IEEE Open Journal of Signal Processing

Received XX Month, XXXX; revised XX Month, XXXX; accepted XX Month, XXXX; Date of publication XX Month, XXXX; date of current version May 31, 2025.

Distributed Multichannel Wiener Filter for Wireless Acoustic Sensor Networks

PAUL DIDIER[1], TOON VAN WATERSCHOOT[1], SIMON DOCLO[2,3], JÖRG BITZER[3], POURYA BEHMANDPOOR[4], HENRI GODE[2], AND MARC MOONEN[1]

[1]STADIUS Center for Dynamical Systems, Department of Electrical Engineering (ESAT), KU Leuven, 3001 Leuven, Belgium
[2]Signal Processing Group, Department of Medical Physics and Acoustics and Cluster of Excellence Hearing4all, Carl von Ossietzky Universität Oldenburg, Oldenburg, Germany
[3]Fraunhofer IDMT, Project Group Hearing, Speech and Audio Technology, Oldenburg, Germany
[4]Department of Electronics and Informatics, Vrije Universiteit Brussel, B-1050 Brussels, Belgium

CORRESPONDING AUTHOR: Paul Didier (e-mail: paul.didier@esat.kuleuven.be).

This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information. The scientific responsibility is assumed by the authors. Funding sources to be added.

ABSTRACT Notes to co-authors are highlighted in **red**.
Within a wireless acoustic sensor network (WASN), devices collaborate using distributed algorithms to jointly solve audio signal processing tasks. This paper addresses distributed optimal estimation of node-specific target signals via network-wide beamforming, where optimality means matching the performance of a centralized system aggregating all sensor signals. To reduce communication bandwidth, the algorithm should aim to let nodes exchange as few signal channels as possible. Existing solutions, such as the distributed adaptive node-specific signal estimation (DANSE) algorithm, iteratively converge towards the optimum, which can be slow and impractical in dynamic scenarios. Many solutions also assume that all nodes observe the same sources, which is often not the case in practice.
To overcome these limitations, we first propose the distributed multichannel Wiener filter (dMWF) algorithm for fully connected WASNs. dMWF is optimal, non-iterative, and works even when nodes observe different sets of sources. Nodes exchange neighbor-pair-specific, low-dimensional (fused) signals estimating the contribution of sources observed by both nodes in the pair. We then extend to arbitrary topologies with the topology-independent dMWF (TI-dMWF), where, after pruning the WASN to a tree, nodes transmit low-dimensional, branch-specific estimates of sources observed by all nodes in a branch. We formally prove the optimality of both dMWF and TI-dMWF and demonstrate their performance in simulated speech enhancement experiments. Both algorithms outperform DANSE and topology-independent DANSE (TI-DANSE), respectively, in terms of objective metrics after short operation times, highlighting the benefit of their iterationless design.

INDEX TERMS Distributed signal processing, wireless acoustic sensor networks, multichannel Wiener filter

I. Introduction

Microphone arrays have been widely used in various applications to tackle various signal processing tasks such as noise reduction, acoustic echo cancellation, or dereverberation [1]. They have traditionally been deployed in *centralized* settings where the audio signals are aggregated and processed in a central processing unit, the so-called *fusion center*. However, *distributed* settings recently garnered increasing attention due to the possibility to build wireless acoustic sensor networks (WASNs), as more and more devices capable of recording, processing, and wirelessly transmitting audio data (e.g., laptops, smartphones, hearing aids) are available in our surroundings. Such networks are commonly described in theoretical terms as wireless communication graphs where each *node* is a device. Nodes in a WASN may be arbitrarily positioned, rendering WASNs flexible, scalable, and capable of covering wider spatial areas than centralized systems. These important advantages have led to the development of distributed audio signal processing algorithms.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

Distributed algorithms enable collaboration between nodes to tackle a particular audio signal processing task without relying on a fusion center. The goal is then to reach a performance that is as good as that of an equivalent centralized system, while accounting for the inherent challenges of distributed settings, e.g., limited communication bandwidth, synchronicity, or wireless link failures [2]. In this paper, we focus on node-specific distributed signal estimation through beamforming [3], i.e., the challenge of distributedly estimating a node-specific target signal as an optimal linear combination of multiple noisy signals. Optimality is here defined as the state where the distributedly computed beamformer generates the same outputs as in a centralized setting where a fusion center would have access to all individual sensor signals in the WASN. We specifically aim at addressing the communication bandwidth constraints present in distributed settings, and explicitly choose not to consider other (albeit equally important) challenges such as synchronicity [4]–[7], optimal sensor selection [8]–[10], packet losses, etc.

Several widely used types of beamformers exist, such as the minimum variance distortionless response (MVDR) beamformer and its generalization, the linearly constrained minimum variance (LCMV) beamformer, which minimizes output power while maintaining a desired response toward the target [3]. Both rely on the estimation of acoustic transfer functions between the sensors and the sources of interest. The multichannel Wiener filter (MWF) [11] minimum mean square error (MMSE) estimator represents another class of beamformers which rely on the estimation of spatial covariance matrices SCMs. The latter is the focus of the present paper.

Research around beamformer-based distributed signal estimation has led to the development of a variety of distributed beamformers, always striving to limit energy consumption or communication bandwidth usage [12]. For instance, distributed versions of the LCMV beamformer have been developed [13], [14], as well as distributed MVDR beamformers [15]–[17], or distributed mean square error (MSE)-optimal beamformers [18]. Attempts to formulate a distributed MWF beamformer with limited communication bandwidth usage [19] have led to the introduction of the distributed adaptive node-specific signal estimation (DANSE) algorithm [20]–[22], originally only applicable in fully connected (FC) WASNs. In DANSE, nodes only transmit dimensionally reduced (i.e., fused) versions of their local sensor signals. Several extensions of this algorithm have then been proposed. Among them, the topology-independent (TI) DANSE algorithm (TI-DANSE) allows, upon convergence, nodes to reach the same performance as an equivalent centralized system in non-FC WASNs [23], [24]. These algorithms all together constitute the DANSE family of algorithms.

All aforementioned algorithms, DANSE-related and others, compute distributed beamformers in an iterative fashion. In other words, they require to go through a certain number of iterations to converge towards and eventually reach an optimum. This optimum is the set of filter coefficients which will provide the same performance as an optimal centralized system deployed in the same acoustic scene with the same sensor positions. Note that algorithmic iterations are not to be confused with time-frames in an online processing setting. Even non-iterative solutions that perform in an online fashion make use of data from different time-frames, e.g., for the estimation of SCMs. In iterative distributed beamformers, the number of iterations required to reach optimality generally increases with the number of nodes in the WASN. In some cases, that number even further increases if the WASN is not FC [23]. To achieve optimal beamforming, each iteration typically involves the estimation of a new set of second-order statistics, which often take the form of SCMs between signals coming from different nodes. In order to estimate SCMs with sufficient accuracy, time-averaging over multiple time-frames is often employed. As a result, current state-of-the-art beamformer-based distributed signal estimation algorithms require the final application to tolerate possibly long waiting times before optimality (i.e., centralized performance) is reached. This significantly limits their relevance in practical applications involving users and/or time-varying acoustic environments, where quick adaptivity is paramount for high quality of service. It should be noted that, although non-iterative distributed beamformers have been proposed [25], [26], they generally rely on heuristic arguments and do not aim at solving a particular optimization problem.

To guarantee optimality upon convergence, many distributed beamformer algorithms, the DANSE family of algorithms [13], [21], [23], assume that all sound sources of interested are observed by all nodes in the WASN, i.e., that the same low-dimensional signal subspace is spanned by the target signals of all nodes. We refer to this as a scenario with *fully overlapping subspace*. In any given acoustic environment, a sound source is only observed by a node if the signal it generates contributes to the sensor signals captured at that node. The case may arise where a particular source of interest to a node does not contribute to the local sensor signals of another node – the source may, e.g., be too far or obstructed from that node. In such case, the subspaces spanned by the target signals of different nodes does not fully overlap, which is generally referred to as a scenario with *partially overlapping subspaces*, and optimality of the aforementioned algorithms cannot be guaranteed.

A. Contributions

The contributions of this paper are summarized as follows. First, we introduce the distributed multichannel Wiener filter (dMWF) algorithm, a theoretically optimal distributed MWF beamformer which allows all nodes in an FC WASNs to match the performance of an equivalent centralized MWF. The dMWF algorithm reaches optimality without iterations while nodes only transmit fused versions of their local sensor signals. These fused signals are in fact node-pair-

specific estimates of the contribution of sources observed by both nodes in the pair. The general and non-iterative formulation of the dMWF algorithm allows it to stay optimal in scenarios with partially overlapping subspaces. Using the dMWF algorithm, all nodes can estimate their node-specific target signal as accurately as if they had access to all sensor signals in the WASN. A formal optimality proof is provided, showing that the dMWF solution is equivalent to the centralized solution in an FC WASN.

Second, we introduce the topology-independent dMWF (TI-dMWF) algorithm, which extends the applicability the dMWF algorithm to any, possibly non-FC and time-varying WASNs. The communication bandwidth usage is kept as low as for the dMWF algorithm, as nodes only exchange low-dimensional signals along a tree pruned from the original topology. A formal optimality proof is also provided, showing that the TI-dMWF solution is also optimal.

Finally, the performance of dMWF and TI-dMWF is demonstrated in simulated experiments considering speech enhancement scenarios. Their advantage over DANSE and TI-DANSE, respectively, are shown in terms of convergence speed and objective metrics after short functioning times.

B. Paper organization

In Section II, the signal model and centralized solutions are defined. In Section III, the dMWF algorithm is formulated for application in FC WASNs and its optimality is formally proven. In Section IV, the TI-dMWF algorithm is formulated for application in any, possibly non-FC WASNs, and its optimality is formally proven. Simulated experiments are provided in Section V. Finally, conclusions are formulated in Section VI.

Notation: Italic letters denote scalars and lowercase boldface letters denote column vectors. Uppercase boldface letters are used for matrices. The superscripts $\cdot^{\mathrm{T}}$ and $\cdot^{\mathrm{H}}$ denote the transpose and Hermitian transpose operators, respectively. The symbols $\mathbb{E}\{\cdot\}$, $\|\cdot\|_2$, and $\|\cdot\|_F$ denote the expected value operator, the Euclidean norm, and the Frobenius norm, respectively. The operator $\mathrm{diag}\{\mathbf{a}\}$ transforms the vector $\mathbf{a}$ in a diagonal matrix with $\mathbf{a}$ on its main diagonal. The symbol $|\mathcal{A}|$ denotes the number of elements in the set $\mathcal{A}$ and $\emptyset$ denotes the empty set. The symbol $\mathbf{0}_{A\times B}$ denotes the $A\times B$ all-zeros matrix and $\mathbf{I}_A$ denotes the $A\times A$ identity matrix. The sets of complex numbers is denoted by $\mathbb{C}$.

II. Problem Statement and Centralized Solution

Consider K nodes constituting a WASN, where each node has M_k sensors, where $k \in \mathcal{K} \triangleq \{1,\dots,K\}$ is the node index. The total number of sensors in the WASN is denoted $M \triangleq \sum_{k\in\mathcal{K}} M_k$. This network is deployed in an acoustic environment where Q mutually uncorrelated sound sources are present. Among those, Q_{d} are desired sources and $Q_{\mathrm{n}} = Q - Q_{\mathrm{d}}$ are noise sources. Each source generates a latent signal which is assumed to be complex-valued to allow frequency-domain representations, e.g., in the short-term Fourier transform (STFT) domain. At time t, the m-th sensor of node k collects the signal $y_{k,m}[t]$, which is one entry of the M_k-dimensional node-specific signal vector $\mathbf{y}_k[t] \triangleq [y_{k,1}[t],\dots,y_{k,M_k}[t]]^{\mathrm{T}} \in \mathbb{C}^{M_k\times 1}$. In the following, we apply our theory to finite time-frames of signals, assuming that all signals are short-term stationary and ergodic, such that signal statistics can be estimated through averaging over several time-frames. For conciseness, the index $[t]$ will be discarded unless explicitly mentioned.

The signal model at node k is then:

$$\mathbf{y}_k = \overbrace{\mathbf{A}_k\mathbf{s}^{\mathrm{lat}}}^{\mathbf{s}_k} + \overbrace{\mathbf{B}_k\mathbf{n}^{\mathrm{lat}}}^{\mathbf{n}_k} + \mathbf{v}_k, \tag{1}$$

where $\mathbf{s}^{\mathrm{lat}} \in \mathbb{C}^{Q_{\mathrm{d}}\times 1}$ and $\mathbf{n}^{\mathrm{lat}} \in \mathbb{C}^{Q_{\mathrm{n}}\times 1}$ contain the stacked latent desired and noise source signals, respectively, while $\mathbf{A}_k \in \mathbb{C}^{M_k\times Q_{\mathrm{d}}}$ and $\mathbf{B}_k \in \mathbb{C}^{M_k\times Q_{\mathrm{n}}}$ are the corresponding desired and noise steering matrices, respectively. The vector $\mathbf{v}_k$ represents the self-noise, uncorrelated across sensors. Note that (1) can correspond to the narrowband approximation of the time-domain convolutive mixing process as a multiplication per frequency bin, in which case all subsequent expressions are frequency-bin-specific. It is important to note that the narrowband approximation is generally only valid when the mixing filters (i.e., the room impulse responses between sources and sensors) are short compared to the STFT window length [27].

We consider a scenario with *partially overlapping subspaces*, i.e., where any source may or may not be observed by any node. A node may not be able to observe a source if, for example, it is located too far from it, or if it is acoustically isolated from the source due to the presence of a solid object in the acoustic path. We let $\mathcal{O}_{\mathrm{d},k}$ and $\mathcal{O}_{\mathrm{n},k}$ denote the sets of desired sources and noise sources observed by node k, respectively. If desired source index $j \in \mathcal{O}_{\mathrm{d},k}$, then the j-th column of $\mathbf{A}_k$ is all-zeros. Similarly, if noise source index $j \in \mathcal{O}_{\mathrm{n},k}$, then the j-th column of $\mathbf{B}_k$ is all-zeros. The combined set of observed sources at node k is $\mathcal{O}_k \triangleq \mathcal{O}_{\mathrm{d},k} \cup \mathcal{O}_{\mathrm{n},k}$. It is assume that, from the point of view of a note, any observable source that isn't a noise source is a source of interest. It should be kept in mind that node-specific source interests, in addition to node-specific source observabilities, may occur in practice [28], yet this generalization is not considered here for the sake of a simpler exposition.

The node-specific signal model from (1) can be made network-wide by stacking the local sensor signal vectors as $\mathbf{y} \triangleq [\mathbf{y}_1^{\mathrm{T}},\dots,\mathbf{y}_K^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{C}^{M\times 1}$, obtaining:

$$\mathbf{y} = \overbrace{\mathbf{A}\mathbf{s}^{\mathrm{lat}}}^{\mathbf{s}} + \overbrace{\mathbf{B}\mathbf{n}^{\mathrm{lat}}}^{\mathbf{n}} + \mathbf{v}, \tag{2}$$

where the node-specific desired and noise steering matrices are stacked in $\mathbf{A} \in \mathbb{C}^{M\times Q_{\mathrm{d}}}$ and $\mathbf{B} \in \mathbb{C}^{M\times Q_{\mathrm{n}}}$, respectively,

and $\mathbf{v} \triangleq [\mathbf{v}_1^{\mathrm{T}}, \dots, \mathbf{v}_K^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{C}^{M \times 1}$. Again, if node k does not observe desired source j, a $M_k \times 1$ block in the j-th column of $\mathbf{A}$ is all-zeros, and a similar reasoning can be made for noise sources and $\mathbf{B}$.

