Distributed Combined Acoustic Echo Cancellation and Noise Reduction in Wireless Acoustic Sensor and Actuator Networks

Santiago Ruiz, Toon van Waterschoot and Marc Moonen

***Abstract*—The paper presents distributed algorithms for combined acoustic echo cancellation (AEC) and noise reduction (NR) in a wireless acoustic sensor and actuator network (WASAN) where each node may have multiple microphones and multiple loudspeakers, and where the desired signal is a speech signal. A centralized integrated AEC and NR algorithm, i.e., multichannel Wiener filter (MWF), is used as starting point where echo signals are viewed as background noise signals and loudspeaker signals are used as additional input signals to the algorithm. By including prior knowledge (PK), namely that the loudspeaker signals do not contain any desired signal component, an alternative centralized cascade algorithm (PK-MWF) is obtained with an AEC stage first followed by an MWF-based NR stage which has a lower computational complexity. Distributed algorithms can then be obtained from the MWF and PK-MWF algorithm, i.e., the generalized eigenvalue decomposition (GEVD)-based distributed adaptive node-specific signal estimation (DANSE) and PK-GEVD-DANSE algorithm, respectively. In the former, each node performs a reduced dimensional integrated AEC and NR algorithm and broadcasts only 1 fused signal (instead of all its signals) to the other nodes. In the PK-GEVD-DANSE algorithm, each node performs a reduced dimensional cascade AEC and NR algorithm and broadcasts only 2 fused signals (instead of all its signals) to the other nodes. The distributed algorithms achieve the same performance, upon convergence, as the corresponding centralized integrated (MWF) and centralized cascade (PK-MWF) algorithm. It is observed, however, that the communication cost in the PK-GEVD-DANSE algorithm can also be reduced, where each node then broadcasts only 1 fused signal (instead of 2 signals) to the other nodes. The resulting algorithm, referred to as the pruned PK-GEVD-DANSE (pPK-GEVD-DANSE) algorithm, then effectively combines the lowest possible communication cost (as low as in the GEVD-DANSE algorithm) with a lowest possible computational complexity in each node (further reduced from the PK-GEVD-DANSE computational complexity), within the class of algorithms considered in this paper.**

***Index Terms*—Distributed signal processing, wireless acoustic sensor and actuator networks, acoustic echo cancellation, noise reduction, multichannel Wiener filter**

This research work was carried out at the ESAT Laboratory of KU Leuven, in the frame of Research Council KU Leuven Project C3-19-00221 "Cooperative Signal Processing Solutions for IoT-based Multi-User Speech Communication Systems", VLAIO O&O Project nr. HBC.2020.2197 'SPIC: Signal Processing and Integrated circuits for Communications', Fonds de la Recherche Scientifique - FNRS and the Fonds Wetenschappelijk Onderzoek - Vlaanderen under EOS Project no 30452698 '(MUSE-WINET) MUlti-SErvice WIreless NETwork' and the European Research Council under the European Union's Horizon 2020 research and innovation program / ERC Consolidator Grant: SONORA (no. 773268). 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 its authors.

I. Introduction

MANY speech and audio signal processing applications, such as teleconferencing/telepresence, in-car communication and ambient intelligence, suffer from acoustic echoes and background noise which corrupt the desired audio signal. Acoustic echo cancellation (AEC) and noise reduction (NR) techniques can be used to enhance the desired signal while reducing undesired signal components [1]–[4].

AEC is a well-studied problem for which a first adaptive solution was proposed in [5], be it still conceived as an analog device. Advances in digital signal processing (DSP) chips and the implementation of adaptive algorithms such as the Least Mean Squares (LMS) algorithm set in motion different approaches to solve the AEC problem [6]. AEC solutions include, but are not limited to, LMS algorithms, subband processing, Kalman filters and deep learning architectures (cfr. [7]–[10] and references therein). Remaining challenges include, but are not limited to, the adaptation of the filter coefficients during double-talk situations, multiple-input multiple-output scenarios, nonlinear echo cancellation and real-world datasets for deep learning methods [10], [11].

Solutions to combined AEC and NR have been presented in the literature, which fundamentally can be divided into cascade and integrated approaches [3], [4], [12]–[15]. A cascade approach consists of an AEC stage and an NR stage which can be combined in two ways, i.e., a multichannel AEC stage followed by a multichannel NR stage, or a single-channel AEC stage preceded by a multichannel NR stage [8], [16]. The order of the stages has performance implications on the combined system. The first combination requires an AEC that is robust against noise in the microphone signals. In the second combination the AEC stage receives a noise-reduced signal which may contain a far-end signal component, therefore the AEC stage should be able to track changes in the acoustic environment as well as in the NR filters. Integrated approaches aim to solve the problem by combining the AEC and NR tasks in a single optimization process [9], [12], [14].

Recently, a multichannel Kalman-based Wiener filter for speaker interference reduction was proposed in [17]. The filter is based on a multichannel AEC stage followed by a NR stage using a multichannel Wiener filter. The proposed method was developed and implemented for a specific set-up with three speakers. Combined AEC and NR was implemented using a Kalman filter for a single-channel scenario in [18]. The use of deep neural networks to solve combined AEC and NR has also

gained significant attention [11], [19], [20]. Although these methods usually outperform model-based methods, their main drawback is their dependency on training sets, which limits their practical deployment in mobile devices [11].

Existing solutions are all based on centralized processing, which is usually prohibitive in a wireless acoustic sensor and actuator network (WASAN) in terms of complexity and communication cost [21]. Distributed algorithms have been developed to overcome this, such as, e.g., the distributed delay-and-sum beamformer for NR based on randomized gossiping presented in [22], which was extended to a distributed MVDR beamformer based on message passing in [23]. Both algorithms do not have a topology constraint and provide good performance at the expense of a high communication cost [22]. The distributed adaptive node-specific signal estimation (DANSE) algorithm as developed in [24], performs distributed NR, i.e., optimally enhances the desired signal component in the local microphone signals of each node. It achieves a performance as if all microphone signals in the network were available to each and every node, while each node is still sharing only a fused version of its microphone signals with the other nodes. A combination of a neural network and beamforming was used in [25] for a real-time multi-channel speech enhancement algorithm, where a spectral mask estimation is performed via the deep neural network together with spatial filtering. All these distributed algorithms only consider NR.

In this paper, distributed algorithms for combined AEC and NR are presented, where each node may have multiple microphones and multiple loudspeakers, and where the desired signal is a speech signal. In a WASAN with K nodes, node $k \in \mathcal{K} = \{1, \ldots, K\}$ contains m_k microphones and l_k loudspeakers. The loudspeakers play given (far-end) signals, and generate echo signals in the microphones (also in other nodes). Node k then has access to an $n_k = m_k + P.l_k$ vector signal, where $P-1$ will be defined as the order of the interframe filtering in the AEC stage in Section II. The total number of microphones and loudspeakers in the WASAN are denoted, respectively, by $M = \sum_{k=1}^{K} m_k$ and $L = \sum_{k=1}^{K} l_k$, and similarly, $N = \sum_{k=1}^{K} n_k$. Centralized, non-cooperative and distributed algorithms can be used for combined AEC and NR, where the following should be considered: A centralized cascade algorithm has an AEC stage with PL AEC filter input signals, and a NR stage with M channels. A non-cooperative cascade algorithm for node k (i.e. node k working in isolation) has an AEC stage with Pl_k AEC filter input signals, and a NR stage with m_k channels. A distributed algorithm aims to reduce computational complexity by performing local operations in each node and exchanging data with other nodes.

In [26] distributed combined AEC and NR was considered in a WASAN. Essentially, a centralized integrated algorithm, i.e., the multichannel Wiener filter (MWF), is first turned into an alternative centralized cascade algorithm by introducing prior knowledge (PK). In the MWF algorithm no distinction is made between loudspeaker and microphone signals, which means echo signals are viewed as additional background noise signals and loudspeaker signals are used as additional input signals to the algorithm. By including PK, namely that the loudspeaker signals do not contain any desired signal component, the MWF algorithm is turned into the PK-MWF algorithm, leading to the alternative centralized cascade algorithm, with an AEC stage first followed by an MWF-based NR stage. The resulting algorithm has a lower computational complexity and allows to substitute alternative algorithms in the AEC stage.

