.

The quantity $\text{MSE}^i_{d_k}$ averaged over all nodes is shown in Fig. 2 for TI-GEVD-DANSE with and without normalization and for the centralized GEVD-MWF, all with $J$=$R$=$S$=3. To highlight the effect of $\gamma^i$, the average over all nodes of $\|\bar{\mathbf{G}}^i_k\|_F$ is also shown for TI-GEVD-DANSE with and without normalization. The reference node index $r$ is kept equal to 1 through the entire simulation.

The results show that the normalization has no impact on the TI-GEVD-DANSE performance as per Eq.(18), even in a dynamic scenario. It can be noticed that a change in the steering matrices triggers a new adaptation phase where the algorithm must re-estimate the SCMs to be able to re-converge towards the centralized GEVD-MWF (visible as peaks in the average $\text{MSE}^i_{d_k}$).

The importance of the normalization becomes clear when inspecting the lower plot. Without normalization, the inherent non-strict convergence of TI-GEVD-DANSE leads to decreasing values of $\|\bar{\mathbf{G}}^i_k\|_F = \|\mathbf{G}^i_k\|_F$. Steeper decreases are observed when the acoustic scenario is changed, as the SCMs must be re-estimated. If kept untouched, this behavior would result in numerical instability over time. With normalization, however, the average norm of $\bar{\mathbf{G}}^i_k$ remains in the vicinity of 1 even though the acoustic scenario changes, which allows TI-GEVD-DANSE to indefinitely perform without numerical overflow or significant precision errors.

## VII. Conclusion

In this paper, we have introduced a GEVD-based TI-DANSE algorithm (TI-GEVD-DANSE) which provides an enhanced and robust performance in scenarios where the accurate estimation of the desired-signal SCM is challenging. An adaptive normalization procedure has been included, ensuring the stability of the estimated filter coefficients through time, even in online processing with time-varying acoustic scenarios.

## References

[1] A. Bertrand, "Applications and trends in wireless acoustic sensor networks: A signal processing perspective," Proc. 18th IEEE Symp. Commun. Veh. Tech., 2011, pp.1–6.

[2] A. Boukerche and P. Sun, "Design of Algorithms and Protocols for Underwater Acoustic Wireless Sensor Networks," ACM Comput. Surv., vol. 53, no. 6, art. 134, pp 1–34, Nov. 2021.

[3] S. X. Wu, H.-T. Wai, L. Li and A. Scaglione, "A Review of Distributed Algorithms for Principal Component Analysis," in Proc. of the IEEE, vol. 106, no. 8, pp. 1321–1340, Aug. 2018.

[4] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks–Part I: Sequential node updating," IEEE Trans. Signal Process., vol. 58, no. 10, pp. 5277–5291, Oct. 2010.

[5] J. Szurley, A. Bertrand, and M. Moonen, "Topology-independent distributed adaptive node-specific signal estimation in wireless sensor networks," IEEE Trans. Signal Inf. Process. Netw., vol. 3, no. 1, pp. 130–144, Mar. 2017.

[6] R. Serizel, M. Moonen, B. Van Dijk, and J. Wouters, "Low-rank approximation based multichannel Wiener filter algorithms for noise reduction with application in cochlear implants," IEEE/ACM Trans. Audio, Speech, Lang. Process., vol. 22, no. 4, pp.785-799, Feb. 2014.

[7] A. Hassani, A. Bertrand, and M. Moonen, "GEVD-based low-rank approximation for distributed adaptive node-specific signal estimation in wireless sensor networks," IEEE Trans. Signal Process., vol. 64, no. 10, pp. 2557–2572, May 2016.

[8] Y. Zhao, J. K. Nielsen, J. Chen, and M. G. Christensen, "Model-based distributed node clustering and multi-speaker speech presence probability estimation in wireless acoustic sensor networks," J. Acoust. Soc. Amer., vol. 147, no. 6, pp. 4189–4201, 2020.