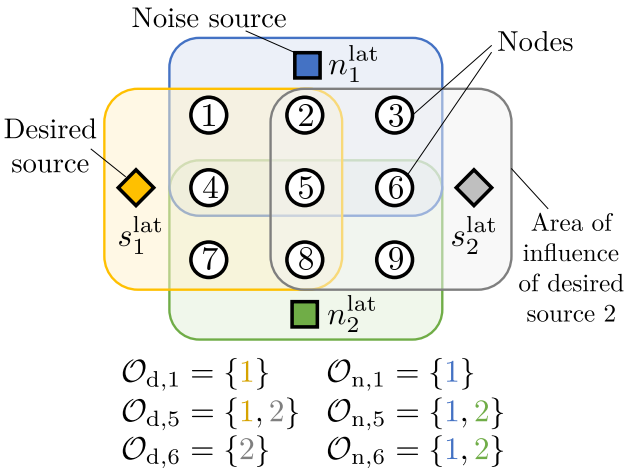

FIGURE 1. Example of sources observability in a $K = 9$ nodes WASN.

We consider a scenario where nodes are interested in estimating the desired components in D of their local sensor signals. Without loss of generality (w.l.o.g.), we assume the first D channels are of interest, at each node. The node-specific target signal $\mathbf{d}_k$ is thus:

$$\mathbf{d}_k \triangleq \mathbf{E}_{kk}^{\mathrm{T}} \mathbf{s}_k = \mathbf{E}_k^{\mathrm{T}} \mathbf{s} \in \mathbb{C}^{D \times 1}, \tag{3}$$

where $\mathbf{E}_{kk} = [\mathbf{I}_D \mid \mathbf{0}]^{\mathrm{T}} \in \{0,1\}^{M_k \times D}$ extracts the first D channels of $\mathbf{s}_k$ and $\mathbf{E}_k \in \{0,1\}^{M \times D}$ extracts $\mathbf{d}_k$ from $\mathbf{s}$.

Consider a centralized scenario where the signals of all nodes in the WASN are aggregated at a fusion center. In this case, the centralized signal vector $\mathbf{y}$ can be used to estimate $\mathbf{d}_k, \forall\, k \in \mathcal{K}$. The following node-specific linear minimum mean square error (LMMSE) problem is defined:

$$\widehat{\mathbf{W}}_k \triangleq \underset{\mathbf{W} \in \mathbb{C}^{M \times D}}{\arg\min}\ \mathbb{E}\left\{ \left\| \mathbf{d}_k - \mathbf{W}^{\mathrm{H}} \mathbf{y} \right\|_2^2 \right\}, \tag{4}$$

which has the well-known MWF closed-form solution:

$$\widehat{\mathbf{W}}_k = \mathbf{R}_{\mathbf{yy}}^{-1} \mathbf{R}_{\mathbf{yd}_k}, \tag{5}$$

with $\mathbf{R}_{\mathbf{yy}} \triangleq \mathbb{E}\left\{\mathbf{y}\mathbf{y}^{\mathrm{H}}\right\}$ and $\mathbf{R}_{\mathbf{yd}_k} \triangleq \mathbb{E}\left\{\mathbf{y}\mathbf{d}_k^{\mathrm{H}}\right\}$. Note that, in general, $\mathbf{R}_{\mathbf{yy}}$ will be full-rank and thus invertible due to the presence of the uncorrelated self-noise $\mathbf{v}$ at all sensors. In practice, $\mathbf{d}_k$ is unavailable and $\mathbf{R}_{\mathbf{yd}_k}$ must thus be indirectly estimated. It holds that:

$$\begin{aligned} \mathbf{R}_{\mathbf{yd}_k} &= \mathbb{E}\left\{\mathbf{y}\mathbf{d}_k^{\mathrm{H}}\right\} && (6)\\ &= \mathbb{E}\left\{ (\mathbf{s} + \mathbf{n} + \mathbf{v}) \left(\mathbf{E}_k^{\mathrm{T}} \mathbf{s}\right)^{\mathrm{H}} \right\} && (7)\\ &= \mathbb{E}\left\{\mathbf{s}\mathbf{s}^{\mathrm{H}}\right\} \mathbf{E}_k \triangleq \mathbf{R}_{\mathbf{ss}} \mathbf{E}_k, && (8) \end{aligned}$$

which shows that estimating $\mathbf{R}_{\mathbf{ss}}$ is sufficient to estimate $\mathbf{R}_{\mathbf{yd}_k}$. Note that (8) holds since noise sources are uncorrelated to desired sources and to self-noise. For the same reason, we can write $\mathbf{R}_{\mathbf{ss}} = \mathbf{R}_{\mathbf{yy}} - \mathbf{R}_{\mathbf{nn}}$. In speech enhancement scenarios, the SCMs $\mathbf{R}_{\mathbf{yy}}$ and $\mathbf{R}_{\mathbf{nn}}$ can be estimated based on $\mathbf{y}$ by leveraging the ON-OFF structure of speech signals and the stationarity of typical noise signals. By means of a voice activity detector (VAD) or speech presence probability (SPP) estimate [29]–[34], noise-only periods can be identified and used to estimate $\mathbf{R}_{\mathbf{nn}}$, while $\mathbf{R}_{\mathbf{yy}}$ is estimated during speech+noise periods. Using exponential averaging with a forgetting factor $1 > \beta \gg 0$ at time frame t:

$$\begin{aligned} \mathbf{R}_{\mathbf{yy}}[t] &= \beta \mathbf{R}_{\mathbf{yy}}[t-1] + (1-\beta)\mathbf{y}\mathbf{y}^{\mathrm{H}} \text{ if VAD=1},\\ \mathbf{R}_{\mathbf{nn}}[t] &= \beta \mathbf{R}_{\mathbf{nn}}[t-1] + (1-\beta)\mathbf{y}\mathbf{y}^{\mathrm{H}} \text{ if VAD=0}. \end{aligned} \tag{9}$$

Once $\mathbf{R}_{\mathbf{yy}}$ and $\mathbf{R}_{\mathbf{nn}}$ (and thus $\mathbf{R}_{\mathbf{yd}_k} = (\mathbf{R}_{\mathbf{yy}} - \mathbf{R}_{\mathbf{nn}})\mathbf{E}_k$) are estimated, $\widehat{\mathbf{W}}_k$ can be computed via (5) and the centralized target signal estimate at node k is obtained as:

$$\widehat{\mathbf{d}}_{k,\mathrm{centr}} \triangleq \widehat{\mathbf{W}}_k^{\mathrm{H}} \mathbf{y}, \forall\, k \in \mathcal{K}. \tag{10}$$

III. Distributed MWF in FC WASNs: dMWF

In Section II, the node-specific signal estimation solution was outlined for a centralized scenario. This approach has various drawbacks, including the need for a fusion center, which may not be available in practice, and the requirement for all nodes to transmit their local sensor signals to the fusion center, which can lead to excessive communication bandwidth usage. In this section, a fusion-center-free scenario is considered, where we leverage the capacity of nodes to wirelessly transmit data between each other. We here consider FC WASNs, i.e., WASNs where every node can directly communicate with every other node. The non-FC case will be addressed in Section IV. The dMWF algorithm is defined in this section. It allows each node to accurately estimate its target signal while only transmitting low-dimensional fused versions of local sensor signals between nodes. The dMWF algorithm is optimal in the sense that it reaches the same performance as an equivalent centralized system deployed in the same acoustic scene with the same sensor positions.

A. Algorithm Structure

Some useful groupings of the sound sources present in the acoustic scene are first introduced. The desired sources observed by both q and k are contained in the set $\mathcal{O}_{\mathrm{d},kq} \triangleq \mathcal{O}_{\mathrm{d},k} \cap \mathcal{O}_{\mathrm{d},q}$. Similarly for noise sources, $\mathcal{O}_{\mathrm{n},kq} \triangleq \mathcal{O}_{\mathrm{n},k} \cap \mathcal{O}_{\mathrm{n},q}$. The set of all sources (desired and noise) observed by both k and q is denoted by $\mathcal{O}_{kq} \triangleq \mathcal{O}_{\mathrm{d},kq} \cup \mathcal{O}_{\mathrm{n},kq}$. All sets are summarized in Table 1.

In order to estimate its own target signal, node k can only make use of the part of $\mathbf{y}_q$ that corresponds to the contributions of sources in $\mathcal{O}_{kq}, \forall\, q \in \mathcal{K}_k$, where $\mathcal{K}_k \triangleq \mathcal{K} \backslash \{k\}$. Indeed, if node q observes a desired source that node k also

Observed by $\rightarrow$	k	q	k and q
Desired sources	$\mathcal{O}_{\mathrm{d},k}$	$\mathcal{O}_{\mathrm{d},q}$	$\mathcal{O}_{\mathrm{d},kq}$
Noise sources	$\mathcal{O}_{\mathrm{n},k}$	$\mathcal{O}_{\mathrm{n},q}$	$\mathcal{O}_{\mathrm{n},kq}$
All sources	$\mathcal{O}_{k}$	$\mathcal{O}_{q}$	$\mathcal{O}_{kq}$

TABLE 1. Notation summary for observed sources sets.

observes, node k can use $\mathbf{y}_q$ to better estimate $\mathbf{d}_k$. If node q observes a noise source that node k also observes, node k can use $\mathbf{y}_q$ to reduce the influence of noise in $\widehat{\mathbf{d}}_k$ and thus better estimate $\mathbf{d}_k$. For the sake of simplicity of exposition, we assume from here on that $M_k \geq |\mathcal{O}_{kq}|, \forall (k,q) \in \mathcal{K}\times\mathcal{K}_k$. Adaptation strategies in scenarios where this does not hold are provided in a later subsection. ← This is not written yet.

The goal for any node in a distributed signal estimation algorithm, aside from estimating its own target signal as well as an equivalent centralized system, is to minimize the amount of data that it must transmit to its neighbors. This requirement is key to meet the communication bandwidth usage constraints of distributed systems, which are typically much more stringent than those of centralized systems [2]. A naive method to distributedly reach the same performance as an equivalent centralized system would be to let each node send all its local sensor signals to all other nodes. We will show in the following that this is not necessary, and that it is possible to reduce the amount of data transmitted while still reaching optimality.

The dMWF algorithm relies on the key concept of local sensor signals *fusion*. Instead of letting node $q \in \mathcal{K}$ send all its M_q local sensor signals $\mathbf{y}_q$ to all other nodes, it aims at sending a dimensionally reduced linear combination of $\mathbf{y}_q$, i.e., a *fused* version of $\mathbf{y}_q$. This fusion operation can be made neighbor-specific, i.e., a different linear combination is used for each neighbor, i.e., node q sends to node $k \in \mathcal{K}_q$ the signal:

$$\mathbf{z}_q^k \triangleq (\mathbf{P}_q^k)^{\mathrm{H}} \mathbf{y}_q \in \mathbb{C}^{Q_{kq}\times 1}, \tag{11}$$

where $\mathbf{P}_q^k \in \mathbb{C}^{M_q \times Q_{kq}}$ is a so-called neighbor-specific *fusion matrix*. The dimension of $\mathbf{z}_q^k$, i.e., Q_{qk}, is also neighbor-specific and defined such that $Q_{qk} \leq M_q$.

As mentioned above, all nodes can reach the same performance as an equivalent centralized system if any node q sends an estimate of the contributions of sources in $\mathcal{O}_{kq}$ to node k. To achieve this, the dMWF algorithm alternates between two computing steps: (i) a "discovery step" during which each node $q \in \mathcal{K}$ estimates its own fusion matrices $\{\mathbf{P}_q^k\}_{k\in\mathcal{K}_q}$ and (ii) an "estimation step" during which each node $k \in \mathcal{K}$ uses the incoming fused signals to estimate $\mathbf{d}_k$.

For the sake of clarity, we will first define the estimation step then focus on the discovery step, since an intuition for the latter can be formulated based on the former.

B. Estimation Step

The fused signals defined in (11) are exchanged between each node pair, such that any node $k \in \mathcal{K}$ has access to a so-called *observation vector* $\widetilde{\mathbf{y}}_k$, defined as:

$$\widetilde{\mathbf{y}}_k \triangleq \left[\mathbf{y}_k^{\mathrm{T}} \mid \mathbf{z}_{-k}^{\mathrm{T}}\right] \in \mathbb{C}^{\widetilde{M}_k \times 1}, \forall\, k \in \mathcal{K} \tag{12}$$

$$\text{where } \mathbf{z}_{-k} \triangleq \left[\mathbf{z}_1^{k,\mathrm{T}} \;\ldots\; \mathbf{z}_{k-1}^{k,\mathrm{T}}\, \mathbf{z}_{k+1}^{k,\mathrm{T}} \;\ldots\; \mathbf{z}_K^{k,\mathrm{T}}\right]^{\mathrm{T}} \tag{13}$$

$$\widetilde{M}_k \triangleq M_k + \sum_{q\in\mathcal{K}_k} Q_{qk}. \tag{14}$$

Node k can estimate its target signal $\mathbf{d}_k$ based on $\widetilde{\mathbf{y}}_k$ via the following LMMSE problem:

$$\widetilde{\mathbf{W}}_k \triangleq \underset{\mathbf{W}\in\mathbb{C}^{\widetilde{M}_k\times D}}{\arg\min}\; \mathbb{E}\left\{\left\|\mathbf{d}_k - \mathbf{W}^{\mathrm{H}}\widetilde{\mathbf{y}}_k\right\|_2^2\right\}, \tag{15}$$

the solution of which is also an MWF:

$$\boxed{\widetilde{\mathbf{W}}_k = \mathbf{R}_{\widetilde{\mathbf{y}}_k\widetilde{\mathbf{y}}_k}^{-1}\mathbf{R}_{\widetilde{\mathbf{y}}_k\mathbf{d}_k},} \tag{16}$$

where $\mathbf{R}_{\widetilde{\mathbf{y}}_k\widetilde{\mathbf{y}}_k} \triangleq \mathbb{E}\left\{\widetilde{\mathbf{y}}_k\widetilde{\mathbf{y}}_k^{\mathrm{H}}\right\}$ and $\mathbf{R}_{\widetilde{\mathbf{y}}_k\mathbf{d}_k} \triangleq \mathbb{E}\left\{\widetilde{\mathbf{y}}_k\mathbf{d}_k^{\mathrm{H}}\right\}$. As in the centralized case, these two SCMs may be estimated based on time-averaging. Letting $\widetilde{\mathbf{y}}_k = \widetilde{\mathbf{s}}_k + \widetilde{\mathbf{n}}_k$, where $\widetilde{\mathbf{s}}_k$ and $\widetilde{\mathbf{n}}_k$ are build from only the desired sources or noise sources contributions, respectively, it is possible to write $\mathbf{R}_{\widetilde{\mathbf{y}}_k\mathbf{d}_k} = \mathbb{E}\left\{\widetilde{\mathbf{s}}_k\widetilde{\mathbf{s}}_k^{\mathrm{H}}\right\}\widetilde{\mathbf{E}}_k = (\mathbb{E}\left\{\widetilde{\mathbf{y}}_k\widetilde{\mathbf{y}}_k^{\mathrm{H}}\right\} - \mathbb{E}\left\{\widetilde{\mathbf{n}}_k\widetilde{\mathbf{n}}_k^{\mathrm{H}}\right\})\widetilde{\mathbf{E}}_k$, where $\widetilde{\mathbf{E}}_k \triangleq [\mathbf{E}_{kk}^{\mathrm{T}} \mid \mathbf{0}]^{\mathrm{T}} \in \{0,1\}^{\widetilde{M}_k\times D}$ under the assumption of mutually uncorrelated sources as for (8). From there, a VAD-based strategy may for example be leveraged in speech enhancement scenarios, following a similar time-averaging strategy as in (9).