Both the MWF and PK-MWF algorithm can be turned into a distributed algorithm, namely the generalized eigenvalue decomposition (GEVD)-based DANSE (GEVD-DANSE) [24] and the PK-GEVD-DANSE [27]. In the GEVD-DANSE algorithm, each node in the network performs a reduced dimensional (dimension $n_k + K - 1$) integrated AEC and NR algorithm and broadcasts only 1 fused signal (instead of n_k signals) to the other nodes, and yet each node achieves the same performance, upon convergence, as the centralized integrated algorithm, i.e., as if all loudspeaker and microphone signals were broadcast in the network. In the PK-GEVD-DANSE algorithm, each node in the network performs a reduced dimensional (dimension $n_k + 2(K-1)$) cascade AEC and NR algorithm and broadcasts only 2 fused signals (instead of n_k signals) to the other nodes, and yet each node again achieves the same performance, upon convergence, as the centralized cascade algorithm.

The PK-GEVD-DANSE algorithm in [26] performs AEC and NR in each node based on sharing not only fused microphone and loudspeaker signals between the nodes, which act as desired signal references, but also fused loudspeaker signals, which act as noise references. In this paper, however, it will be shown that in an AEC context (unlike in the general PK-GEVD-DANSE context) there is no need for sharing noise references between the nodes, reducing the communication cost in the PK-GEVD-DANSE algorithm. Each node then effectively performs a reduced dimensional (dimension $n_k + K - 1$) cascade AEC and NR algorithm and broadcasts only 1 fused signal (instead of 2 signals) to the other nodes. The resulting algorithm, referred to as the pruned PK-GEVD-DANSE (pPK-GEVD-DANSE) algorithm, then effectively combines the lowest possible communication cost (as low as in the GEVD-DANSE algorithm) with a lowest possible computational complexity in each node (further reduced from the PK-GEVD-DANSE computational complexity), within the class of algorithms considered in this paper. It will be shown that, this pPK-GEVD-DANSE algorithm again achieves a performance as if all signals were available to each and every node. Implementations of the pPK-GEVD-DANSE algorithm are presented using the normalized least mean squares (NLMS) algorithm and QR decomposition based recursive least squares (QRD-RLS) algorithm in the AEC stage. Furthermore, monitoring of the loudspeaker activity by means of a voice activity detector (VAD) is proposed.

The paper is organized as follows. The data model is presented in Section II. The formulations for the centralized integrated and cascade algorithm are provided in Sections III and IV. The distributed integrated and cascade algorithm are described in Sections V and VI. Section VII describes the NLMS- and QRD-RLS-based algorithm in the AEC stage of the pPK-GEVD-DANSE algorithm. Simulations are shown in

Section VIII, and finally Section IX concludes the paper.

II. Problem formulation and notation

Consider a fully connected WASAN with K nodes (see Fig. 1), where node $k \in \mathcal{K} = \{1, \ldots, K\}$ contains m_k microphones and l_k loudspeakers, and hence has access to the short-time Fourier transform (STFT) domain $n_k \times 1$ signal vector $\mathbf{y}_k(\kappa, l) = \begin{bmatrix} \mathbf{x}_k(\kappa, l) \\ \mathbf{u}_k(\kappa, l) \end{bmatrix}$, where κ is the frequency bin index, l the frame index (for brevity κ and l will be omitted in the following, except for a few cases where l has to be included explicitly) and, $n_k = m_k + Pl_k$. Vector $\mathbf{u}_k$ contains l_k local loudspeaker signals sampled at the current and previous $P-1$ frames, i.e.,

$$\mathbf{u}_k(l) = \begin{bmatrix} u_1(l) \\ \vdots \\ u_1(l-P+1) \\ \vdots \\ u_{l_k}(l) \\ \vdots \\ u_{l_k}(l-P+1) \end{bmatrix}. \tag{1}$$

Vector $\mathbf{x}_k$ contains m_k local microphone signals sampled only at the current frame and is modeled as

$$\mathbf{x}_k = \mathbf{s}_k + \mathbf{n}_k = \mathbf{a}_k s + \mathbf{n}_k. \tag{2}$$

Here, s is the desired speech source signal (also known as the dry signal), $\mathbf{a}_k$ contains the acoustic transfer functions from the desired speech source position to the local microphones, $\mathbf{s}_k$ is the desired speech component and $\mathbf{n}_k$ is the noise component in the microphone signals of node k, modeled as

$$\mathbf{n}_k = \mathbf{G}_{kk}\mathbf{u}_k + \sum_{q \neq k} \mathbf{G}_{kq}\mathbf{u}_q + \mathbf{b}_k \tag{3}$$

where $\mathbf{G}_{kk}$ is an $m_k \times Pl_k$ matrix representing the local echo paths from the local loudspeakers to the local microphones, $\mathbf{G}_{kq}$ is an $m_k \times Pl_q$ matrix representing the echo paths from the loudspeakers in node q to the microphones in node k and finally $\mathbf{u}_q$ contains the loudspeaker signals from node q. The background noise is assumed to be stationary with correlation matrix

$$\bar{\mathbf{R}}_{\mathbf{b}_k\mathbf{b}_k} = E\{\mathbf{b}_k\mathbf{b}_k^H\} \tag{4}$$

where $(\cdot)^H$ denotes the conjugate transpose operator and $E\{\cdot\}$ is the expected value operator. The following vectors are also defined,

$$\tilde{\mathbf{s}}_k = \left[\mathbf{s}_k^H \ \mathbf{0}_{1\times Pl_k}\right]^H, \tag{5}$$

$$\tilde{\mathbf{n}}_k = \left[\mathbf{n}_k^H \ \mathbf{u}_k^H\right]^H, \tag{6}$$

$$\tilde{\mathbf{a}}_k = \left[\mathbf{a}_k^H \ \mathbf{0}_{1\times Pl_k}\right]^H, \tag{7}$$

$$\tilde{\mathbf{b}}_k = \left[\mathbf{b}_k^H \ \mathbf{0}_{1\times Pl_k}\right]^H, \tag{8}$$

where $\mathbf{0}_{1\times Pl_k}$ is a Pl_k-dimensional all-zero vector, and so that

$$\mathbf{y}_k = \tilde{\mathbf{s}}_k + \tilde{\mathbf{n}}_k = \tilde{\mathbf{a}}_k s + \tilde{\mathbf{n}}_k. \tag{9}$$

The N-dimensional vectors ($N = \sum_{k=1}^{K} n_k$), $\mathbf{y}$, $\mathbf{s}$, $\mathbf{n}$, $\mathbf{a}$ and $\mathbf{b}$ are the stacked versions of $\mathbf{y}_k$, $\tilde{\mathbf{s}}_k$, $\tilde{\mathbf{n}}_k$, $\tilde{\mathbf{a}}_k$ and $\tilde{\mathbf{b}}_k$ respectively, such that the signal vector $\mathbf{y}$ can be characterized as follows

$$\mathbf{y} = \mathbf{s} + \mathbf{n} = \mathbf{a}s + \mathbf{n}. \tag{10}$$

Assuming that the desired speech source signal and background noise are uncorrelated, and uncorrelated with the loudspeaker signals, correlation matrices can be defined as follows

$$\bar{\mathbf{R}}_{\mathbf{yy}} = E\{\mathbf{y}\mathbf{y}^H\} = E\{\mathbf{s}\mathbf{s}^H\} + E\{\mathbf{n}\mathbf{n}^H\} = \bar{\mathbf{R}}_{\mathbf{ss}} + \bar{\mathbf{R}}_{\mathbf{nn}} \tag{11}$$

$$\bar{\mathbf{R}}_{\mathbf{ss}} = \mathbf{a}\phi_s\mathbf{a}^H \tag{12}$$

$$\bar{\mathbf{R}}_{\mathbf{nn}} = \mathbf{G}\Phi_{\mathbf{u}}\mathbf{G}^H + \bar{\mathbf{R}}_{\mathbf{bb}} \tag{13}$$