The estimation filter $\widetilde{\mathbf{W}}_k$ can be partitioned as:

$$\widetilde{\mathbf{W}}_k = \left[\mathbf{W}_{kk}^{\mathrm{T}}\, \mathbf{G}_{k1}^{\mathrm{T}} \;\ldots\; \mathbf{G}_{k,k-1}^{\mathrm{T}}\, \mathbf{G}_{k,k+1}^{\mathrm{T}} \;\ldots\; \mathbf{G}_{kK}^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{17}$$

where $\mathbf{W}_{kk} \in \mathbb{C}^{M_k\times D}$ is applied to $\mathbf{y}_k$ and $\mathbf{G}_{kq} \in \mathbb{C}^{Q_{kq}\times D}$ is applied to $\mathbf{z}_q^k, \forall\, q \in \mathcal{K}_k$. This partitioning becomes useful when formulating the discovery step. The final target signal estimate at node k is obtained as:

$$\widehat{\mathbf{d}}_{k,\mathrm{dMWF}} \triangleq \widetilde{\mathbf{W}}_k^{\mathrm{H}}\widetilde{\mathbf{y}}_k. \tag{18}$$

C. Discovery Step

Consider node pair $(k,q) \in \mathcal{K}\times\mathcal{K}_k$. During a discovery step, the fusion matrices $\mathbf{P}_k^q$ and $\mathbf{P}_q^k$ are to be estimated. In this section, we restrict our attention to $\mathbf{P}_q^k$ since the computation of $\mathbf{P}_k^q$ follows by merely swapping indices q and k.

1) Intuitive Definition

The signal model (1) can be rewritten to isolate the contribution of sources in $\mathcal{O}_{kq}$:

$$\mathbf{y}_q = \mathbf{A}_q\mathbf{s}^{\text{lat}} + \mathbf{B}_q\mathbf{n}^{\text{lat}} + \mathbf{v}_q \tag{19}$$
$$= \left(\mathbf{A}_q^k + \mathbf{A}_q^{\setminus k}\right)\mathbf{s}^{\text{lat}} + \left(\mathbf{B}_q^k + \mathbf{B}_q^{\setminus k}\right)\mathbf{n}^{\text{lat}} + \mathbf{v}_q \tag{20}$$
$$= \mathbf{A}_q^k\mathbf{s}^{\text{lat}} + \mathbf{A}_q^{\setminus k}\mathbf{s}^{\text{lat}} + \mathbf{B}_q^k\mathbf{n}^{\text{lat}} + \mathbf{B}_q^{\setminus k}\mathbf{n}^{\text{lat}} + \mathbf{v}_q, \tag{21}$$

where $\mathbf{A}_q^k$ and $\mathbf{A}_q^{\setminus k}$ are equal to $\mathbf{A}_q$ with the columns corresponding to sources outside of $\mathcal{O}_{kq}$ or in $\mathcal{O}_{kq}$ set to zero, respectively, and similarly for $\mathbf{B}_q^k$ and $\mathbf{B}_q^{\setminus k}$. Equation (21) can be further rewritten by discarding the all-zero columns of the previously defined matrices:

$$\mathbf{y}_q = \underbrace{\underline{\mathbf{A}}_q^k\mathbf{s}_{qk}^{\text{lat}} + \underline{\mathbf{B}}_q^k\mathbf{n}_{qk}^{\text{lat}}}_{\triangleq \mathbf{g}_q^k\ (\text{sources} \in \mathcal{O}_{kq})} + \underbrace{\underline{\mathbf{A}}_q^{\setminus k}\mathbf{s}_{q\setminus k}^{\text{lat}} + \underline{\mathbf{B}}_q^{\setminus k}\mathbf{n}_{q\setminus k}^{\text{lat}} + \mathbf{v}_q}_{=\mathbf{y}_q - \mathbf{g}_q^k\ (\text{sources} \notin \mathcal{O}_{kq})}, \tag{22}$$

where $\mathbf{s}_{qk}^{\text{lat}} \in \mathbb{C}^{\mathcal{O}_{\text{d},kq}\times 1}$ and $\mathbf{s}_{q\setminus k}^{\text{lat}} \in \mathbb{C}^{(Q_{\text{d}}-\mathcal{O}_{\text{d},kq})\times 1}$ contain the latent signals of the desired sources in $\mathcal{O}_{\text{d},kq}$ or outside of $\mathcal{O}_{\text{d},kq}$, respectively. The matrices $\underline{\mathbf{A}}_q^k \in \mathbb{C}^{M_q\times|\mathcal{O}_{\text{d},kq}|}$ and $\underline{\mathbf{A}}_q^{\setminus k} \in \mathbb{C}^{M_q\times(Q_{\text{d}}-|\mathcal{O}_{\text{d},kq}|)}$ are equal to $\mathbf{A}_q^k$ and $\mathbf{A}_q^{\setminus k}$, respectively, without all-zeros columns. Analogous definitions are adopted for $\mathbf{n}_{qk}^{\text{lat}}$, $\mathbf{n}_{q\setminus k}^{\text{lat}}$, $\underline{\mathbf{B}}_q^k$, and $\underline{\mathbf{B}}_q^{\setminus k}$. Note that, since all sources are mutually uncorrelated and $\underline{\mathbf{A}}_q^k$ and $\underline{\mathbf{B}}_q^k$ only relate to sources observed by q, they are both full-column-rank.

During the discovery step, focusing on the neighbor k of q, the goal of node q is to estimate enough channels of $\mathbf{g}_q^k$ to entirely span the $\mathcal{O}_{kq}$-dimensional latent subspace to which the signals generated by sources $\mathcal{O}_{kq}$ belong. For conciseness, we let $Q_{kq} \triangleq |\mathcal{O}_{kq}|, \forall\,(k,q) \in \mathcal{K}\times\mathcal{K}_q$. A first intuitive definition of the fusion matrix to be used in (11) may be found by considering the LMMSE problem that node q formulates to estimate, w.l.o.g., the first Q_{kq} channels of $\mathbf{g}_q^k$:

$$\underset{\mathbf{P}\in\mathbb{C}^{M_q\times Q_{kq}}}{\arg\min}\ \mathbb{E}\left\{\left\|\mathbf{E}_{q\to k}^{\text{T}}\mathbf{g}_q^k - \mathbf{P}^{\text{H}}\mathbf{y}_q\right\|_2^2\right\} = \mathbf{R}_{\mathbf{y}_q\mathbf{y}_q}^{-1}\mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k}, \tag{23}$$

where $\mathbf{E}_{q\to k} \triangleq [\mathbf{I}_{Q_{kq}} \mid \mathbf{0}]^{\text{T}} \in \{0,1\}^{M_q\times Q_{kq}}$ is a selection matrix and $\mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k} \triangleq \mathbb{E}\left\{\mathbf{y}_q(\mathbf{g}_q^k)^{\text{H}}\right\}\mathbf{E}_{q\to k} \in \mathbb{C}^{M_q\times Q_{kq}}$. If node q could perfectly estimate $\mathbf{E}_{q\to k}^{\text{T}}\mathbf{g}_q^k$ and send it to node k, then node k will have access to all the information needed to optimally estimate $\mathbf{d}_k$ based on $\mathbf{y}_q$ and $\mathbf{z}_q^k$. However, accurately estimating $\mathbf{E}_{q\to k}^{\text{T}}\mathbf{g}_q^k$ through (23) without any knowledge of the signals at node k is extremely challenging for node q. The main purpose of the dMWF discovery step becomes clear at this point: overcoming this difficulty.

2) Practical Solution

In this section, we will describe how the dMWF discovery step allows node q to estimate $\mathbf{P}_q^k$ in a practical manner, i.e., without requiring node q to estimate $\mathbf{E}_{q\to k}^{\text{T}}\mathbf{g}_q^k$ directly. We first let node k transmit Q_{kq} local sensor signals to node q. Note that this is only occurring during the discovery step, and that node k will not need to transmit these signals to node q during the estimation step (instead, during the estimation step, node k will transmit the Q_{kq}-dimensional fused signal $\mathbf{z}_k^q$ to node q). Consider, w.l.o.g., that the first Q_{kq} channels of $\mathbf{y}_k$ are selected and sent to node q:

$$\mathbf{y}_{k\to q} \triangleq \mathbf{E}_{k\to q}^{\text{T}}\mathbf{y}_k \in \mathbb{C}^{Q_{kq}\times 1}. \tag{24}$$

Note that (24) assumes that node k knows the value Q_{kq}. The estimation of Q_{kq} is a source enumeration problem, which may be addressed at each node through well-established techniques [35]–[38]. As we are concerned with the estimation of the number of sources observed by both k and q, the estimation of Q_{kq} itself requires nodes to exchange (some) local sensor signals. An example of methodology may be to allow node k to transmit all a single local sensor signal to node q at first, for a sufficient amount of time for node q to estimate Q_{kq} using one of the aforementioned techniques, then accordingly adapt the number of signals that node k sends to node q to properly conduct the dMWF discovery step.

With access to $\mathbf{y}_{k\to q}$, node q is able to estimate the SCM $\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}} \triangleq \mathbb{E}\left\{\mathbf{y}_q\mathbf{y}_{k\to q}^{\text{H}}\right\} \in \mathbb{C}^{M_q\times Q_{kq}}$. This SCM can be expanded as:

$$\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}} = \mathbb{E}\left\{\mathbf{y}_q\mathbf{y}_k^{\text{H}}\right\}\mathbf{E}_{k\to q} \tag{25}$$
$$= \left(\mathbb{E}\left\{\mathbf{s}_q\mathbf{s}_k^{\text{H}}\right\} + \mathbb{E}\left\{\mathbf{n}_q\mathbf{n}_k^{\text{H}}\right\}\right)\mathbf{E}_{k\to q} \tag{26}$$
$$= \left(\underline{\mathbf{A}}_q^k\mathbf{R}_{\mathbf{ss},qk}^{\text{lat}}(\underline{\mathbf{A}}_k^q)^{\text{H}} + \underline{\mathbf{B}}_q^k\mathbf{R}_{\mathbf{nn},qk}^{\text{lat}}(\underline{\mathbf{B}}_k^q)^{\text{H}}\right)\mathbf{E}_{k\to q} \tag{27}$$

with the newly defined SCMs $\mathbf{R}_{\mathbf{ss},qk}^{\text{lat}} \triangleq \mathbb{E}\{\mathbf{s}_{qk}^{\text{lat}}(\mathbf{s}_{qk}^{\text{lat}})^{\text{H}}\}$ and $\mathbf{R}_{\mathbf{nn},qk}^{\text{lat}} \triangleq \mathbb{E}\{\mathbf{n}_{qk}^{\text{lat}}(\mathbf{n}_{qk}^{\text{lat}})^{\text{H}}\}$. For simplicity it is assumed in (27) that latent sources are always ordered such that $\mathbf{s}_{qk}^{\text{lat}} = \mathbf{s}_{kq}^{\text{lat}}$ and $\mathbf{n}_{qk}^{\text{lat}} = \mathbf{n}_{kq}^{\text{lat}}$. Note that (26) holds since desired, noise, and self-noise components are mutually uncorrelated, and that (27) holds because only the contributions of sources in $\mathcal{O}_{\text{d},kq}$ are correlated between node k and q. The desired and noise terms in (27) can be combined, yielding:

$$\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}} = \underline{\mathbf{C}}_q^k\mathbf{R}_{qk}^{\text{lat}}(\underline{\mathbf{C}}_k^q)^{\text{H}}\mathbf{E}_{k\to q} \tag{28}$$

where $\underline{\mathbf{C}}_q^k \triangleq \left[\underline{\mathbf{A}}_q^k \mid \underline{\mathbf{B}}_q^k\right]$, $\underline{\mathbf{C}}_k^q \triangleq [\underline{\mathbf{A}}_k^q \mid \underline{\mathbf{B}}_k^q]$, and $\mathbf{R}_{qk}^{\text{lat}} \triangleq \text{blkdiag}\left\{\mathbf{R}_{\mathbf{ss},qk}^{\text{lat}}, \mathbf{R}_{\mathbf{nn},qk}^{\text{lat}}\right\}$. Note that, since $\underline{\mathbf{A}}_q^k$ and $\underline{\mathbf{B}}_q^k$ are both full-column-rank and sources are mutually uncorrelated, $\underline{\mathbf{C}}_q^k$ is full-column-rank, $\forall\,(k,q) \in \mathcal{K}\times\mathcal{K}_q$.

The SCM $\mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k}$ introduced in (23) can be written in a similar fashion using (22):

$$\mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k} = \underline{\mathbf{C}}_q^k\mathbf{R}_{qk}^{\text{lat}}(\underline{\mathbf{C}}_k^q)^{\text{H}}\mathbf{E}_{q\to k}. \tag{29}$$

In both (28) and (29), $\underline{\mathbf{C}}_q^k\mathbf{R}_{qk}^{\text{lat}}$ is right-multiplied by a full-rank $Q_{kq}\times Q_{kq}$ matrix. Indeed, $(\underline{\mathbf{C}}_k^q)^{\text{H}}\mathbf{E}_{k\to q}$ in (28) is full-rank since $\underline{\mathbf{C}}_k^q$ is full-column-rank, and similarly for $(\underline{\mathbf{C}}_q^k)^{\text{H}}\mathbf{E}_{q\to k}$ in (29). Consequently, it holds that:

$$\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}} = \mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k}\mathbf{X}_{kq} \tag{30}$$

$$\text{with } \mathbf{X}_{kq} \triangleq \left((\underline{\mathbf{C}}_q^k)^{\mathrm{H}}\mathbf{E}_{q\to k}\right)^{-1}(\underline{\mathbf{C}}_k^q)^{\mathrm{H}}\mathbf{E}_{k\to q}. \tag{31}$$

In an idealistic scenario where $\mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k}$ would be known a priori or possible to estimate, it could be directly used to compute the dMWF fusion matrix $\mathbf{P}_q^k$ using (23). The resulting fused signal $\mathbf{z}_q^k$ would then be transformed by $\mathbf{G}_{kq}$ at node k during the estimation step (cf. Section III-B and (17)). Notice that, if the fused signal $\mathbf{z}_q^k$ were to be linearly transformed by an arbitrary full-rank $Q_{qk}\times Q_{qk}$ matrix, the final target signal estimate $\widehat{\mathbf{d}}_k$ obtained via (18) would not be affected. Indeed, if $\mathbf{P}_{q,\text{alt}}^k \triangleq \mathbf{P}_q^k\mathbf{X}$ with $\mathbf{X}\in\mathbb{C}^{Q_{kq}\times Q_{kq}}$ full-rank was used instead of $\mathbf{P}_q^k$, (15) would simply yield $\mathbf{G}_{kq,\text{alt}} \triangleq \mathbf{X}^{-1}\mathbf{G}_{kq} \in \mathbb{C}^{Q_{kq}\times D}$ instead of $\mathbf{G}_{kq}$, which would lead to the same $\widehat{\mathbf{d}}_k$ when applying $\widetilde{\mathbf{W}}_k^{\mathrm{H}}$ to $\widetilde{\mathbf{y}}_k$ (i.e., when applying $\mathbf{G}_{kq,\text{alt}}^{\mathrm{H}}$ to $(\mathbf{P}_{q,\text{alt}}^k)^{\mathrm{H}}\mathbf{y}_q$).

Consequently, using $\mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k}\mathbf{X}_{kq}$ instead of $\mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k}$ in (23) would lead, after the estimation step, to the same optimal target signal estimation performance. Unlike $\mathbf{R}_{\mathbf{y}_q\mathbf{g}_q^k}$, $\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}}$ can be straightforwardly estimated by node q using time-averaging over the available signal $\mathbf{y}_{k\to q}$. A practical solution to the dMWF discovery step is thus possible to formulate by defining $\mathbf{P}_q^k$ as:

$$\boxed{\mathbf{P}_q^k \triangleq \mathbf{R}_{\mathbf{y}_q\mathbf{y}_q}^{-1}\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}},} \tag{32}$$

This fusion matrix estimation is performed by all nodes in the WASN, such that each node $q\in\mathcal{K}$ has access to $\{\mathbf{P}_q^k\}_{k\in\mathcal{K}_q}$. Each node $q\in\mathcal{K}$ then computes $\mathbf{z}_q^k$ and sends it to node $k, \forall\, k\in\mathcal{K}_q$. This ends the discovery step.

It can be noted that, interestingly, (32) is the MWF solution to an LMMSE problem where node q attempts to estimate $\mathbf{y}_{k\to q}$ from $\mathbf{y}_q$:

$$\mathbf{P}_q^k = \underset{\mathbf{P}\in\mathbb{C}^{M_q\times Q_{kq}}}{\arg\min}\ \mathbb{E}\left\{\left\|\mathbf{y}_{k\to q} - \mathbf{P}^{\mathrm{H}}\mathbf{y}_q\right\|_2^2\right\}. \tag{33}$$

D. dMWF Overview and Optimality

The dMWF algorithm is summarized in Fig. 2 and Algorithm 1.

Theorem 1: under the signal model (1), the dMWF target signal estimate is equal to the centralized target signal estimate at node $k, \forall\, k\in\mathcal{K}$, i.e., $\widetilde{\mathbf{W}}_k^{\mathrm{H}}\widetilde{\mathbf{y}}_k = \widehat{\mathbf{W}}_k^{\mathrm{H}}\mathbf{y}, \forall\, k\in\mathcal{K}$, when $\mathbf{P}_q^k$ is defined as in (32), $\forall\,(k,q)\in\mathcal{K}\times\mathcal{K}_k$.

Proof: cf. Appendix A.

Unlike the DANSE family of algorithms [21]–[24], [39], the dMWF algorithm considers a more general case where $\mathbf{A}_k$ in (1) may not necessarily be full-rank (i.e., it may have all-zeros columns). The assumption "$\mathbf{A}_k$ full-rank, $\forall k\in\mathcal{K}$", omnipresent in the DANSE family of algorithms, is not necessary for Theorem 1 to hold.

In any given time-frame, the number of signal channels exchanged through an FC WASN where dMWF is deployed

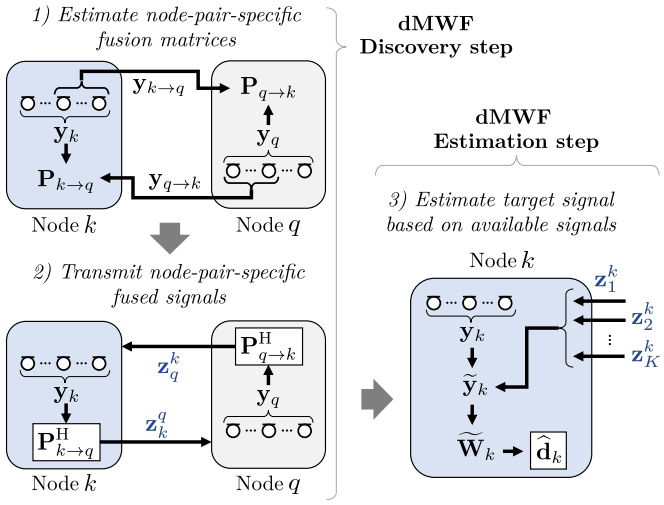

FIGURE 2. Schematic representation of the dMWF algorithmic steps, focusing on two nodes k and q.

Algorithm 1 dMWF algorithm

for time frame $l = 0, 1, \ldots$ **do**
  Discovery Step:
  for $q\in\mathcal{K}$ **do**
    Node q estimates $\mathbf{R}_{\mathbf{y}_q\mathbf{y}_q}$ using $\mathbf{y}_q[l]$.
    for $k\in\mathcal{K}_q$ **do**
      Node q receives $\mathbf{y}_{k\to q}[l]$ from k.
      Node q estimates $\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}}$.
      Node q computes $\mathbf{P}_q^k$ as in (32).
      Node q computes $\mathbf{z}_q^k[l]$ as in (11) and sends to k.
    end for
  end for
  Estimation Step:
  for $k\in\mathcal{K}$ **do**
    Node k gathers $\{\mathbf{z}_q^k[l]\}_{q\in\mathcal{K}_k}$.
    Node k builds $\widetilde{\mathbf{y}}_k[l]$ as in (12).
    Node k estimates $\mathbf{R}_{\widetilde{\mathbf{y}}_k\widetilde{\mathbf{y}}_k}$ and $\mathbf{R}_{\widetilde{\mathbf{y}}_k\mathbf{d}_k}$.
    Node k computes $\widetilde{\mathbf{W}}_k$ as in (16).
    Node k computes $\widehat{\mathbf{d}}_{k,\text{dMWF}}[l]$ as in (18).
  end for
end for

is $\overline{Q} \triangleq \sum_{k\in\mathcal{K}}\sum_{q\in\mathcal{K}_k} Q_{kq}$. With respect to a fusion-center-free scenario where every node would transmit all their raw local sensor signals to all other nodes, this corresponds to a compression factor of $K(K-1)M/\overline{Q}$. In a scenario with fully overlapping subspaces where every node observes every sources in the environment, $Q_{kq} = Q, \forall\,(k,q)$ and the compression factor becomes $K(K-1)M/(K(K-1)Q) = M/Q$. This value is lower than the DANSE compression factor M/Q_{d} [21], which highlights a trade-off between adaptivity and communication bandwidth usage. Indeed, while DANSE requires several iterations to converge and is not guaranteed to converge in a scenario with partially

overlapping subspaces, the dMWF algorithm converges in a single iteration and is guaranteed to converge in any scenario with partially overlapping subspaces, at the cost of a higher communication bandwidth usage (yet still generally lower than the one required by an FC WASN where every node transmits all their local sensor signals to every other node).

IV. Distributed MWF in non-FC WASNs: TI-dMWF

In Section III, we only considered FC WASNs where every node can directly communicate with every other node. In practice, WASN topologies can vary due to link failures and nodes entering or leaving the network. Topologies may not be FC and may even be time-varying. In this section, we defined the TI-dMWF algorithm, which can function in any WASNs with potentially non-FC and time-varying topologies. The TI-dMWF algorithm shares similarities with dMWF but differs in the way nodes exchange information and in the way the WASN topology is handled.

In a non-FC WASN, the TI-dMWF algorithm achieves optimality while nodes only exchange low-dimensional signals. This is done through cascaded estimation-fusion processes, as will be defined in this section. To enable this cascade of operations, the WASN is first pruned to a tree, which can be done in various ways depending on the constraints of the problem. We here focus on the common choice of minimum spanning trees (MSTs) in order to minimize the total distance between connected nodes, and consequently further reduce the transmit power usage of the algorithm [23].

An MST can be computed from any given WASN topology using, e.g., Prim's algorithm [40]. In the TI-dMWF algorithm, the tree resulting from tree-pruning can be oriented by setting a particular node as the root. In a rooted tree, non-root nodes that have only one neighbor are referred to as leaf nodes. Each non-root node in the tree has one downstream neighbor (towards the root node) and each non-leaf has at least one upstream neighbor (towards the leaf nodes). We let $\mathcal{U}_q$ denote the set of upstream neighbors of node $q, \forall\, q \in \mathcal{K}$ and $\overline{\mathcal{U}}_q$ denote the set of all nodes upstream of q. It holds that $\mathcal{U}_q \subseteq \overline{\mathcal{U}}_q$. An example of a tree-pruning operation is illustrated in Fig. 3.

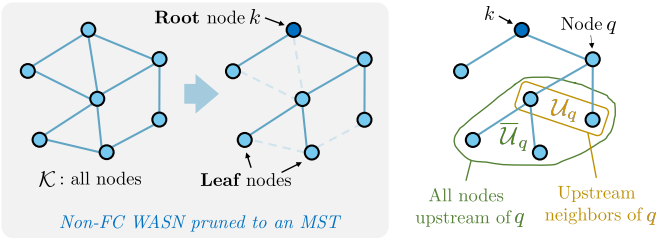

FIGURE 3. Tree-pruning example with nodes sets notation.

Consider a node $k \in \mathcal{K}$ and let it be the root of the tree resulting from tree-pruning of the original (possibly non-FC) WASN topology. We aim at obtaining an observation vector $\widetilde{\mathbf{y}}_k$ at node k which can be used to estimate $\mathbf{d}_k$ through an LMMSE problem, similarly to (15). For that purpose, a sum-and-send data flow starting from the leaf nodes and ending at the root node k is defined. In order to outline this sum-and-send data flow, we focus on one non-root node $q \in \mathcal{K}_k$ in the tree topology. We first define the set $\mathcal{C}_q^k$ of (desired or noise) sources observed by either (i) both root node k and node q or by (ii) both k and any node in $\overline{\mathcal{U}}_q$:

$$\mathcal{C}_q^k \triangleq \mathcal{O}_{kq} \cup \mathcal{O}_{kn_1} \cup \cdots \cup \mathcal{O}_{kn_U} \tag{34}$$

where $\{n_1, \ldots, n_U\} = \overline{\mathcal{U}}_q$ is the set of nodes upstream of q. The definition (34) implies that $|\mathcal{C}_q^k| \leq |\mathcal{O}_k|, \forall\, q \in \mathcal{K}_k$. For the sake of simplicity of exposition, we assume that $M_k \geq |\mathcal{C}_q^k|, \forall\, (k,q) \in \mathcal{K} \times \mathcal{K}_k$. Adaptation strategies in scenarios where this does not hold are provided in a later subsection. ← This is not written yet. It is now possible to define the node-specific fused signal dimension $\widehat{Q}_{kq} \triangleq \left|\mathcal{C}_q^k\right|, \forall\, q \in \mathcal{K}_k$. Note that, for any node q' downstream of q, $|\mathcal{C}_q^k| \geq |\mathcal{C}_{q'}^k|$ and thus $\widehat{Q}_{kq} \geq \widehat{Q}_{kq'}$. In other words, the number of fused signal channels transmitted downstream may increase as we approach the root node k, since the number of commonly observed sources may increase.

In the TI-dMWF algorithm, node q will send a signal $\widehat{\mathbf{z}}_q^k \in \mathbb{C}^{\widehat{Q}_{kq} \times 1}$ to its (only) downstream neighbor. This signal is recursively defined as a linear combination of the local sensor signals $\mathbf{y}_q$ and of the signals received by node q from its own upstream neighbors, i.e.:

$$\widehat{\mathbf{z}}_q^k \triangleq (\widehat{\mathbf{P}}_q^k)^{\mathrm{H}} \widehat{\mathbf{y}}_q, \tag{35}$$

$$\text{with } \widehat{\mathbf{y}}_q \triangleq \left[\mathbf{y}_q^{\mathrm{T}}, (\widehat{\mathbf{z}}_{u_1}^q)^{\mathrm{T}}, \ldots, (\widehat{\mathbf{z}}_{u_B}^q)^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{C}^{\widehat{M}_q \times 1}, \tag{36}$$

$$\text{where } \widehat{M}_q \triangleq M_q + \sum_{i=1}^{B} \widehat{Q}_{ku_i} \text{ and } \{u_1, \ldots, u_B\} = \mathcal{U}_q. \tag{37}$$

For notation simplicity, we have omitted to refer to the root node k in $\widehat{\mathbf{y}}_q$ and $\widehat{M}_q$, but it is understood that these vectors and matrices are defined with respect to the root node k. When the root node changes, so do the definitions of $\widehat{\mathbf{y}}_q$ and $\widehat{M}_q$, depending on the new sets $\{\mathcal{C}_q^k\}_{q \in \mathcal{K}_k}$.

While the dMWF algorithm can be divided in two distinct steps (cf. Section III), the TI-dMWF algorithm functions in three steps. First, an *upstream data flow* is performed where low-dimensional signals are transmitted from the root node k to the leaf nodes, allowing the latter to estimate their fusion matrices. Second, a *downstream data flow* is performed where low-dimensional signals are transmitted from the leaf nodes to the root node k, allowing fused signals $\widehat{\mathbf{z}}_q^k$ to be computed and sent towards the root. Finally, the root node is able to optimally estimate its target signal $\mathbf{d}_k$ using the observation vector $\widehat{\mathbf{y}}_k$ in an estimation step similar to the one defined for the dMWF algorithm (cf. Section III-B).

A. Upstream Data Flow

We will first focus on the upstream data flow performed from the root node k to the leaf nodes in the tree topology.

The upstream data flow is initiated by the root node k, which sends a signal $\mathbf{y}_{k\uparrow q}$ to each of its upstream neighbors $q, \forall q \in \mathcal{U}_k$. This signal is defined as a subset of the local sensor signals $\mathbf{y}_k$ at node k:

$$\mathbf{y}_{k\uparrow q} \triangleq \mathbf{E}_{\uparrow q}^{\mathrm{T}} \mathbf{y}_k \in \mathbb{C}^{\widehat{Q}_{kq} \times 1}, \tag{38}$$

where $\mathbf{E}_{\uparrow q} \triangleq \left[\mathbf{I}_{\widehat{Q}_{kq}} \,|\, \mathbf{0}\right]^{\mathrm{T}} \in \{0,1\}^{M_k \times \widehat{Q}_{kq}}$ is a selection matrix. This assumes that node k knows the values $\{\widehat{Q}_{kq}\}_{q\in\mathcal{U}_k}$ a priori or can estimate them. Much like for Q_{kq} in the FC case, estimation of $\{\widehat{Q}_{kq}\}_{q\in\mathcal{U}_k}$ is a source enumeration problem which may be addressed through the same techniques as mentioned in (III-C), i.e. [35]–[38].

Once node k has sent $\mathbf{y}_{k\uparrow q}$ to each of its upstream neighbors $q \in \mathcal{U}_k$, each node q in $\mathcal{U}_k$ sends $\mathbf{y}_{k\uparrow q'}$ to its own upstream neighbor $q', \forall\, q' \in \mathcal{U}_q$. The process continues until every non-root node, i.e., every $q \in \mathcal{K}_k$, has access to $\mathbf{y}_{k\uparrow q}$. The upstream data flow is schematically depicted in the top-left part of Fig. 4.