$$\begin{aligned} \bar{\mathbf{R}}_{\mathbf{bb}} &= E\{\mathbf{b}\mathbf{b}^H\} \\ &= \text{blockdiag}\{\bar{\mathbf{R}}_{\mathbf{b}_1\mathbf{b}_1}, \mathbf{0}, \bar{\mathbf{R}}_{\mathbf{b}_2\mathbf{b}_2}, \mathbf{0}, \ldots, \bar{\mathbf{R}}_{\mathbf{b}_k\mathbf{b}_k}, \mathbf{0}\} \end{aligned} \tag{14}$$

where ϕ_s is the power spectral density (PSD) of the desired speech source signal, the operator $\text{blockdiag}\{\cdot\}$ creates a block-diagonal matrix from its input arguments, $\Phi_{\mathbf{u}} = E\{\mathbf{u}\mathbf{u}^H\}$ a $PL \times PL$ matrix representing the PSD of the loudspeaker signals ($L = \sum_{k=1}^{K} l_k$) with the PL-dimensional vector $\mathbf{u}$ the stacked version of $\mathbf{u}_k$ and

$$\tilde{\mathbf{G}}_{kk} = \left[\mathbf{G}_{kk}^H \ \mathbf{I}_{Pl_k\times Pl_k}\right]^H, \tag{15}$$

$$\tilde{\mathbf{G}}_{kq} = \left[\mathbf{G}_{kq}^H \ \mathbf{0}_{Pl_q\times Pl_k}\right]^H, \tag{16}$$

$$\mathbf{G} = \begin{bmatrix} \tilde{\mathbf{G}}_{11} & \ldots & \tilde{\mathbf{G}}_{1K} \\ \vdots & \ddots & \vdots \\ \tilde{\mathbf{G}}_{K1} & \ldots & \tilde{\mathbf{G}}_{KK} \end{bmatrix}. \tag{17}$$

Given that loudspeaker signals are generally non-stationary, e.g., speech and/or music signals, $\Phi_{\mathbf{u}}(l) \neq \Phi_{\mathbf{u}}(l')$ for $l \neq l'$. It is first assumed that $\Phi_{\mathbf{u}}(l) = \Phi_{\mathbf{u}}(l')$, $\forall l$, so that the noise $\mathbf{n}$ is stationary, as required in the MWF algorithm in Section III. However this assumption will be revisited in Section III-A. In addition, it is also assumed that all signals in the network are perfectly synchronized i.e. there is no sampling rate offset between signals in different nodes. Sampling rate offset estimation and compensation in a WASAN can be tackled similarly to sampling rate offset estimation and compensation in a wireless acoustic sensor network (WASN), e.g. based on [28]–[30] or similar.

III. Centralized integrated AEC and NR (MWF)

The node-specific combined AEC and NR task for node k is to estimate the desired signal d_k, defined here as the speech component in the first local microphone, i.e, $d_k = [1\ \mathbf{0}]\,\mathbf{s}_k = \mathbf{e}_{d_k}^H\mathbf{s}$, where $\mathbf{0}$ is an all-zero vector with matching dimensions and $\mathbf{e}_{d_k}^H$ is a vector that selects the desired speech component in $\mathbf{s}$. The minimization of the mean squared error (MSE) between the desired signal and the filtered microphone and loudspeaker signals defines an optimal filter for node k,

$$\bar{\mathbf{w}}_k = \underset{\mathbf{w}_k}{\arg\min}\ E\left\{\left|d_k - \mathbf{w}_k^H\mathbf{y}\right|^2\right\}. \tag{18}$$

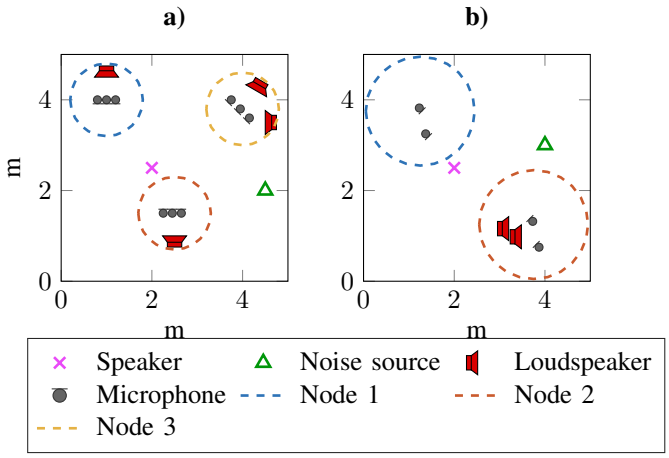

Fig. 1: Two example scenarios for a WASAN with a single target speaker and a single noise source: **a)**. Three nodes each with 3 microphones and 1 or 2 loudspeakers. **b)**. Two nodes each with 2 microphones. One node with a stereo loudspeaker signal.

The node-specific signal estimate is then obtained as $\hat{d}_k = \bar{\mathbf{w}}_k^H \mathbf{y}$. The solution to this is the well-known MWF [31], [32], given by

$$\bar{\mathbf{w}}_k = \bar{\mathbf{R}}_{\mathbf{yy}}^{-1} \bar{\mathbf{R}}_{\mathbf{y}d_k} = \bar{\mathbf{R}}_{\mathbf{yy}}^{-1} \bar{\mathbf{R}}_{\mathbf{ys}} \mathbf{e}_{d_k} = \bar{\mathbf{R}}_{\mathbf{yy}}^{-1} \bar{\mathbf{R}}_{\mathbf{ss}} \mathbf{e}_{d_k} \tag{19}$$

where $\bar{\mathbf{R}}_{\mathbf{y}d_k} = E\{\mathbf{y} d_k^H\}$ and $\bar{\mathbf{R}}_{\mathbf{ys}} = E\{\mathbf{y}\mathbf{s}^H\}$. The final expression in (19) is obtained based on the assumption that s and $\mathbf{n}$ are uncorrelated (Section II).

In practice, by using a voice activity detector (VAD), $\bar{\mathbf{R}}_{\mathbf{yy}}$ and $\bar{\mathbf{R}}_{\mathbf{nn}}$ are first estimated during *speech-plus-noise* periods where the desired speech signal, loudspeaker signals and background noise are active, and *noise-only* periods where there is no activity of the desired speech signal and the other signals are active, respectively [33], i.e.,

$$\begin{aligned} &\text{if VAD}(l) = 1: \\ &\hat{\mathbf{R}}_{\mathbf{yy}}(l) = \beta \hat{\mathbf{R}}_{\mathbf{yy}}(l-1) + (1-\beta)\mathbf{y}(l)\mathbf{y}^H(l), \\ &\text{if VAD}(l) = 0: \\ &\hat{\mathbf{R}}_{\mathbf{nn}}(l) = \beta \hat{\mathbf{R}}_{\mathbf{nn}}(l-1) + (1-\beta)\mathbf{y}(l)\mathbf{y}^H(l), \end{aligned} \tag{20}$$

where $\hat{\mathbf{R}}_{\mathbf{yy}}(l)$, $\hat{\mathbf{R}}_{\mathbf{nn}}(l)$, $\mathbf{y}(l)$ represent $\hat{\mathbf{R}}_{\mathbf{yy}}$, $\hat{\mathbf{R}}_{\mathbf{nn}}$ and $\mathbf{y}$ at frame l, respectively. The forgetting factor $0 < \beta < 1$ can be chosen depending on the variation of the statistics of the signals, i.e., if the statistics change slowly then β should be chosen close to 1 to obtain long-term estimates that mainly capture the spatial coherence between the microphone signals. For the time being, it is assumed that the loudspeaker signals and background noise are stationary (Section II), so that their contribution in $\hat{\mathbf{R}}_{\mathbf{yy}}$ and $\hat{\mathbf{R}}_{\mathbf{nn}}$ is the same. The following criterion will then be used to estimate $\bar{\mathbf{R}}_{\mathbf{ss}}$ [27], [31],

$$\hat{\mathbf{R}}_{\mathbf{ss}} = \underset{\substack{\text{rank}(\mathbf{R}_{\mathbf{ss}})=1 \\ \mathbf{R}_{\mathbf{ss}} \succeq 0}}{\arg\min} \left|\left| \hat{\mathbf{R}}_{\mathbf{nn}}^{-1/2} \left(\hat{\mathbf{R}}_{\mathbf{yy}} - \hat{\mathbf{R}}_{\mathbf{nn}} - \mathbf{R}_{\mathbf{ss}} \right) \hat{\mathbf{R}}_{\mathbf{nn}}^{-H/2} \right|\right|_F^2 \tag{21}$$