B. Downstream Data Flow

Consider a leaf node l in the tree topology. Node l has access to $\mathbf{y}_{k\uparrow l}$ and to its own local sensor signals $\mathbf{y}_l$. As a leaf node has no upstream neighbors, its TI-dMWF observation vector as defined in (36) is simply $\widehat{\mathbf{y}}_l = \mathbf{y}_l$, with dimension $\widehat{M}_l = M_l$. Using $\mathbf{y}_{k\uparrow l}$, node l can estimate the following SCM:

$$\mathbf{R}_{\widehat{\mathbf{y}}_l \mathbf{y}_{k\uparrow l}} \triangleq \mathbb{E}\left\{\widehat{\mathbf{y}}_l \mathbf{y}_{k\uparrow l}^{\mathrm{H}}\right\} = \mathbb{E}\left\{\mathbf{y}_l \mathbf{y}_k^{\mathrm{H}}\right\} \mathbf{E}_{\uparrow q} \in \mathbb{C}^{\widehat{M}_l \times \widehat{Q}_{kl}}. \tag{39}$$

In $\mathbf{y}_{k\uparrow l}$, contributions from sources in $\mathcal{O}_k$ are present (cf. (38)) while, in $\widehat{\mathbf{y}}_l$, contributions from sources in $\mathcal{O}_l$ are present. Only the signals correlated between $\widehat{\mathbf{y}}_l$ and $\mathbf{y}_{k\uparrow l}$ will remain in $\mathbf{R}_{\widehat{\mathbf{y}}_l \mathbf{y}_{k\uparrow l}}$, which corresponds to the contributions of sources in $\mathcal{O}_{kl}$. Note that $\mathcal{O}_{kl} = \mathcal{C}_l^k$ since $\overline{\mathcal{U}}_l = \emptyset$, cf. (34). Following a similar logic as for the dMWF algorithm, node l can then define its TI-dMWF fusion matrix towards root node k as:

$$\widehat{\mathbf{P}}_l^k \triangleq \mathbf{R}_{\widehat{\mathbf{y}}_l \widehat{\mathbf{y}}_l}^{-1} \mathbf{R}_{\widehat{\mathbf{y}}_l \mathbf{y}_{k\uparrow l}} \in \mathbb{C}^{\widehat{M}_l \times \widehat{Q}_{kl}}, \forall \text{ leaf node } l. \tag{40}$$

Leaf node l then computes $\widehat{\mathbf{z}}_l^k$ via (35) and sends it to its only downstream neighbor in the tree, denoted by q. All leaf nodes $l \in \mathcal{U}_q$ perform this operation, such that node q receives $\{\widehat{\mathbf{z}}_l^k\}_{l\in\mathcal{U}_q}$ and can build its own observation vector $\widehat{\mathbf{y}}_q$ as in (36). During the upstream data flow, node q has received $\mathbf{y}_{k\uparrow q}$ and can thus compute the SCM $\mathbf{R}_{\widehat{\mathbf{y}}_q \mathbf{y}_{k\uparrow q}}$ in a similar way as for leaf nodes:

$$\mathbf{R}_{\widehat{\mathbf{y}}_q \mathbf{y}_{k\uparrow q}} \triangleq \mathbb{E}\left\{\widehat{\mathbf{y}}_q \mathbf{y}_{k\uparrow q}^{\mathrm{H}}\right\} = \mathbb{E}\left\{\widehat{\mathbf{y}}_q \mathbf{y}_k^{\mathrm{H}}\right\} \mathbf{E}_{\uparrow q} \in \mathbb{C}^{\widehat{M}_q \times \widehat{Q}_{kq}} \tag{41}$$

In $\mathbf{y}_{k\uparrow q}$, contributions from sources in $\mathcal{O}_k$ are present (cf. (38)) while, in $\widehat{\mathbf{y}}_q$, contributions from sources in $\mathcal{O}_q \cup \mathcal{C}_{u_1}^k \cup \cdots \cup \mathcal{C}_{u_B}^k$ are present (where $\{u_1, \ldots, u_B\} = \mathcal{U}_q$, cf. (36)). Only the signals correlated between $\widehat{\mathbf{y}}_q$ and $\mathbf{y}_{k\uparrow q}$ will remain in $\mathbf{R}_{\widehat{\mathbf{y}}_q \mathbf{y}_{k\uparrow q}}$, which corresponds to the contributions of sources in $\mathcal{C}_q^k$ as defined in (34). Node q can then also build its TI-dMWF fusion matrix as:

$$\boxed{\widehat{\mathbf{P}}_q^k \triangleq \mathbf{R}_{\widehat{\mathbf{y}}_q \widehat{\mathbf{y}}_q}^{-1} \mathbf{R}_{\widehat{\mathbf{y}}_q \mathbf{y}_{k\uparrow q}}, \forall\, q \in \mathcal{K}_k.} \tag{42}$$

This downstream data flow is repeated from leaf nodes to the root node k, such that each node $q \in \mathcal{K}_k$ estimates its fusion matrix $\widehat{\mathbf{P}}_q^k$ and sends $\mathbf{z}_q^k$ downstream. The downstream data flow is schematically depicted in the top-right and bottom parts of Fig. 4.

C. Estimation Step

At the end of the TI-dMWF downstream data flow, root node k has access to $\{\widehat{\mathbf{z}}_q^k\}_{q\in\mathcal{U}_k}$ and can build its own observation vector $\widehat{\mathbf{y}}_k$ as in (36). Node k can then estimate its target signal based on $\widehat{\mathbf{y}}_k$ via:

$$\widetilde{\mathbf{W}}_k^{\mathrm{TI}} \triangleq \underset{\mathbf{W}\in\mathbb{C}^{\widehat{M}_k \times D}}{\arg\min}\ \mathbb{E}\left\{\left\|\mathbf{d}_k - \mathbf{W}^{\mathrm{H}} \widehat{\mathbf{y}}_k\right\|_2^2\right\}, \tag{43}$$

$$\Leftrightarrow \boxed{\widetilde{\mathbf{W}}_k^{\mathrm{TI}} = \mathbf{R}_{\widehat{\mathbf{y}}_k \widehat{\mathbf{y}}_k}^{-1} \mathbf{R}_{\widehat{\mathbf{y}}_k \mathbf{d}_k},} \tag{44}$$

where $\mathbf{R}_{\widehat{\mathbf{y}}_k \mathbf{d}_k} \triangleq \mathbb{E}\left\{\widehat{\mathbf{y}}_k \mathbf{d}_k^{\mathrm{H}}\right\}$. In the same way as for the dMWF algorithm, the two SCMs necessary to compute (44) may be estimated based on time-averaging, following an analogous reasoning as that available under (16). The TI-dMWF target signal estimate at node k is obtained as:

$$\widehat{\mathbf{d}}_{k,\text{TI-dMWF}} \triangleq (\widetilde{\mathbf{W}}_k^{\mathrm{TI}})^{\mathrm{H}} \widehat{\mathbf{y}}_k. \tag{45}$$

Note that each part of the TI-dMWF algorithm is specific to one root node k and only allows the estimation of the target signal $\mathbf{d}_k$ at this root node. The TI-dMWF downstream data flow, upstream data flow, and estimation step must thus be repeated for each node $k \in \mathcal{K}$, each time setting it as the root of the tree (the sets $\{\mathcal{C}_q^k\}_{q\in\mathcal{K}_k}$ will change accordingly).

D. TI-dMWF Overview and Optimality

The TI-dMWF algorithm is summarized in Fig. 4 and Algorithm 2.

Theorem 2: under the signal model (1), the TI-dMWF target signal estimate is equal to the centralized target signal estimate at node $k, \forall\, k \in \mathcal{K}$, i.e., $(\widetilde{\mathbf{W}}_k^{\mathrm{TI}})^{\mathrm{H}} \widehat{\mathbf{y}}_k = \widehat{\mathbf{W}}_k^{\mathrm{H}} \mathbf{y}, \forall\, k \in \mathcal{K}$, when $\widehat{\mathbf{P}}_q$ is defined as in (42), $\forall\, q \in \mathcal{K}$.

Proof: cf. Appendix B.

In any given time-frame, the number of signal channels exchanged through a non-FC WASN where TI-dMWF is

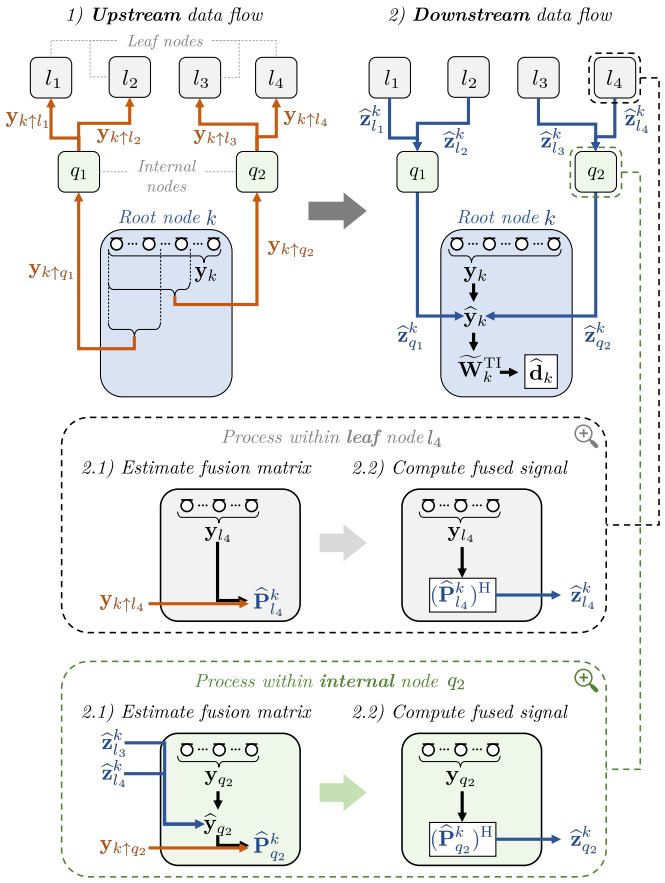

FIGURE 4. Schematic representation of the TI-dMWF algorithm directed towards root node k.

deployed is $\overline{Q}_{\text{TI}} \triangleq \sum_{k\in\mathcal{K}} \sum_{q\in\mathcal{K}_k} \widehat{Q}_{kq}$[1]. In a scenario with fully overlapping subspaces where every node observes every sources in the environment, the compression factor with respect to a fusion-center-free scenario where every node would transmit all their raw local sensor signals to all other nodes again becomes M/Q (as for dMWF), which is lower in DANSE [21] and possibly significantly lower than the compression factor of $K(K-1)M/(2(K-1)Q) = KM/(2Q)$ of TI-DANSE [23]. However, this observation again highlights a trade-off between adaptivity and communication bandwidth usage, as TI-DANSE requires many iterations to converge and is not guaranteed to converge in a scenario with partially overlapping subspaces, while TI-dMWF converges in a single iteration and is guaranteed to converge in any scenario with partially overlapping subspaces, at the cost of a higher communication bandwidth usage (yet still generally lower than the one required by a non-FC WASN where every node transmits all their local sensor signals to every other node).

Option to include a section on (TI-)GEVD-dMWF (without optimality proof) - similar to what was done in the TI-DANSE$^+$ paper.

[1]Note that the number of signals exchanged during the operation of TI-dMWF may be reduced, depending on the particular topology. Indeed, the same signals may be sent several times through the up- and downstream data flows aiming at different root nodes. If the topology is known in advance, this may be avoided by identifying, for any given root node, which signals do need to be transmitted and which are already available from previous up- and downstream data flows aimed at other root nodes. This is, however, not formally developed here.

Algorithm 2 TI-dMWF algorithm

for time frame $l = 0, 1, \ldots$ **do**
  for $k \in \mathcal{K}$ **do**
    Upstream Data Flow:
    for q from root to leaf nodes (excluded) **do**
      Node q sends $\mathbf{y}_{k\uparrow q'}[l]$ to $q', \forall\, q' \in \mathcal{U}_q$.
    end for
    Downstream Data Flow:
    for q from leaves to root k (excluded) **do**
      Node q gathers $\{\widehat{\mathbf{z}}^k_{q'}[l]\}_{q'\in\mathcal{U}_q}$.
      Node q builds $\widehat{\mathbf{y}}_q[l]$ as in (36).
      Node q estimates $\mathbf{R}_{\widehat{\mathbf{y}}_q\widehat{\mathbf{y}}_q}$ using $\widehat{\mathbf{y}}_q[l]$.
      Node q estimates $\mathbf{R}_{\widehat{\mathbf{y}}_q\mathbf{y}_{k\uparrow q}}$ using $\mathbf{y}_{k\uparrow q}[l]$.
      Node q computes $\widehat{\mathbf{P}}^k_q$ as in (42).
      Node q computes $\widehat{\mathbf{z}}^k_q[l]$ as in (35) and sends it downstream.
    end for
    Estimation Step:
    Node k gathers $\{\widehat{\mathbf{z}}^k_q[l]\}_{q\in\mathcal{U}_k}$.
    Node k builds $\widehat{\mathbf{y}}_k[l]$ as in (36).
    Node k estimates $\mathbf{R}_{\widehat{\mathbf{y}}_k\widehat{\mathbf{y}}_k}$ and $\mathbf{R}_{\widehat{\mathbf{y}}_k\mathbf{d}_k}$.
    Node k computes $\widetilde{\mathbf{W}}^{\text{TI}}_k$ as in (44).
    Node k computes $\widehat{\mathbf{d}}_{k,\text{TI-dMWF}}[l]$ as in (45).
  end for
end for

V. Simulated Results

NB: as of 30/05/2025, the figures in this section have yet to be generated.

In this section, we provide simulated validation of the proposed dMWF and TI-dMWF algorithms in comparison to centralized processing and to other state-of-the-art distributed signal estimation algorithms, namely, DANSE [21] and TI-DANSE [23]. We only consider static acoustic scenarios, where the positions of the nodes and sources are fixed throughout the simulation. This also implies that the observability of each source by each node does not change, i.e., $\mathcal{O}_{kq}$ is constant $, \forall\, (k,q) \in \mathcal{K}\times\mathcal{K}_q$.

First, a set of batch-mode simulations are performed to assess the optimality of dMWF and TI-dMWF. Scenarios with fully or partially overlapping subspaces are considered. Their advantages compared to the DANSE and TI-DANSE algorithms, respectively, already transpire through this set of simulations.

Second, online-mode simulations are conducted in a more realistic speech enhancement scenario with partially overlap-

ping subspaces. The performance of the evaluated algorithms is quantified through commonly used objective metrics.

A. Setup

We consider a WASN of K=6 nodes with M_k=10, $\forall k \in \mathcal{K}$, deployed in a 5 m × 5 m × 3 m room with a reverberation time T_{60}=0.2 s. Each acoustic scenario contains Q_{d}=5 desired sources and Q_{n}=5 noise sources. Nodes and sources positions are randomly generated on the horizontal plane at height 1.3 m, ensuring a minimum distance of 0.25 m to the walls. The sensor positions for each node are randomly generated within a horizontal disc of radius 10 cm centered at the node position. The reference sensor of each node is set to be the first sensor, w.l.o.g.. Thermal noise is simulated at each microphone with a power equal to 1% of the desired component power at the reference sensor of the corresponding node. The WASN is set to be either FC, for the dMWF and DANSE algorithms, or non-FC, for the TI-dMWF and TI-DANSE algorithms. To define the topology of non-FC WASNs once node and sources positions have been generated, a communication radius is initialized at 1.5 m and increased until the WASN is connected. The aim of each node is to estimate its own D=1-dimensional target signal. An example of the resulting scenario is provided in Fig. **??**. Figure to be made.