where $||\cdot||_F$ denotes the Frobenius norm. Spatial pre-whitening is applied by pre- and post-multiplying by $\hat{\mathbf{R}}_{\mathbf{nn}}^{-1/2}$ and $\hat{\mathbf{R}}_{\mathbf{nn}}^{-H/2}$, respectively. The solution to (21) is based on a generalized eigenvalue decomposition (GEVD) of the $(N \times N)$ matrix pencil $\{\hat{\mathbf{R}}_{\mathbf{yy}}, \hat{\mathbf{R}}_{\mathbf{nn}}\}$ [31], [34]

$$\begin{aligned} \hat{\mathbf{R}}_{\mathbf{yy}} &= \hat{\mathbf{Q}} \hat{\mathbf{\Sigma}}_{\mathbf{yy}} \hat{\mathbf{Q}}^H, \\ \hat{\mathbf{R}}_{\mathbf{nn}} &= \hat{\mathbf{Q}} \hat{\mathbf{\Sigma}}_{\mathbf{nn}} \hat{\mathbf{Q}}^H, \end{aligned} \tag{22}$$

where $\hat{\mathbf{\Sigma}}_{\mathbf{yy}}$ and $\hat{\mathbf{\Sigma}}_{\mathbf{nn}}$ are diagonal matrices and $\hat{\mathbf{Q}}$ is an invertible matrix. The speech correlation matrix estimate $\hat{\mathbf{R}}_{\mathbf{ss}}$ is then [31]

$$\hat{\mathbf{R}}_{\mathbf{ss}} = \hat{\mathbf{Q}} \text{diag}\{\hat{\sigma}_{y_1} - \hat{\sigma}_{n_1}, 0, \ldots, 0\} \hat{\mathbf{Q}}^H \tag{23}$$

where $\hat{\sigma}_{y_1}$ and $\hat{\sigma}_{n_1}$ are the first diagonal element of $\hat{\mathbf{\Sigma}}_{\mathbf{yy}}$ and $\hat{\mathbf{\Sigma}}_{\mathbf{nn}}$, respectively, corresponding to the largest ratio $\hat{\sigma}_{y_i}/\hat{\sigma}_{n_i}$. Using (23) and $\hat{\mathbf{R}}_{\mathbf{yy}}$ (cfr. (22)) in (19), the MWF estimate $\hat{\mathbf{w}}_k$ can be expressed as

$$\hat{\mathbf{w}}_k = \hat{\mathbf{Q}}^{-H} \text{diag}\left\{ 1 - \frac{\hat{\sigma}_{n_1}}{\hat{\sigma}_{y_1}}, 0, \ldots, 0 \right\} \hat{\mathbf{Q}}^H \mathbf{e}_{d_k}. \tag{24}$$

The node-specific signal estimate is then obtained as $\hat{d}_k = \hat{\mathbf{w}}_k^H \mathbf{y}$. In this integrated algorithm, the MWF estimate depends on the loudspeaker signal statistics without exploiting the prior knowledge that there is no desired speech component in these loudspeaker signals. As a consequence, the combined AEC and NR fundamentally consists of a single NR stage in which acoustic echo is treated similarly to background noise.

A. Non-stationarity of loudspeaker signals and MWF assumptions

As mentioned in Section II, the loudspeaker signals are generally non-stationary, i.e., $\Phi_{\mathbf{u}}(l) \neq \Phi_{\mathbf{u}}(l')$ for $l \neq l'$. As a consequence their contribution in the *speech-plus-noise* and *noise-only* correlation matrices, $\hat{\mathbf{R}}_{\mathbf{yy}}$ and $\hat{\mathbf{R}}_{\mathbf{nn}}$, respectively, may be different. This violates the basic stationarity assumption in the MWF algorithm described above. However, it is observed that this non-stationarity does not change significantly the GEVD of the matrix pencil $\{\hat{\mathbf{R}}_{\mathbf{yy}}, \hat{\mathbf{R}}_{\mathbf{nn}}\}$ because of the specific structure of $\hat{\mathbf{R}}_{\mathbf{yy}}$ and $\hat{\mathbf{R}}_{\mathbf{nn}}$ corresponding to the fact that the loudspeaker signals do not contain any desired speech and background noise component. In particular, this will lead to the following structure in $\hat{\mathbf{Q}}$, $\hat{\mathbf{\Sigma}}_{\mathbf{yy}}$ and $\hat{\mathbf{\Sigma}}_{\mathbf{nn}}$ in (22)

$$\hat{\mathbf{Q}} = \left[\underset{N\times 1}{\hat{\mathbf{q}}_1} \middle| \underset{N\times PL}{\hat{\mathbf{Q}}_1} \middle| \hat{\mathbf{q}}_2 \cdots \hat{\mathbf{q}}_M \right], \tag{25}$$

$$\hat{\mathbf{\Sigma}}_{\mathbf{yy}} = \begin{bmatrix} \hat{\sigma}_{y_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underset{PL\times PL}{\hat{\mathbf{\Sigma}}_{\mathbf{yy},1}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \underset{M-1\times M-1}{\hat{\mathbf{\Sigma}}_{\mathbf{yy},2}} \end{bmatrix}, \tag{26}$$

$$\hat{\mathbf{\Sigma}}_{\mathbf{nn}} = \begin{bmatrix} \hat{\sigma}_{n_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underset{PL\times PL}{\hat{\mathbf{\Sigma}}_{\mathbf{nn},1}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \underset{M-1\times M-1}{\hat{\mathbf{\Sigma}}_{\mathbf{nn},2}} \end{bmatrix}, \tag{27}$$

where $(\hat{\mathbf{q}}_1 \ldots \hat{\mathbf{q}}_M)$ are column vectors uniquely defined by the desired speech component and background noise ($M =$

$\sum_{k=1}^{K} m_k$), hence containing zeros in the positions corresponding to the loudspeaker signals, and $\hat{\mathbf{Q}}_1$ contains PL columns which are uniquely defined by the loudspeaker signals and echo paths. The non-stationarity of the loudspeaker signals does not modify $(\hat{\mathbf{q}}_1 \dots \hat{\mathbf{q}}_M)$, $\hat{\sigma}_{y_1}$,$\hat{\sigma}_{n_1}$, $\hat{\mathbf{\Sigma}}_{\mathbf{yy},2}$ and $\hat{\mathbf{\Sigma}}_{\mathbf{nn},2}$. It also does not modify the column space spanned by $\hat{\mathbf{Q}}_1$. As a result, the first column of $\hat{\mathbf{Q}}^H$ in (24) is not modified, as well as all other relevant quantities in (24). Therefore, the MWF estimate in (24) is also not modified. Note that it is assumed here that the GEVLs corresponding to $\hat{\mathbf{Q}}_1$ are smaller than the GEVL corresponding to $\hat{\mathbf{q}}_1$, i.e. to the desired speech signal, so the latter continues to be the largest GEVL. For the unlikely scenario that a GEVL corresponding to $\hat{\mathbf{Q}}_1$ becomes the largest GEVL, $\hat{\mathbf{q}}_1$ may be monitored (based on its zeros structure) and tracked, so that the correct GEVL is still chosen.