To control the observability patterns within the WASN, i.e., whether each source is observed by each node, we adopt the following strategy. It is first assumed that each node observes all sources, i.e., $\mathcal{O}_{kq} = \mathcal{O}_k, \forall (k,q) \in \mathcal{K} \backslash \mathcal{K}_k$. Then, for each (desired or noise) source $j \in \{1, \ldots, Q\}$, the n_j nodes closest to the source position are selected to observe it. The number n_j is uniformly drawn from the set $\{1, \ldots, K\}$. This procedure ensures that each source is observed by at least one node. An example of the resulting observability pattern is shown in Fig. **??**. Figure to be made.

The simulations are all conducted at a 16 kHz sampling rate and the signals are processed in the STFT domain with an discrete Fourier transform (DFT) length of 1024, a 50% inter-frame overlap, and a square-root Hanning window. It should be kept in mind that, due to the non-zero T_{60}, the narrowband assumption used in the signal model (1) is not perfectly satisfied.

B. Batch-mode simulations

We first evaluate the performance of dMWF and TI-dMWF in a batch-mode setting. The total signal length is set to 20 s, which corresponds to $L = 781$ frames in the STFT domain. The SCMs required to compute all necessary MWFs are estimated using time-averaging over all frames of the STFT sequence. For example, for the desired+noise centralized SCM, $\mathbf{R}_{\mathbf{yy}} = \frac{1}{L}\sum_{l=1}^{L} \mathbf{y}[l]\mathbf{y}^{\mathrm{H}}[l]$.

The latent signals generated by the desired sources are speech signals, generated using the VCTK database [41], and latent noise signal samples are drawn from the standard normal distribution. The latent noise signals are normalized in order to ensure an signal-to-noise ratio (SNR) of 0 dB at the reference sensor of node 1. When computing the results for dMWF and DANSE, the WASN is assumed to be FC. Conversely, non-FC WASNs are considered when computing the results for TI-dMWF and TI-DANSE.

For each algorithm, the SNR and short-term objective intelligibility (STOI) [42] are computed after application of each algorithm at each node, and averaged over all nodes. More specifically, the SNR is defined as:

$$\mathrm{SNR} = \frac{1}{K}\sum_{k\in\mathcal{K}} 10\log_{10}\left(\frac{\sum_{t=0}^{T-1}|\widehat{\mathbf{d}}_k[t]|^2}{\sum_{t=0}^{T-1}|\widehat{\mathbf{n}}_k[t]|^2}\right), \tag{46}$$

where T is the total number of time-domain samples and $\widehat{\mathbf{n}}_k[t]$ and $\widehat{\mathbf{d}}_k[t]$ denotes the t-th sample of the time-domain target signal estimate and noise-only component in the target signal estimate at node k, respectively. More specifically, $\widehat{\mathbf{n}}_k[t]$ is the t-th sample in the inverse STFT of $\widehat{\mathbf{n}}_k[t] = \mathrm{STFT}^{-1}\{\widetilde{\mathbf{W}}_k^{\mathrm{H}}\widetilde{\mathbf{n}}_k\}$, where $\widetilde{\mathbf{W}}_k$ is the estimation filter and $\widetilde{\mathbf{n}}_k$ the noise component in the STFT-domain observation vector $\widetilde{\mathbf{y}}_k$. The results are shown on Fig. 5 as averages over 10 randomly generated acoustic environments as defined in Section V-A. Final figure, including TI-dMWF and TI-DANSE to be made. Fig. 5 is only a placeholder including partial results.

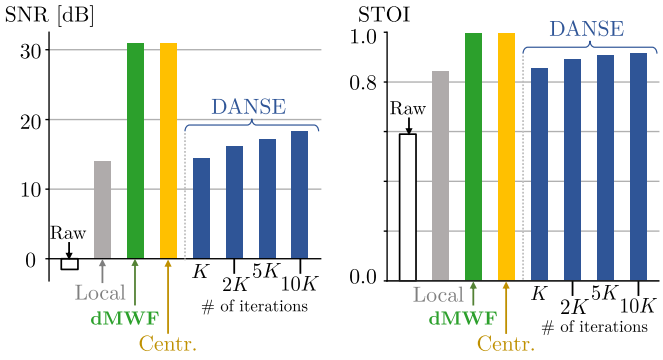

FIGURE 5. SNR and STOI for different batch-mode algorithms, as averages over all nodes in the WASN and over 10 randomly generated environments. The label "Raw" refers to the unprocessed reference microphone signals, "Local" to the MWF using only local sensor signals, and "Centr." the centralized MWF. The DANSE performance is shown for different numbers of iterations.

The results indicate...

C. Online simulations

In this section, the performance of dMWF and TI-dMWF is evaluated in an online setting where the SCMs are estimated by averaging signal time-frames. Once again, performances are compared to those of DANSE and TI-DANSE. To be written...

VI. Conclusions

We introduced dMWF and TI-dMWF, two distributed MWF-based algorithms for distributed signal estimation in a WASN. In both, nodes estimate their own node-specific

target signal based on their local sensor signals and low-dimensional (i.e., fused) signals received from other nodes. The dMWF algorithm is designed for FC WASNs, while the TI-dMWF algorithm is designed for non-FC WASNs. Both algorithms are optimal in the sense that they yield the same solution as an equivalent centralized MWF, even in scenarios with partially overlapping subspaces, i.e., where some sources are not observed by all nodes in the WASN. This provides clear advantages with respect to state-of-the-art solutions such as DANSE and TI-DANSE which do not guarantee optimality in scenarios with partially overlapping subspaces and require multiple iterations to converge. The development of dMWF and TI-dMWF is based on the observation that, in a WASN, the observability patterns of the sources by the nodes can be exploited to reduce the number of signal channels exchanged between nodes, while still allowing optimal target signal estimation. The advantages of the proposed algorithms come at the cost of a slightly higher communication bandwidth usage than DANSE and TI-DANSE, but still possibly significantly lower than the one required by a WASN where every node would transmit all their local sensor signals to every other node.

Possible avenues for future work include, but are not limited to, addressing the functioning of dMWF and TI-dMWF in dynamic acoustic environments where the observability patterns change over time, as well as the formal evaluation of the algorithms robustness to errors occurring when estimating the number of sources observed by each node.

Appendix A – dMWF optimality

We here prove Theorem 1, i.e., the optimality of the dMWF algorithm in the sense that its signal estimation performance is the same as that of an equivalent centralized system. The centralized MWF solution is first rewritten by making use of the matrix inversion lemma (MIL), as defined in (47). The local fusion problem defined by the fusion matrix formulation in (33) is then also rewritten using the MIL. Based on these new formulations, we show that the fusion matrices used to send fused signals to node k are in the same column space as the corresponding parts of the centralized filter at node $k, \forall\, k \in \mathcal{K}$, which proves that the centralized solution is reached at the end of the dMWF estimation phase. For matrices $\mathbf{A}$, $\mathbf{U}$, $\mathbf{C}$, and $\mathbf{V}$ with appropriate dimensions, the MIL is [43, pp.65]:

$$\begin{aligned}&(\mathbf{A}+\mathbf{UCV})^{-1} = \\ &\quad \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}\left(\mathbf{C}^{-1}+\mathbf{V}^{\mathrm{H}}\mathbf{A}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}\mathbf{A}^{-1}.\end{aligned} \tag{47}$$

Consider the centralized SCM $\mathbf{R}_{\mathbf{yy}}$. Based on the centralized signal model (2), it holds that:

$$\begin{aligned}\mathbf{R}_{\mathbf{yy}} &= \mathbf{R}_{\mathbf{ss}} + \mathbf{R}_{\mathbf{nn}} + \mathbf{R}_{\mathbf{vv}} && (48)\\ &= \mathbf{A}\mathbf{R}_{\mathbf{ss}}^{\mathrm{lat}}\mathbf{A}^{\mathrm{H}} + \mathbf{B}\mathbf{R}_{\mathbf{nn}}^{\mathrm{lat}}\mathbf{B}^{\mathrm{H}} + \mathbf{R}_{\mathbf{vv}} && (49)\\ &= \mathbf{C}\mathring{\mathbf{R}}^{\mathrm{lat}}\mathbf{C}^{\mathrm{H}} + \mathbf{R}_{\mathbf{vv}}, && (50)\end{aligned}$$

with $\mathbf{R}_{\mathbf{ss}}^{\mathrm{lat}} \triangleq \mathbb{E}\left\{\mathbf{s}^{\mathrm{lat}}(\mathbf{s}^{\mathrm{lat}})^{\mathrm{H}}\right\} \in \mathbb{C}^{Q_{\mathrm{d}}\times Q_{\mathrm{d}}}$, $\mathbf{R}_{\mathbf{nn}}^{\mathrm{lat}} \triangleq \mathbb{E}\left\{\mathbf{n}^{\mathrm{lat}}(\mathbf{n}^{\mathrm{lat}})^{\mathrm{H}}\right\} \in \mathbb{C}^{Q_{\mathrm{n}}\times Q_{\mathrm{n}}}$, $\mathbf{C} \triangleq [\mathbf{A}\,|\,\mathbf{B}] \in \mathbb{C}^{M\times Q}$, and $\mathring{\mathbf{R}}^{\mathrm{lat}} \triangleq \mathrm{blkdiag}\left\{\mathbf{R}_{\mathbf{ss}}^{\mathrm{lat}}, \mathbf{R}_{\mathbf{nn}}^{\mathrm{lat}}\right\} \in \mathbb{C}^{Q\times Q}$ with $\mathrm{blkdiag}\{\cdot\}$ the block-diagonalizing operator. Applying the MIL to $\mathbf{R}_{\mathbf{yy}}$ gives:

$$\mathbf{R}_{\mathbf{yy}}^{-1} = \mathbf{\Gamma} - \mathbf{\Gamma}\mathbf{C}\mathbf{X}^{-1}\mathbf{C}^{\mathrm{H}}\mathbf{\Gamma}, \tag{51}$$

where $\mathbf{\Gamma} \triangleq \mathbf{R}_{\mathbf{vv}}^{-1}$ and $\mathbf{X} \triangleq (\mathring{\mathbf{R}}^{\mathrm{lat}})^{-1} + \mathbf{C}^{\mathrm{H}}\mathbf{\Gamma}\mathbf{C} \in \mathbb{C}^{Q\times Q}$. This, in turn, allows to rewrite the centralized MWF filter from (5) at node k:

$$\begin{aligned}\widehat{\mathbf{W}}_k &= \mathbf{R}_{\mathbf{yy}}^{-1}\mathbf{R}_{\mathbf{ss}}\mathbf{E}_k && (52)\\ &= \left(\mathbf{\Gamma} - \mathbf{\Gamma}\mathbf{C}\mathbf{X}^{-1}\mathbf{C}^{\mathrm{H}}\mathbf{\Gamma}\right)\mathbf{C}\mathbf{R}_{\mathbf{ss},\mathrm{ext}}^{\mathrm{lat}}\mathbf{C}^{\mathrm{H}}\mathbf{E}_k && (53)\\ &= \mathbf{\Gamma}\mathbf{C}\mathbf{M}_k && (54)\end{aligned}$$

where $\mathbf{R}_{\mathbf{ss},\mathrm{ext}}^{\mathrm{lat}} \triangleq \mathrm{blkdiag}\left\{\mathbf{R}_{\mathbf{ss}}^{\mathrm{lat}}, \mathbf{0}_{Q_{\mathrm{n}}\times Q_{\mathrm{n}}}\right\} \in \mathbb{C}^{Q\times Q}$ and $\mathbf{M}_k \triangleq \left(\mathbf{I}_Q - \mathbf{X}^{-1}\mathbf{C}^{\mathrm{H}}\mathbf{\Gamma}\mathbf{C}\right)\mathbf{R}_{\mathbf{ss},\mathrm{ext}}^{\mathrm{lat}}\mathbf{C}^{\mathrm{H}}\mathbf{E}_k \in \mathbb{C}^{Q\times D}$.

Since self-noise in uncorrelated across all sensors, $\mathbf{\Gamma}$ is a diagonal matrix and we can define each node-specific block of $\widehat{\mathbf{W}}_k \triangleq \left[\widehat{\mathbf{W}}_{k1}^{\mathrm{T}}, \ldots, \widehat{\mathbf{W}}_{kK}^{\mathrm{T}}\right]^{\mathrm{T}}$ as:

$$\widehat{\mathbf{W}}_{kq} = \mathbf{\Gamma}_q\mathbf{C}_q\mathbf{M}_k \in \mathbb{C}^{M_q\times D}. \tag{55}$$

where the partitionings $\mathbf{C} \triangleq \left[\mathbf{C}_1^{\mathrm{T}}, \ldots, \mathbf{C}_K^{\mathrm{T}}\right]^{\mathrm{T}}$ and $\mathbf{\Gamma} \triangleq \mathrm{blkdiag}\left\{\mathbf{\Gamma}_1, \ldots, \mathbf{\Gamma}_K\right\}$ were used. We can now note that only the columns of $\mathbf{C}_q$ at indices corresponding to sources that are observed by node q (i.e., the columns at indices $\mathcal{O}_q$) are non-zero. Furthermore, since $\mathbf{R}_{\mathbf{ss},\mathrm{ext}}^{\mathrm{lat}}$ is diagonal and $\mathbf{C}^{\mathrm{H}}\mathbf{E}_k = \mathbf{C}_k^{\mathrm{H}}$, only the rows of $\mathbf{M}_k$ at indices $\mathcal{O}_k$ are non-zero. Consequently, the columns of $\mathbf{C}_q$ and rows of $\mathbf{M}_k$ at indices $\mathcal{O}_q \cup \mathcal{O}_k = \mathcal{O}_{kq}$ do not contribute to $\widehat{\mathbf{W}}_{kq}$. We let the full-column-rank matrix $\underline{\mathbf{C}}_q^k \in \mathbb{C}^{M_q\times|\mathcal{O}_{kq}|}$ denote $\mathbf{C}_q$ with only the columns at indices $\mathcal{O}_{kq}$ kept (equivalent to the definition from (28)), and $\underline{\mathbf{M}}_k \in \mathbb{C}^{|\mathcal{O}_{kq}|\times D}$ denote $\mathbf{M}_k$ with only the corresponding rows kept. We can then write:

For consistency, we could use $\underline{\mathbf{M}}_k^q$ instead of $\underline{\mathbf{M}}_k$ → too heavy?

$$\widehat{\mathbf{W}}_{kq} = \mathbf{\Gamma}_q\underline{\mathbf{C}}_q^k\underline{\mathbf{M}}_k \in \mathbb{C}^{M_q\times D}. \tag{56}$$