IV. Centralized cascade AEC and NR (PK-MWF)

Exploiting the prior knowledge that $\bar{\mathbf{R}}_{\mathbf{ss}}$ has a specific zero structure (cfr. definition of $\mathbf{s}$ and $\tilde{\mathbf{s}}_k$), the criterion in (21) can be redefined as

$$\hat{\mathbf{R}}_{\mathbf{ss}} = \underset{\substack{\text{rank}(\mathbf{R}_{\mathbf{ss}})=1 \\ \mathbf{B}^H\mathbf{R}_{\mathbf{ss}}\mathbf{B}=0 \\ \mathbf{R}_{\mathbf{ss}}\succeq 0}}{\arg\min} \left|\left| \hat{\mathbf{R}}_{\mathbf{nn}}^{-1/2} \left(\hat{\mathbf{R}}_{\mathbf{yy}} - \hat{\mathbf{R}}_{\mathbf{nn}} - \mathbf{R}_{\mathbf{ss}} \right) \hat{\mathbf{R}}_{\mathbf{nn}}^{-H/2} \right|\right|_F^2 \tag{28}$$

where $\mathbf{B}$ is an $N \times PL$ block diagonal matrix

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_1 & 0 & \dots & 0 \\ 0 & \mathbf{B}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{B}_K \end{bmatrix} \tag{29}$$

with the kth diagonal block $\mathbf{B}_k$ equal to

$$\mathbf{B}_k = \begin{bmatrix} \mathbf{0}_{m_k \times Pl_k} \\ \mathbf{I}_{Pl_k} \end{bmatrix}, \tag{30}$$

where $\mathbf{I}_{Pl_k}$ is a $Pl_k \times Pl_k$ identity matrix. In the combined AEC and NR context $\mathbf{B}$ is a selection matrix that selects the loudspeaker signals. In [27] it is shown that the inclusion of the constraint $\mathbf{B}^H\mathbf{R}_{\mathbf{ss}}\mathbf{B} = 0$ leads to the reduced dimensional $(M \times M)$ matrix pencil $\{\mathbf{R}_{\mathbf{yy}}^{\text{red}}, \mathbf{R}_{\mathbf{nn}}^{\text{red}}\}$ with GEVD

$$\begin{aligned} \hat{\mathbf{R}}_{\mathbf{yy}}^{\text{red}} &= \hat{\mathbf{Q}}^{\text{red}} \hat{\mathbf{\Sigma}}_{\mathbf{yy}}^{\text{red}} (\hat{\mathbf{Q}}^{\text{red}})^H, \\ \hat{\mathbf{R}}_{\mathbf{nn}}^{\text{red}} &= \hat{\mathbf{Q}}^{\text{red}} \hat{\mathbf{\Sigma}}_{\mathbf{nn}}^{\text{red}} (\hat{\mathbf{Q}}^{\text{red}})^H, \end{aligned} \tag{31}$$

where $\hat{\mathbf{R}}_{\mathbf{yy}}^{\text{red}} = \hat{\mathbf{C}}^H \hat{\mathbf{R}}_{\mathbf{yy}} \hat{\mathbf{C}}$, $\hat{\mathbf{R}}_{\mathbf{nn}}^{\text{red}} = \hat{\mathbf{C}}^H \hat{\mathbf{R}}_{\mathbf{nn}} \hat{\mathbf{C}}$, $\mathbf{y}^{\text{red}} = \hat{\mathbf{C}}^H \mathbf{y}$, and with $\hat{\mathbf{C}}$ an $N \times M$ matrix obtained from the linearly-constrained minimum variance (LCMV) beamformer optimization criterion

$$\hat{\mathbf{C}} = \underset{\text{s.t.} \quad \mathbf{H}^H\mathbf{C}=\mathbf{I}_M}{\arg\min} \left|\left| \text{trace}\{\mathbf{C}^H \hat{\mathbf{R}}_{\mathbf{nn}} \mathbf{C}\} \right|\right| \tag{32}$$

where $\mathbf{H}$ is a $N \times M$ block diagonal matrix

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & 0 & \dots & 0 \\ 0 & \mathbf{H}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{H}_K \end{bmatrix} \tag{33}$$

with the kth diagonal block equal to

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{I}_{m_k} \\ \mathbf{0}_{Pl_k \times m_k} \end{bmatrix}, \tag{34}$$

such that $\mathbf{H}^H\mathbf{H} = \mathbf{I}_M$ and $\mathbf{B}^H\mathbf{H} = \mathbf{0}$. Hence $\hat{\mathbf{C}}$ can be defined based on a generalized sidelobe canceller (GSC) implementation as [27], [35]

$$\hat{\mathbf{C}} = \mathbf{H} - \mathbf{B}\hat{\mathbf{F}}, \tag{35}$$

$$\hat{\mathbf{F}} = (\mathbf{B}^H \hat{\mathbf{R}}_{\mathbf{nn}} \mathbf{B})^{-1} \mathbf{B}^H \hat{\mathbf{R}}_{\mathbf{nn}} \mathbf{H}, \tag{36}$$

where the filter $\hat{\mathbf{F}}$ operates on the loudspeaker signals ($\mathbf{B}^H\mathbf{y}$) and effectively serves as an AEC filter cancelling the echo components in the so-called fixed beamformer outputs corresponding to $\mathbf{H}$, i.e., the microphone signals ($\mathbf{H}^H\mathbf{y}$). The inclusion of the prior knowledge thus leads to a cascade algorithm where AEC is performed first and then NR. The AEC filter $\hat{\mathbf{F}}$ can also be implemented adaptively via an NLMS or QRD-RLS algorithm as will be explained in Section VII.

The prior knowledge speech correlation matrix estimate $\hat{\mathbf{R}}_{\mathbf{ss}}$, i.e., the solution to (28), is then given as [27], [31],

$$\hat{\mathbf{R}}_{\mathbf{ss}} = \mathbf{H}\hat{\mathbf{Q}}^{\text{red}} \text{diag}\{\hat{\sigma}_{y_1} - \hat{\sigma}_{n_1}, 0, \dots, 0\} (\hat{\mathbf{Q}}^{\text{red}})^H \mathbf{H}^H, \tag{37}$$

where $\hat{\sigma}_{y_1}$ and $\hat{\sigma}_{n_1}$ are the first diagonal element of $\hat{\mathbf{\Sigma}}_{\mathbf{yy}}^{\text{red}}$ and $\hat{\mathbf{\Sigma}}_{\mathbf{nn}}^{\text{red}}$, respectively, corresponding to the largest ratio $\hat{\sigma}_{y_i}/\hat{\sigma}_{n_i}$. Using this expression and the reduced dimensional $\hat{\mathbf{R}}_{\mathbf{yy}}^{\text{red}}$ (cfr. (31)), the PK-MWF estimate $\hat{\mathbf{w}}_k$ can finally be expressed as [27]

$$\hat{\mathbf{w}}_k = \hat{\mathbf{C}} (\hat{\mathbf{Q}}^{\text{red}})^{-H} \text{diag} \left\{ 1 - \frac{\hat{\sigma}_{n_1}}{\hat{\sigma}_{y_1}}, 0, \dots, 0 \right\} (\hat{\mathbf{Q}}^{\text{red}})^H \mathbf{H}^H \mathbf{e}_{d_k}. \tag{38}$$

The non-stationarity of the loudspeaker signals in this case does not affect the NR stage, as the joint-diagonalization is performed on the reduced dimensional $(M \times M)$ matrix pencil $\{\mathbf{R}_{\mathbf{yy}}^{\text{red}}, \mathbf{R}_{\mathbf{nn}}^{\text{red}}\}$, therefore $\hat{\mathbf{Q}}^{\text{red}}$ will only have M columns defined by the desired speech components and background noise, and the echo signals are effectively removed by the AEC stage.

V. Distributed integrated AEC and NR (GEVD-DANSE)

The integrated AEC and NR algorithm of Section III can be implemented in a distributed fashion by means of the GEVD-DANSE algorithm [24] where each node instead of broadcasting n_k microphone and loudspeaker signals, broadcasts only 1 fused signal to the other nodes. Each node performs local operations, corresponding to a reduced dimensional version (dimension $n_k + (K-1)$ in node k) of the MWF-based integrated AEC and NR algorithm of Section III (dimension N), based on n_k local microphone and loudspeaker signals and $(K-1)$ fused signals received from the other nodes. The fused signal broadcast by node k is

$$z_k = \hat{\mathbf{p}}_k^H \mathbf{y}_k \tag{39}$$

where $\hat{\mathbf{p}}_k$ is an n_k-dimensional fusion vector. Then each node has access to a signal vector $\check{\mathbf{y}}_k = \left[\mathbf{y}_k^H \ \mathbf{z}_{-k}^H\right]^H$, where the

subscript $_{-k}$ refers to the concatenation of the fused signals of nodes other than k, so that $\mathbf{z}_{-k} = [z_1^* \dots z_{k-1}^* z_{k+1}^* \dots z_K^*]^H$, where * represents the complex conjugate. The local filter $\hat{\mathbf{w}}_k$ is defined as

$$\hat{\mathbf{w}}_k = \hat{\mathbf{Q}}_k^{-H} \text{diag}\left\{1 - \frac{\hat{\sigma}_{n_1}}{\hat{\sigma}_{y_1}}, 0, \dots, 0\right\} \hat{\mathbf{Q}}_k^H [1\ \mathbf{0}]^H \tag{40}$$

with the GEVD of the $(n_k + K - 1) \times (n_k + K - 1)$ matrix pencil $\{\hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}, \hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}\}$ given as

$$\begin{aligned} \hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} &= \hat{\mathbf{Q}}_k \hat{\mathbf{\Sigma}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} \hat{\mathbf{Q}}_k^H, \\ \hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} &= \hat{\mathbf{Q}}_k \hat{\mathbf{\Sigma}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} \hat{\mathbf{Q}}_k^H, \end{aligned} \tag{41}$$

where $\hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}$ is an estimate of $\bar{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} = E\{\check{\mathbf{y}}_k\check{\mathbf{y}}_k^H\}$, $\hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}$ is an estimate of $\bar{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} = E\{\check{\mathbf{n}}_k\check{\mathbf{n}}_k^H\}$ and $\check{\mathbf{n}}_k$ corresponds to $\check{\mathbf{y}}_k$ in *noise-only* periods. The fusion vector is finally defined as

$$\hat{\mathbf{p}}_k = [\mathbf{I}_{n_k}\ \mathbf{0}]\hat{\mathbf{w}}_k. \tag{42}$$

In each time frame the nodes broadcast fused signals (39) using their current fusion vectors. One node then updates its fusion vector by means of (40)-(42). When the nodes update sequentially in a round-robin fashion (e.g. one node updates per time frame) the local signal estimates $\hat{d}_k = \hat{\mathbf{w}}_k^H \check{\mathbf{y}}_k$ have been shown to converge in each node to the centralized signal estimates obtained with (24) [24]. It has also been shown that when the nodes update simultaneously a relaxation factor (α_{rS}) is needed to avoid limit cycles. With this each filter is updated as a convex combination of its previous and newly computed version in (40) [24], [36].

VI. Distributed cascade AEC and NR (pPK-GEVD-DANSE)

The cascade AEC and NR algorithm of Section IV can be implemented in a distributed fashion by means of the PK-GEVD-DANSE algorithm [27] where each node broadcasts 2 fused signals, i.e., a desired signal reference and a noise reference. In the context of combined AEC and NR, the second fused signal will be a fused loudspeaker signal. Each node then performs local operations, effectively corresponding to a reduced dimensional version (dimension $n_k+2(K-1)$ in node k) of the PK-MWF-based cascade AEC and NR algorithm of Section IV (dimension N), based on n_k local microphone and loudspeaker signals and $2(K-1)$ fused signals received from the other nodes. The first fused signal broadcast by node k is given by (39) with the n_k-dimensional fusion vector $\hat{\mathbf{p}}_k$ to be redefined. The second fused signal broadcast by node k is

$$\underline{z}_k = \hat{\boldsymbol{\lambda}}_k^H \mathbf{B}_k^H \mathbf{y}_k = \hat{\boldsymbol{\lambda}}_k^H \mathbf{u}_k \tag{43}$$

where $\hat{\boldsymbol{\lambda}}_k$ is a Pl_k-dimensional fusion vector. Then each node has access to a signal vector $\check{\mathbf{y}}_k = \left[\mathbf{y}_k^H\ \mathbf{z}_{-k}^H\ \underline{\mathbf{z}}_{-k}^H\right]^H$, where $\mathbf{z}_{-k}$ is defined as in Section V and $\underline{\mathbf{z}}_{-k} = [\underline{z}_1^* \cdots \underline{z}_{k-1}^* \underline{z}_{k+1}^* \cdots \underline{z}_K^*]^H$. A modification must be introduced in $\mathbf{H}_k$ and $\mathbf{B}_k$ to account for the extra signals broadcast from the other nodes, hence

$$\begin{aligned} \check{\mathbf{H}}_k &= \left[\begin{array}{c|c} \mathbf{I}_{m_k} & \mathbf{0} \\ \mathbf{0}_{Pl_k \times m_k} & \\ \hline \mathbf{0} & \begin{array}{c}\mathbf{I}_{K-1} \\ \mathbf{0}_{(K-1)\times(K-1)}\end{array} \end{array}\right], \\ \check{\mathbf{B}}_k &= \left[\begin{array}{c|c} \mathbf{0}_{m_k \times Pl_k} & \mathbf{0} \\ \mathbf{I}_{Pl_k} & \\ \hline \mathbf{0} & \begin{array}{c}\mathbf{0}_{(K-1)\times(K-1)} \\ \mathbf{I}_{K-1}\end{array} \end{array}\right], \end{aligned} \tag{44}$$

where $\check{\mathbf{H}}_k$ is an $(n_k + 2(K-1)) \times (m_k + K - 1)$ matrix and $\check{\mathbf{B}}_k$ is an $(n_k + 2(K-1)) \times (Pl_k + K - 1)$ matrix. Then equations (35) and (36) become respectively

$$\hat{\mathbf{C}}_k = \check{\mathbf{H}}_k - \check{\mathbf{B}}_k \hat{\mathbf{F}}_k, \tag{45}$$

$$\hat{\mathbf{F}}_k = (\check{\mathbf{B}}_k^H \hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} \check{\mathbf{B}}_k)^{-1} \check{\mathbf{B}}_k^H \hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} \check{\mathbf{H}}_k, \tag{46}$$

where $\hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}$ is an estimate of $\bar{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} = E\{\check{\mathbf{y}}_k\check{\mathbf{y}}_k^H\}$, $\hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}$ is an estimate of $\bar{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} = E\{\check{\mathbf{n}}_k\check{\mathbf{n}}_k^H\}$ and $\check{\mathbf{n}}_k$ corresponds to $\check{\mathbf{y}}_k$ in *noise-only* periods. The fusion vectors are defined as in (42) and as [27]

$$\hat{\boldsymbol{\lambda}}_k = [\mathbf{I}_{Pl_k}\ \mathbf{0}](\check{\mathbf{B}}_k^H \hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} \check{\mathbf{B}}_k)^{-1} \check{\mathbf{B}}_k^H \hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} \hat{\mathbf{w}}_k \tag{47}$$

where the local filter $\hat{\mathbf{w}}_k$ is defined as

$$\hat{\mathbf{w}}_k = \hat{\mathbf{C}}_k (\hat{\mathbf{Q}}_k^{\text{red}})^{-H} \text{diag}\left\{1 - \frac{\hat{\sigma}_{n_1}}{\hat{\sigma}_{y_1}}, 0, \dots, 0\right\} (\hat{\mathbf{Q}}_k^{\text{red}})^H \check{\mathbf{H}}_k^H \mathbf{e}_{d_k} \tag{48}$$

with the GEVD of the reduced dimensional $(m_k + K - 1) \times (m_k + K - 1)$ pencil $\{\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}, \hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}\}$ given as

$$\begin{aligned} \hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} &= \hat{\mathbf{Q}}_k^{\text{red}} \hat{\mathbf{\Sigma}}^{\text{red}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} (\hat{\mathbf{Q}}_k^{\text{red}})^{\text{H}}, \\ \hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} &= \hat{\mathbf{Q}}_k^{\text{red}} \hat{\mathbf{\Sigma}}^{\text{red}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} (\hat{\mathbf{Q}}_k^{\text{red}})^{\text{H}}, \end{aligned} \tag{49}$$

where $\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} = \hat{\mathbf{C}}_k^H \hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} \hat{\mathbf{C}}_k$, $\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} = \hat{\mathbf{C}}_k^H \hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} \hat{\mathbf{C}}_k$ and $\check{\mathbf{y}}_k^{\text{red}} = \hat{\mathbf{C}}_k^H \check{\mathbf{y}}_k$. In each time frame the nodes broadcast fused signals (39) and (43) using their current fusion vectors. One node then updates its fusion vectors by means of (44)-(49). When the nodes update sequentially in a round-robin fashion (e.g. one node updates per time frame) the local signal estimates $\hat{d}_k = \hat{\mathbf{w}}_k^H \check{\mathbf{y}}_k$ have been shown to converge in each node to the centralized signal estimates obtained with (38) [27].