Consider now the dMWF local fusion problem at node q directed towards $k \in \mathcal{K}_q$. Using the MIL on $\mathbf{R}_{\mathbf{y}_q\mathbf{y}_q}$ gives:

$$\mathbf{R}_{\mathbf{y}_q\mathbf{y}_q}^{-1} = \mathbf{\Gamma}_q - \mathbf{\Gamma}_q\mathbf{C}_q\mathbf{X}_q^{-1}\mathbf{C}_q^{\mathrm{H}}\mathbf{\Gamma}_q, \tag{57}$$

where $\mathbf{\Gamma}_q \triangleq (\mathbb{E}\left\{\mathbf{v}_q\mathbf{v}_q^{\mathrm{H}}\right\})^{-1}$ and $\mathbf{X}_q \triangleq (\mathring{\mathbf{R}}^{\mathrm{lat}})^{-1} + \mathbf{C}_q^{\mathrm{H}}\mathbf{\Gamma}_q\mathbf{C}_q \in \mathbb{C}^{Q\times Q}$. It can be shown that:

$$\begin{aligned}\mathbf{C}_q\mathbf{X}_q^{-1}\mathbf{C}_q^{\mathrm{H}} &= \underline{\mathbf{C}}_q^k\underline{\mathbf{X}}_q^{-1}(\underline{\mathbf{C}}_q^k)^{\mathrm{H}} && (58)\\ \text{where } \underline{\mathbf{X}}_q &= (\mathbf{R}_{qk}^{\mathrm{lat}})^{-1} + (\underline{\mathbf{C}}_q^k)^{\mathrm{H}}\mathbf{\Gamma}_q\underline{\mathbf{C}}_q^k, && (59)\end{aligned}$$

where $\mathbf{R}_{qk}^{\text{lat}} \in \mathbb{C}^{|\mathcal{O}_{kq}|\times|\mathcal{O}_{kq}|}$ has been defined in Section III-C (cf. (28)) and is equal to $\mathring{\mathbf{R}}^{\text{lat}}$ with only the rows and columns at indices $\mathcal{O}_{kq}$ kept. Recall now the alternative expression for $\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}}$ in (28), repeated below for convenience:

$$\mathbf{R}_{\mathbf{y}_q\mathbf{y}_{k\to q}} = \underline{\mathbf{C}}_q^k \mathbf{R}_{qk}^{\text{lat}} (\underline{\mathbf{C}}_k^q)^{\text{H}} \mathbf{E}_{k\to q}. \tag{60}$$

Inserting (58) into (57), then combining the result with (60), an alternative expression for the dMWF fusion matrix from q to k can be obtained as:

$$\mathbf{P}_q^k = \left(\mathbf{\Gamma}_q - \mathbf{\Gamma}_q \underline{\mathbf{C}}_q^k \underline{\mathbf{X}}_q^{-1} (\underline{\mathbf{C}}_q^k)^{\text{H}} \mathbf{\Gamma}_q\right) \underline{\mathbf{C}}_q^k \mathbf{R}_{qk}^{\text{lat}} (\underline{\mathbf{C}}_k^q)^{\text{H}} \mathbf{E}_{k\to q} \tag{61}$$

$$= \mathbf{\Gamma}_q \underline{\mathbf{C}}_q^k \underline{\mathbf{M}}_{qk}, \tag{62}$$

where $\underline{\mathbf{M}}_{qk} \triangleq \left(\mathbf{I}_{|\mathcal{O}_{kq}|} - \underline{\mathbf{X}}_q^{-1} (\underline{\mathbf{C}}_q^k)^{\text{H}} \mathbf{\Gamma}_q \underline{\mathbf{C}}_q^k\right) \mathbf{R}_{qk}^{\text{lat}} (\underline{\mathbf{C}}_k^q)^{\text{H}} \mathbf{E}_{k\to q}$. Note that, since $\mathbf{R}_{kq}^{\text{lat}}$ and $\underline{\mathbf{C}}_k^q$ are full-rank and full-column-rank, respectively, $\underline{\mathbf{M}}_{qk}$ is also full-rank.

A clear similarity can be noticed between (62) and (56). The column space of $\mathbf{\Gamma}_q \underline{\mathbf{C}}_q^k$ is indeed either transformed by $\underline{\mathbf{M}}_k$ to lead to $\widehat{\mathbf{W}}_{kq}$, or transformed by $\underline{\mathbf{M}}_{qk}$ to lead to $\mathbf{P}_q^k$. We will now show that the estimation phase of the dMWF algorithm yields the centralized solution.

Consider the partitioning of the dMWF estimation filter $\widetilde{\mathbf{W}}_k$ from (17). The network-wide version $\mathbf{W}_k$ of the dMWF filters $\widetilde{\mathbf{W}}_k$ at node k can be written as:

$$\mathbf{W}_k \triangleq \begin{bmatrix} \mathbf{W}_{k1} \\ \vdots \\ \mathbf{W}_{kk} \\ \vdots \\ \mathbf{W}_{kK} \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{P}_1^k \mathbf{G}_{k1} \\ \vdots \\ \mathbf{W}_{kk} \\ \vdots \\ \mathbf{P}_K^k \mathbf{G}_{kK} \end{bmatrix} = \begin{bmatrix} \mathbf{\Gamma}_1 \underline{\mathbf{C}}_1^k \underline{\mathbf{M}}_{1k} \mathbf{G}_{k1} \\ \vdots \\ \mathbf{W}_{kk} \\ \vdots \\ \mathbf{\Gamma}_K \underline{\mathbf{C}}_K^k \underline{\mathbf{M}}_{Kk} \mathbf{G}_{kK} \end{bmatrix}, \tag{63}$$

where it can be verified that $\mathbf{W}_k^{\text{H}}\mathbf{y} = \widetilde{\mathbf{W}}_k^{\text{H}}\widetilde{\mathbf{y}}_k$. Computing the solution to (15), i.e., optimizing $\mathbf{W}_{kk}$ and $\{\mathbf{G}_{kq}\}_{q\in\mathcal{K}_k}$ to estimate $\mathbf{d}_k$, is thus sufficient to obtain:

$$\mathbf{G}_{kq} = \underline{\mathbf{M}}_{qk}^{-1} \underline{\mathbf{M}}_k, \forall\, q \in \mathcal{K}_k, \tag{64}$$

$$\mathbf{W}_{kk} = \widehat{\mathbf{W}}_{kk}, \tag{65}$$

which implies $\mathbf{W}_{kq} = \widehat{\mathbf{W}}_{kq}, \forall\, q \in \mathcal{K}$ and thus $\mathbf{W}_k = \widehat{\mathbf{W}}_k, \forall\, k \in \mathcal{K}$. ■

Appendix B – TI-dMWF optimality

We here prove Theorem 2, i.e., the optimality of the TI-dMWF algorithm in the sense that its signal estimation performance is the same as that of an equivalent centralized system. Consider the following partitioning of the TI-dMWF estimation filter $\widetilde{\mathbf{W}}_k^{\text{TI}}$ from (43) as:

$$\widetilde{\mathbf{W}}_k^{\text{TI}} = \left[(\mathbf{W}_{kk}^{\text{TI}})^{\text{T}}\ (\mathbf{G}_{k1}^{\text{TI}})^{\text{T}}\ \dots\ (\mathbf{G}_{kB}^{\text{TI}})^{\text{T}}\right]^{\text{T}}, \tag{66}$$

where $\mathbf{W}_{kk}^{\text{TI}} \in \mathbb{C}^{M_k\times D}$ is to be applied to $\mathbf{y}_k$ and $\mathbf{G}_{kq}^{\text{TI}} \in \mathbb{C}^{\widehat{Q}_{kq}\times D}$ is to be applied to $\widehat{\mathbf{z}}_q^k, \forall\, q \in \mathcal{U}_k$.

We now aim at defining the network-wide version $\mathbf{W}_k^{\text{TI}}$ of the TI-dMWF filter $\widetilde{\mathbf{W}}_k^{\text{TI}}$ at node k, such that $(\mathbf{W}_k^{\text{TI}})^{\text{H}}\mathbf{y} = (\widetilde{\mathbf{W}}_k^{\text{TI}})^{\text{H}}\widehat{\mathbf{y}}_k$. Consider the following partitioning of the TI-dMWF fusion matrix $\widehat{\mathbf{P}}_q^k$:

$$\widehat{\mathbf{P}}_q^k = \left[(\widehat{\mathbf{P}}_{qq}^k)^{\text{T}}\ (\widehat{\mathbf{P}}_{qu_1}^k)^{\text{T}}\ \dots\ (\widehat{\mathbf{P}}_{qu_B}^k)^{\text{T}}\right]^{\text{T}}. \tag{67}$$

where $\widehat{\mathbf{P}}_{qq}^k \in \mathbb{C}^{M_q\times\widehat{Q}_{kq}}$ is applied to $\mathbf{y}_q$ and $\widehat{\mathbf{P}}_{qu}^k \in \mathbb{C}^{\widehat{Q}_{ku}\times\widehat{Q}_{kq}}$ to $\widehat{\mathbf{z}}_u^q, \forall\, u \in \mathcal{U}_q = \{u_1, \dots, u_B\}$, as in (35).

Let $\{q \to d_1 \to d_2 \to \dots d_N \to q'\}$ denote the downstream path from node q to node q', where $q \in \overline{\mathcal{U}}_{q'}$. We define the following sequence of matrix multiplication, $\forall\, q \in \mathcal{K}_k$, based on the partitioning defined in (67):

$$\widehat{\mathbf{P}}_{q\to q'}^k \triangleq \widehat{\mathbf{P}}_{qq}^k \widehat{\mathbf{P}}_{d_1q}^k \widehat{\mathbf{P}}_{d_2d_1}^k \dots \widehat{\mathbf{P}}_{q'd_N}^k \in \mathbb{C}^{M_q\times\widehat{Q}_{kq'}}, \tag{68}$$

This allows us to express the TI-dMWF network-wide filter at node k as:

$$\mathbf{W}_k^{\text{TI}} \triangleq \begin{bmatrix} \mathbf{W}_{k1}^{\text{TI}} \\ \vdots \\ \mathbf{W}_{kk}^{\text{TI}} \\ \vdots \\ \mathbf{W}_{kK}^{\text{TI}} \end{bmatrix} \triangleq \begin{bmatrix} \widehat{\mathbf{P}}_{1\to d(1)}^k \mathbf{G}_{kd(1)}^{\text{TI}} \\ \vdots \\ \mathbf{W}_{kk}^{\text{TI}} \\ \vdots \\ \widehat{\mathbf{P}}_{K\to d(K)}^k \mathbf{G}_{kd(K)}^{\text{TI}} \end{bmatrix}, \tag{69}$$

where $d(q)$ denotes the unique node in $\mathcal{U}_k$ downstream of $q, \forall\, q \in \mathcal{K}_k$ (note that, $q \in \mathcal{U}_k \Leftrightarrow d(q) = q$) and it can indeed be verified that $(\mathbf{W}_k^{\text{TI}})^{\text{H}}\mathbf{y} = (\widetilde{\mathbf{W}}_k^{\text{TI}})^{\text{H}}\widehat{\mathbf{y}}_k$. We now rewrite $\widehat{\mathbf{P}}_{q\to k}$, defined in (68), as a function of the steering matrices in $\mathbf{C}, \forall\, q \in \mathcal{K}_k$. First, $\widehat{\mathbf{y}}_q$ can be expressed as a function of the centralized signal vector $\mathbf{y}$:

$$\widehat{\mathbf{y}}_q = \left[\mathbf{y}_q^{\text{T}}, (\widehat{\mathbf{z}}_{u_1}^q)^{\text{T}}, \dots, (\widehat{\mathbf{z}}_{u_B}^q)^{\text{T}}\right]^{\text{T}} \tag{70}$$

$$= \mathbf{D}_q^{\text{H}}\mathbf{y} \tag{71}$$

$$\text{where } \mathbf{D}_q \triangleq [\mathbf{S}_q \mid \mathbf{V}_{qu_1} \mid \dots \mid \mathbf{V}_{qu_B}] \in \mathbb{C}^{M\times\widehat{M}_q}, \tag{72}$$

$$\text{with } \mathbf{S}_q \triangleq \left[\mathbf{0}_{M_q\times\sum_{q'<q}M_{q'}} \mid \mathbf{I}_{M_q} \mid \mathbf{0}_{M_q\times\sum_{q'>q}M_{q'}}\right]^{\text{T}}, \tag{73}$$

$$\text{and } \mathbf{V}_{qu} \triangleq \left[\mathbf{V}_{qu,1}^{\text{T}}, \dots, \mathbf{V}_{qu,K}^{\text{T}}\right]^{\text{T}}, \forall\, u \in \mathcal{U}_q, \tag{74}$$

$$\text{with } \mathbf{V}_{qu,q'} \triangleq \begin{cases} \widehat{\mathbf{P}}_{q'\to u}^k & \text{if } q' \in \overline{\mathcal{U}}_u \cup \{u\} \\ \mathbf{0}_{M_{q'}\times\widehat{Q}_{ku}} & \text{otherwise} \end{cases}. \tag{75}$$

Possibility for conciseness to omit (72) to (75) (?)

We can rewrite $\mathbf{R}_{\widehat{\mathbf{y}}_q\widehat{\mathbf{y}}_q}$ using (71) and (50) as:

$$\mathbf{R}_{\widehat{\mathbf{y}}_q\widehat{\mathbf{y}}_q} = \mathbb{E}\left\{\widehat{\mathbf{y}}_q\widehat{\mathbf{y}}_q^{\text{H}}\right\} \tag{76}$$

$$= \mathbf{D}_q^{\text{H}}\mathbf{R}_{\mathbf{yy}}\mathbf{D}_q \tag{77}$$

$$= \mathbf{D}_q^{\text{H}}\mathbf{C}\mathring{\mathbf{R}}^{\text{lat}}\mathbf{C}^{\text{H}}\mathbf{D}_q + \mathbf{D}_q^{\text{H}}\mathbf{R}_{\mathbf{vv}}\mathbf{D}_q. \tag{78}$$

The MIL can then be applied to (78) to obtain:

$$\mathbf{R}_{\widehat{\mathbf{y}}_q\widehat{\mathbf{y}}_q}^{-1} = \mathbf{\Gamma}' - \mathbf{\Gamma}'\mathbf{D}_q^{\mathrm{H}}\mathbf{C}(\mathbf{X}')^{-1}\mathbf{C}^{\mathrm{H}}\mathbf{D}_q\mathbf{\Gamma}', \tag{79}$$

$$\text{where } \mathbf{\Gamma}' \triangleq \left(\mathbf{D}_q^{\mathrm{H}}\mathbf{R}_{\mathbf{vv}}\mathbf{D}_q\right)^{-1} \in \mathbb{C}^{\widehat{M}_q\times\widehat{M}_q}, \tag{80}$$

$$\text{and } \mathbf{X}' \triangleq (\mathring{\mathbf{R}}^{\mathrm{lat}})^{-1} + \mathbf{C}^{\mathrm{H}}\mathbf{D}_q\mathbf{\Gamma}'\mathbf{D}_q^{\mathrm{H}}\mathbf{C} \in \mathbb{C}^{Q\times Q}. \tag{81}$$

It can be verified that, due to the respective structures of $\mathbf{D}_q$ and $\mathbf{R}_{\mathbf{vv}}$, it holds that:

$$\mathbf{\Gamma}' = \mathrm{blkdiag}\left\{\mathbf{\Gamma}_q, \mathbf{\Gamma}'_{qu_1}, \dots, \mathbf{\Gamma}'_{qu_1}\right\}, \tag{82}$$

where $\mathbf{\Gamma}_q \in \mathbb{C}^{M_q\times M_q}$ still denotes the diagonal matrix $(\mathbb{E}\left\{\mathbf{v}_q\mathbf{v}_q^{\mathrm{H}}\right\})^{-1}$ and $\mathbf{\Gamma}'_{qu} \in \mathbb{C}^{\widehat{Q}_{ku}\times\widehat{Q}_{ku}}, \forall\, u \in \mathcal{U}_q$. The SCM $\mathbf{R}_{\widehat{\mathbf{y}}_q\mathbf{y}_{k\uparrow q}}$ from (41) can now be rewritten as:

$$\mathbf{R}_{\widehat{\mathbf{y}}_q\mathbf{y}_{k\uparrow q}} = \mathbb{E}\left\{\widehat{\mathbf{y}}_q\mathbf{y}_k^{\mathrm{H}}\right\}\mathbf{E}_{\uparrow q} = \mathbf{D}_q^{\mathrm{H}}\mathbb{E}\left\{\mathbf{y}\mathbf{y}_k^{\mathrm{H}}\right\}\mathbf{E}_{\uparrow q} \tag{83}$$

$$= \mathbf{D}_q^{\mathrm{H}}\mathbb{E}\left\{\left(\mathbf{C}\begin{bmatrix}\mathbf{s}^{\mathrm{lat}}\\ \mathbf{n}^{\mathrm{lat}}\end{bmatrix} + \mathbf{v}\right)\left(\mathbf{C}_k\begin{bmatrix}\mathbf{s}^{\mathrm{lat}}\\ \mathbf{n}^{\mathrm{lat}}\end{bmatrix} + \mathbf{v}_k\right)^{\mathrm{H}}\right\}\mathbf{E}_{\uparrow q} \tag{84}$$

$$= \mathbf{D}_q^{\mathrm{H}}\left(\mathbf{C}\mathring{\mathbf{R}}^{\mathrm{lat}}\mathbf{C}_k^{\mathrm{H}} + \mathbb{E}\left\{\mathbf{v}\mathbf{v}_k^{\mathrm{H}}\right\}\right)\mathbf{E}_{\uparrow q} \tag{85}$$