In an AEC context (unlike the general PK-GEVD-DANSE context), the above algorithm and its communication cost can be reduced as follows. First, the loudspeaker signals ($\check{\mathbf{B}}_k^H \check{\mathbf{y}}_k$) do not contain any desired speech component, hence

$$\check{\mathbf{B}}_k^H \hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k} \hat{\mathbf{w}}_k \simeq \check{\mathbf{B}}_k^H \hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k} \hat{\mathbf{w}}_k = \mathbf{0}, \tag{50}$$

which in (47) leads to $\hat{\boldsymbol{\lambda}}_k \simeq \mathbf{0}$. In (50) the "$\simeq$" is replaced by an equality if estimated correlation matrices are replaced by true statistical quantities. The "$=$" in (50) is obtained by substituting (45), (46) and (48). Therefore the extra fused signal $\underline{z}_k$ does not need to be communicated to the other nodes. Therefore, $\check{\mathbf{H}}_k$ and $\check{\mathbf{B}}_k$ are reduced to

$$\check{\mathbf{H}}_k = \left[\begin{array}{c|c} \mathbf{I}_{m_k} & \mathbf{0} \\ \mathbf{0} & \\ \hline \mathbf{0} & \mathbf{I}_{K-1} \end{array}\right], \ \check{\mathbf{B}}_k = \left[\begin{array}{c} \mathbf{0} \\ \hline \mathbf{I}_{Pl_k} \\ \hline \mathbf{0} \end{array}\right] \tag{51}$$

with $\check{\mathbf{H}}_k$ an $(n_k+K-1)\times(m_k+K-1)$ matrix and $\check{\mathbf{B}}_k$ an $(n_k+K-1)\times Pl_k$ matrix. The vector $\check{\mathbf{y}}_k$ is then reduced to

$$\check{\mathbf{y}}_k = \begin{bmatrix}\mathbf{y}_k \\ \mathbf{z}_{-k}\end{bmatrix}. \tag{52}$$

This effectively leads to a distributed cascade algorithm where each node k shares only 1 fused signal with the other nodes. Each node then performs local operations corresponding to a reduced dimensional version (dimension n_k+K-1 in node k) of the PK-MWF-based cascade AEC and NR algorithm of Section IV. The resulting algorithm, referred to as the pruned PK-GEVD-DANSE (pPK-GEVD-DANSE) algorithm, then effectively combines the lowest possible communication cost (as low as in the GEVD-DANSE algorithm) with a lowest possible computational complexity in each node (further reduced from the PK-GEVD-DANSE computational complexity), within the class of algorithms considered in this paper.

VII. Practical considerations

Up until now it has been assumed that the loudspeaker signals are stationary and so always active, which allows to estimate the speech correlation matrix $\bar{\mathbf{R}}_{\mathbf{ss}}$ based on (21) or (28). In practice this assumption is not a valid one when the loudspeakers play back speech or music signals. Therefore the VAD should be able to detect the activity of the desired speech signal in the presence of loudspeaker signals which may also contain speech signals, and other background noise signals. A cascade approach then allows to consider two aspects in the AEC stage. Firstly, different adaptive filtering algorithms can be used such as NLMS, RLS, etc., to estimate the AEC filters. Secondly, the activity of the loudspeaker signals can be monitored and then used to control the adaptive filters. This section describes adaptive implementations of the pPK-GEVD-DANSE algorithm described in Section VI. An NLMS or QRD-RLS algorithm is used to update $\mathbf{F}_k$ in the AEC stage. Then the procedure to deal with the loudspeaker activity detection is also presented.

A. NLMS-based pPK-GEVD-DANSE

A first adaptive implementation of the pPK-GEVD-DANSE algorithm (Section VI) can be obtained using the NLMS algorithm as [37]

$$\begin{aligned}\hat{\mathbf{F}}_k(l) =& \hat{\mathbf{F}}_k(l-1) \\ &+ \mu\mathbf{u}_k(l)\left(\left(\check{\mathbf{H}}_k - \check{\mathbf{B}}_k\hat{\mathbf{F}}_k(l-1)\right)^H\check{\mathbf{y}}_k(l)\right)^H,\end{aligned} \tag{53}$$

$$\mu = \frac{\mu_F}{\hat{\Phi}_{\mathbf{u}_k}(l)+\delta}, \tag{54}$$

$$\hat{\Phi}_{\mathbf{u}_k}(l) = \mathbf{u}_k^H(l)\mathbf{u}_k(l), \tag{55}$$

where $\hat{\Phi}_{\mathbf{u}_k}(l)$ is an estimate of the PSD of the local loudspeaker signals, δ is a regularization term and μ_F is the step size. The filter $\hat{\mathbf{F}}_k(l)$ is updated whenever the desired speech signal is not active and the loudspeaker signals are active (i.e., no double-talk).

B. QRD-RLS-based pPK-GEVD-DANSE

Alternatively $\hat{\mathbf{F}}_k$ can be computed using a QRD-RLS algorithm. It is assumed that

$$\underset{L_f\times Pl_k}{\mathbf{U}_k} = \begin{bmatrix}(\check{\mathbf{B}}_k^H\check{\mathbf{y}}_k(l))^H \\ (\check{\mathbf{B}}_k^H\check{\mathbf{y}}_k(l-1))^H \\ \vdots \\ (\check{\mathbf{B}}_k^H\check{\mathbf{y}}_k(l-L_f+1))^H\end{bmatrix} \tag{56}$$

admits a QR-decomposition (QRD)

$$\underset{L_f\times Pl_k}{\mathbf{U}_k} = \mathcal{Q}_k\begin{bmatrix}\mathcal{R}_k \\ \mathbf{0}\end{bmatrix} \tag{57}$$

where $\mathcal{Q}_k$ is an orthogonal matrix, such that $\mathcal{Q}_k^H\mathcal{Q}_k=\mathbf{I}$, $\mathcal{R}_k$ is an upper triangular matrix and $L_f > Pl_k$. The microphone signals are expressed as

$$\mathbf{Y}_k = \begin{bmatrix}(\check{\mathbf{H}}_k^H\check{\mathbf{y}}_k(l))^H \\ (\check{\mathbf{H}}_k^H\check{\mathbf{y}}_k(l-1))^H \\ \vdots \\ (\check{\mathbf{H}}_k^H\check{\mathbf{y}}_k(l-L_f+1)^H\end{bmatrix} \tag{58}$$

and based on this, the following optimization criterion can be defined

$$\begin{aligned}\hat{\mathbf{F}}_k &= \arg\min_{\mathbf{F}_k}\ ||\mathbf{Y}_k - \mathbf{U}_k\mathbf{F}_k||_2^2, \\ &= \arg\min_{\mathbf{F}_k}\ \left|\left|\mathcal{Q}_k^H(\mathbf{Y}_k - \mathbf{U}_k\mathbf{F}_k)\right|\right|_2^2, \\ &= \arg\min_{\mathbf{F}_k}\ \left|\left|\begin{bmatrix}\mathcal{Z}_k \\ *\end{bmatrix} - \begin{bmatrix}\mathcal{R}_k \\ \mathbf{0}\end{bmatrix}\mathbf{F}_k\right|\right|_2^2,\end{aligned} \tag{59}$$

where the rows of $\mathcal{Z}_k$ correspond to the first Pl_k rows of $\mathcal{Q}_k^H\mathbf{Y}_k$. The star $*$ represents entries to the matrix that are not of interest, however it has been shown that these can be used to compute least squares residuals [37]. The solution to (59) is given by

$$\hat{\mathbf{F}}_k = \mathcal{R}_k^{-1}\mathcal{Z}_k \tag{60}$$

which can be computed by backsubstitution. A QRD-updating can be used in frame l if $\mathcal{R}_k$ is available from frame $l-1$, by means of the following recursions

$$\begin{bmatrix}\mathcal{R}_k(l) & \mathcal{Z}_k(l) \\ \mathbf{0} & *\end{bmatrix} = \mathcal{G}^H\begin{bmatrix}\gamma\mathcal{R}_k(l-1) & \gamma\mathcal{Z}_k(l-1) \\ (\check{\mathbf{B}}_k^H\check{\mathbf{y}}_k(l))^H & (\check{\mathbf{H}}_k^H\check{\mathbf{y}}_k(l))^H\end{bmatrix}, \tag{61}$$

$$\hat{\mathbf{F}}_k(l) = \mathcal{R}_k(l)^{-1}\mathcal{Z}_k(l). \tag{62}$$

where γ is a forgetting factor and $\mathcal{G}$ is an orthogonal transformation. The matrix on the right hand side of (61) is retriangularized via a series of complex Givens rotations [38]. The new quantities are then used to obtain $\hat{\mathbf{F}}_k(l)$ (cfr. (60)).