Note that $\mathbf{v}_k$ does not appear at all in $\mathbf{D}_q^{\mathrm{H}}$ since k is the root node (which is never involved in $\mathbf{D}_q$ according to (75)). Therefore, the second term of the sum in (85) vanishes and we obtain:

$$\mathbf{R}_{\widehat{\mathbf{y}}_q\mathbf{y}_{k\uparrow q}} = \mathbf{D}_q^{\mathrm{H}}\mathbf{C}\mathring{\mathbf{R}}^{\mathrm{lat}}\mathbf{C}_k^{\mathrm{H}}\mathbf{E}_{\uparrow q}. \tag{86}$$

Combining (86) with (79) and inserting into (42), we obtain:

$$\widehat{\mathbf{P}}_q^k = \mathbf{R}_{\widehat{\mathbf{y}}_q\widehat{\mathbf{y}}_q}^{-1}\mathbf{R}_{\widehat{\mathbf{y}}_q\mathbf{y}_{k\uparrow q}} \tag{87}$$

$$= \left(\mathbf{\Gamma}' - \mathbf{\Gamma}'\mathbf{F}_q(\mathbf{X}')^{-1}\mathbf{F}_q^{\mathrm{H}}\mathbf{\Gamma}'\right)\mathbf{F}_q\mathring{\mathbf{R}}^{\mathrm{lat}}\mathbf{C}_k^{\mathrm{H}}\mathbf{E}_{\uparrow q} \tag{88}$$

$$= \mathbf{\Gamma}'\mathbf{F}_q\mathbf{M}_q^k, \tag{89}$$

where we have defined, for notation conciseness:

$$\mathbf{F}_q \triangleq \mathbf{D}_q^{\mathrm{H}}\mathbf{C} = \begin{bmatrix}\mathbf{S}_q^{\mathrm{H}}\mathbf{C}\\ \mathbf{V}_{qu_1}^{\mathrm{H}}\mathbf{C}\\ \vdots\\ \mathbf{V}_{qu_B}^{\mathrm{H}}\mathbf{C}\end{bmatrix} \in \mathbb{C}^{\widehat{M}_q\times Q}, \tag{90}$$

and:

$$\mathbf{M}_q^k \triangleq \left(\mathbf{I}_Q - (\mathbf{X}')^{-1}\mathbf{F}_q^{\mathrm{H}}\mathbf{\Gamma}'\mathbf{F}_q\right)\mathring{\mathbf{R}}^{\mathrm{lat}}\mathbf{C}_k^{\mathrm{H}}\mathbf{E}_{\uparrow q} \in \mathbb{C}^{Q\times\widehat{Q}_{kq}}. \tag{91}$$

Making use of (82) and (90) in the $\widehat{\mathbf{P}}_q^k$ partitioning defined in (67), we obtain:

$$\widehat{\mathbf{P}}_{qq}^k = \mathbf{\Gamma}_q\mathbf{S}_q^{\mathrm{H}}\mathbf{C}\mathbf{M}_q^k = \mathbf{\Gamma}_q\mathbf{C}_q\mathbf{M}_q^k, \tag{92}$$

$$\widehat{\mathbf{P}}_{qu}^k = \mathbf{\Gamma}'_{qu}\mathbf{V}_{qu}^{\mathrm{H}}\mathbf{C}\mathbf{M}_q^k, \forall\, u \in \mathcal{U}_q, \tag{93}$$

which, inserting in (68), gives:

$$\widehat{\mathbf{P}}_{q\to q'}^k = \mathbf{\Gamma}_q\mathbf{C}_q\mathbf{M}_{q\to q'}^k, \tag{94}$$

where $\mathbf{M}_{q\to q'}^k \triangleq \mathbf{M}_q^k\widehat{\mathbf{P}}_{d_1q}^k\widehat{\mathbf{P}}_{d_2d_1}^k\dots\widehat{\mathbf{P}}_{q'd_N}^q \in \mathbb{C}^{Q\times\widehat{Q}_{kq'}}$. Following the same logic as in the Theorem 1 proof and (22), we can discard the all-zero columns of $\mathbf{C}_q$ and the corresponding rows of $\mathbf{M}_{q\to q'}^k$ to obtain:

$$\widehat{\mathbf{P}}_{q\to q'}^k = \mathbf{\Gamma}_q\underline{\mathbf{C}}_q^k\underline{\mathbf{M}}_{q\to q'}^k, \tag{95}$$

where $\underline{\mathbf{M}}_{q\to q'}^k \in \mathbb{C}^{|\mathcal{O}_{kq}|\times\widehat{Q}_{kq}}$ denotes $\mathbf{M}_{q\to q'}^k$ with only the appropriate rows kept. A clear similarity can be noticed between (95) and (56). The column space of $\mathbf{\Gamma}_q\underline{\mathbf{C}}_q^k$ is indeed either transformed by $\underline{\mathbf{M}}_k$ to lead to $\widehat{\mathbf{W}}_{kq}$, or transformed by $\underline{\mathbf{M}}_{q\to q'}^k$ to lead to $\widehat{\mathbf{P}}_{q\to q'}^k$. Given the form of the TI-dMWF network-wide filters in (69), computing the solution to (43) (i.e., optimizing $\mathbf{W}_{kk}^{\mathrm{TI}}$ and $\{\mathbf{G}_{kq}^{\mathrm{TI}}\}_{q\in\mathcal{U}_k}$ to estimate $\mathbf{d}_k$) is sufficient to obtain:

$$\mathbf{G}_{kq}^{\mathrm{TI}} = \left(\underline{\mathbf{M}}_{q\to q'}^k\right)^{-1}\underline{\mathbf{M}}_k, \forall\, q \in \mathcal{U}_k, \tag{96}$$

$$\mathbf{W}_{kk}^{\mathrm{TI}} = \widehat{\mathbf{W}}_{kk}, \tag{97}$$

which implies $\mathbf{W}_{kq}^{\mathrm{TI}} = \widehat{\mathbf{W}}_{kq}, \forall\, q \in \mathcal{K}$ and thus $\mathbf{W}_k^{\mathrm{TI}} = \widehat{\mathbf{W}}_k, \forall\, k \in \mathcal{K}$. ■

REFERENCES

[1] M. Brandstein and D. Ward, *Microphone Arrays: Signal Processing Techniques and Applications*, Springer Science & Business Media, May 2001.

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

[3] B. Van Veen and K. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Mag.*, vol. 5, no. 2, pp. 4–24, Apr. 1988.

[4] P. Didier, T. Van Waterschoot, S. Doclo, and M. Moonen, "Sampling Rate Offset Estimation and Compensation for Distributed Adaptive Node-Specific Signal Estimation in Wireless Acoustic Sensor Networks," *IEEE Open J. Sig. Proc.*, vol. 4, pp. 71–79, 2023.

[5] T. Gburrek, J. Schmalenstroeer, and R. Haeb-Umbach, "On synchronization of wireless acoustic sensor networks in the presence of time-varying sampling rate offsets and speaker changes," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2022, pp. 916–920.

[6] J. Schmalenstroeer, J. Heymann, L. Drude, C. Boeddecker, and R. Haeb-Umbach, "Multi-stage coherence drift based sampling rate synchronization for acoustic beamforming," in *Proc. Int. Workshop Multimedia Signal Process.*, 2017, pp. 1–6.

[7] R. Lienhart, I. Kozintsev, S. Wehr, and M. Yeung, "On the importance of exact synchronization for distributed audio signal processing," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2003, pp. 840–843.

[8] J. Zhang, S. P. Chepuri, R. C. Hendriks, and R. Heusdens, "Microphone Subset Selection for MVDR Beamformer Based Noise Reduction," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 3, pp. 550–563, Mar. 2018.

[9] A. Bertrand, J. Szurley, P. Ruckebusch, I. Moerman, and M. Moonen, "Efficient Calculation of Sensor Utility and Sensor Removal in Wireless Sensor Networks for Adaptive Signal Estimation and Beamforming," *IEEE Trans. Signal Process.*, vol. 60, no. 11, pp. 5857–5869, Nov. 2012.

[10] S. Liu, S. P. Chepuri, M. Fardad, E. Masazade, G. Leus, and P. K. Varshney, "Sensor Selection for Estimation with Correlated Measurement Noise," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3509–3522, July 2016.

[11] B. Cornelis, M. Moonen, and J. Wouters, "Performance Analysis of Multichannel Wiener Filter-Based Noise Reduction in Hearing Aids Under Second Order Statistics Estimation Errors," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 19, no. 5, pp. 1368–1381, July 2011.

[12] C. A. Musluoglu, *Distributed Spatial Filtering in Wireless Sensor Networks*, Ph.D. thesis, KU Leuven, 2024.

[13] S. Markovich-Golan, A. Bertrand, M. Moonen, and S. Gannot, "Optimal distributed minimum-variance beamforming approaches for speech enhancement in wireless acoustic sensor networks," *Signal Processing*, vol. 107, pp. 4–20, Feb. 2015.

[14] A. Bertrand and M. Moonen, "Distributed Node-Specific LCMV Beamforming in Wireless Sensor Networks," *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 233–246, Jan. 2012.

[15] D. Hu, Q. Si, and W. Zhang, "Distributed-Robust MVDR Beamforming with Energy-Efficient Topology Control in Wireless Acoustic Sensor Networks," *IEEE Wireless Commun.*, pp. 1–1, 2024.

[16] M. O'Connor, W. B. Kleijn, and T. Abhayapala, "Distributed sparse MVDR beamforming using the bi-alternating direction method of multipliers," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2016, pp. 106–110.

[17] A. I. Koutrouvelis, T. W. Sherson, R. Heusdens, and R. C. Hendriks, "A Low-Cost Robust Distributed Linearly Constrained Beamformer for Wireless Acoustic Sensor Networks With Arbitrary Topology," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 8, pp. 1434–1448, Aug. 2018.

[18] I. D. Schizas, G. B. Giannakis, and Z.-Q. Luo, "Distributed Estimation Using Reduced-Dimensionality Sensor Observations," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4284–4299, Aug. 2007.

[19] S. Doclo, M. Moonen, T. Van den Bogaert, and J. Wouters, "Reduced-Bandwidth and Distributed MWF-Based Noise Reduction Algorithms for Binaural Hearing Aids," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 17, no. 1, pp. 38–51, Jan. 2009.

[20] A. Bertrand and M. Moonen, "Robust Distributed Noise Reduction in Hearing Aids with External Acoustic Sensor Nodes," *EURASIP J. Adv. Signal Process.*, vol. 2009, no. 1, pp. 530435, Dec. 2009.

[21] 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, 2010.

[22] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks—Part II: Simultaneous and asynchronous node updating," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5292–5306, 2010.

[23] 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, 2017.

[24] A. Bertrand and M. Moonen, "Distributed Adaptive Estimation of Node-Specific Signals in Wireless Sensor Networks With a Tree Topology," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2196–2210, May 2011.

[25] Y. Hioka and W. B. Kleijn, "Distributed blind source separation with an application to audio signals," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2011, pp. 233–236.

[26] Y. Jia, Y. Luo, Y. Lin, and I. Kozintsev, "Distributed Microphone Arrays for Digital Home and Office," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2006, vol. 5, pp. V–V.

[27] M. Kowalski, E. Vincent, and R. Gribonval, "Beyond the Narrowband Approximation: Wideband Convex Methods for Under-Determined Reverberant Audio Source Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 18, no. 7, pp. 1818–1829, Sept. 2010.

[28] J. Plata-Chaves, A. Bertrand, and M. Moonen, "Distributed signal estimation in a wireless sensor network with partially-overlapping node-specific interests or source observability," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, South Brisbane, Queensland, Australia, Apr. 2015, pp. 5808–5812, IEEE.

[29] M. H. Bahari, L. K. Hamaidi, M. Muma, J. Plata-Chaves, M. Moonen, A. M. Zoubir, and A. Bertrand, "Distributed Multi-Speaker Voice Activity Detection for Wireless Acoustic Sensor Networks," Mar. 2017.

[30] A. Bertrand and M. Moonen, "Energy-based multi-speaker voice activity detection with an ad hoc microphone array," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Dallas, TX, USA, 2010, pp. 85–88, IEEE.

[31] J. Freudenberger and S. Stenzel, "Time-frequency dependent voice activity detection based on a simple threshold test," in *2011 IEEE Statistical Signal Processing Workshop (SSP)*, June 2011, pp. 541–544.

[32] M. Souden, Jingdong Chen, J. Benesty, and S. Affes, "Gaussian Model-Based Multichannel Speech Presence Probability," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 18, no. 5, pp. 1072–1077, July 2010.

[33] M. Tammen, D. Fischer, B. T. Meyer, and S. Doclo, "DNN-Based Speech Presence Probability Estimation for Multi-Frame Single-Microphone Speech Enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Barcelona, Spain, May 2020, pp. 191–195, IEEE.

[34] 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," *The Journal of the Acoustical Society of America*, vol. 147, no. 6, pp. 4189–4201, June 2020.

[35] D. Williams and D. Johnson, "Using the sphericity test for source detection with narrow-band passive arrays," vol. 38, no. 11, pp. 2008–2014, Nov. 1990.

[36] R. Brcich, A. Zoubir, and P. Pelin, "Detection of sources using bootstrap techniques," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 206–215, Feb. 2002.

[37] P. Stoica and Y. Selen, "Model-order selection: A review of information criterion rules," *IEEE Signal Process. Mag.*, vol. 21, no. 4, pp. 36–47, July 2004.

[38] M. Wax and A. Adler, "Detection of the Number of Signals by Signal Subspace Matching," *IEEE Trans. Signal Process.*, vol. 69, pp. 973–985, 2021.

[39] 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, 2016.

[40] R. C. Prim, "Shortest connection networks and some generalizations," *Bell Syst. Tech. J.*, vol. 36, no. 6, pp. 1389–1401, 1957.

[41] C. Veaux, J. Yamagishi, and K. MacDonald, "CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit," Tech. Rep., The Centre for Speech Technology Research (CSTR), University of Edinburgh, 2017, Accessed: 2025-05-27.

[42] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An Algorithm for Intelligibility Prediction of Time–Frequency Weighted Noisy Speech," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 2125–2136, Sept. 2011.

[43] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins Studies in the Mathematical Sciences. The Johns Hopkins University Press, Baltimore, fourth edition, 2013.