A complete description of the NLMS-based and QRD-RLS-based pPK-GEVD-DANSE algorithm using simultaneous updating is shown as Algorithm 1 and 2, respectively. The fusion vectors in line 12 and 13 in Algorithm 1 and 2 respectively, can be updated once every D frames, using the previous and current values, and a relaxation factor as in [36].

Algorithm 1: NLMS based pPK-GEVD-DANSE

Construct $\check{\mathbf{H}}_k$ and $\check{\mathbf{B}}_k$ based on m_k, l_k and K (cfr. (51));
Randomly initialize fusion vector $\hat{\mathbf{p}}_k$ $\forall k \in \mathcal{K}$;
Each node $k \in \mathcal{K}$ performs the following simultaneously;
for $l = 1, 2, 3, ...$ **do**
  Collect observations $\mathbf{y}_k(l)$
  Compute $z_k(l)$ (cfr. (39)) and broadcast to other nodes;
  Construct $\check{\mathbf{y}}_k(l)$ based on $\mathbf{z}_{-k}(l)$;
  Update $\hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}$ or $\hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}$ similar to (20) based on VAD;
  Compute $\Phi_{u_k}(l)$ and update $\hat{\mathbf{F}}_k(l)$, following (55) and (53), respectively;
  Update its local LCMV beamformer $\hat{\mathbf{C}}_k$ using (45) and estimate $\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}$ and $\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}$;
  Update $\hat{\mathbf{w}}_k$ using (48) by means of the GEVD of $\{\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}, \hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}\}$;
  Compute fusion vector $\hat{\mathbf{p}}_k$ following (42);
  Compute estimated node-specific desired signal $\hat{d}_k(l) = \hat{\mathbf{w}}_k^H \check{\mathbf{y}}_k(l)$;

Algorithm 2: QRD-RLS based pPK-GEVD-DANSE

Construct $\check{\mathbf{H}}_k$ and $\check{\mathbf{B}}_k$ based on m_k, l_k and K (cfr. (51));
Randomly initialize fusion vector $\hat{\mathbf{p}}_k$ $\forall k \in \mathcal{K}$;
Initialize the matrices $\mathcal{R}_k = 10^{-6}\mathbf{I}_{Pl_k}$ and $\mathcal{Z}_k = \mathbf{0}_{Pl_k \times (m_k+K-1)}$;
Each node $k \in \mathcal{K}$ performs the following simultaneously;
for $l = 1, 2, 3, ...$ **do**
  Collect observations $\mathbf{y}_k(l)$;
  Compute $z_k(l)$ (cfr. (39)) and broadcast to other nodes;
  Construct $\check{\mathbf{y}}_k(l)$ based on $\mathbf{z}_{-k}(l)$;
  Update $\hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}$ or $\hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}$ similar to (20) based on VAD;
  Update $\mathcal{R}_k(l)$ and $\mathcal{Z}_k(l)$ based on (61) and compute $\hat{\mathbf{F}}_k(l)$ according to (62);
  Update its local LCMV beamformer $\hat{\mathbf{C}}_k$ using (45) and estimate $\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}$ and $\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}$;
  Update $\hat{\mathbf{w}}_k$ using (48) by means of the GEVD of $\{\hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}, \hat{\mathbf{R}}^{\text{red}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}\}$;
  Compute fusion vector $\hat{\mathbf{p}}_k$ following (42);
  Compute estimated node-specific desired signal $\hat{d}_k(l) = \hat{\mathbf{w}}_k^H \check{\mathbf{y}}_k(l)$;

C. Loudspeaker activity detection

The loudspeaker signals activity at each node directly affects the adaptation and convergence of the adaptive filters in the AEC stage. For this, a VAD to monitor the activity of the local loudspeaker signals is required to control the updating of the *noise-only* correlation matrix and therefore the adaptation of $\hat{\mathbf{F}}_k$. Using two binary VADs, i.e., VAD_s for the desired speech signal and VAD_p for the local loudspeaker signals, brings four possible combined outcomes for which the operations are defined in the following sections.

1) $\text{VAD}_s = 1$ and $\text{VAD}_p = 0$: The desired speech signal is active and the local loudspeaker signals are inactive. Update *speech-plus-noise* correlation matrix. Do not update $\hat{\mathbf{F}}_k$. This stage allows to collect information about the desired speech signal in the correlation matrix.

2) $\text{VAD}_s = 0$ and $\text{VAD}_p = 1$: The desired speech signal is inactive and the local loudspeaker signals are active. Update *noise-only* correlation matrix and update $\hat{\mathbf{F}}_k$. This is similar to what a standard AEC filter does, where the information about the loudspeaker signals is used to update the AEC filters when the desired speech signal is not active.

3) $\text{VAD}_s = 0$ and $\text{VAD}_p = 0$: Both the desired speech signal and local loudspeaker signals are inactive. Update *noise-only* correlation matrix and do not update $\hat{\mathbf{F}}_k$. This updates the background noise component in the *noise-only* correlation matrix, and given the absence of loudspeaker signals activity, no adaptation is performed of the AEC filters.

4) $\text{VAD}_s = 1$ and $\text{VAD}_p = 1$: Both the desired speech signal and local loudspeaker signals are active (double-talk). Update *speech-plus-noise* correlation matrix and do not update $\hat{\mathbf{F}}_k$. The effects of double-talk in the filter adaptation have been well documented in the literature, where it is shown that the filter adaptation can suffer from this, and even diverge.

D. Computational complexity and communication cost

In each iteration the proposed NLMS-pPK-GEVD-DANSE and QRD-RLS-pPK-GEVD-DANSE algorithm transmit B and receive $(K-1)B$ complex scalars per node, where B is the number of observations. The total communication cost per node in each iteration will then be $2BK$ and for the whole network will be $2BK^2$. The original PK-GEVD-DANSE algorithm transmits $2B$ and receives $2(K-1)B$ complex scalars per node, which means that the communication cost per node is reduced by half in the proposed method. If every node in the WASAN acts as fusion center, the total communication cost for node k becomes $n_k BK$, and NBK for the whole network. The proposed NLMS-pPK-GEVD-DANSE and QRD-RLS-pPK-GEVD-DANSE algorithm reduce the total communication cost at node k if $N > 2$. The centralized PK-MWF algorithm has a computational complexity in terms of complex multiplications of $O(M^3)$ for the GEVD computation and $O\left((PL)^3\right)$ for the matrix inversion in (36), whereas the original PK-GEVD-DANSE algorithm has a computational complexity of $O\left({m'_k}^3\right)$ and $O\left((Pl_k + K - 1)^3\right)$, respectively, with $m'_k = m_k + K - 1$. By using an NLMS or QRD-RLS algorithm in the AEC stage, the computational complexity can be reduced to ${m'_k}^3 + Pl_k n'_k m'_k$

and ${m'_k}^3+\frac{11}{2}(Pl_k)^2+Pl_k(5m'_k+n'_k+\frac{7}{2})+m'_k n'_k$, respectively, in each frequency bin, with $n'_k = n_k + K - 1$.

Table I shows the computational complexity and communication cost of the discussed algorithms. The computational complexity is detailed in terms of the number of complex multiplications. Assuming an scenario as follows $P = 8$, $K = 3$, $l_k = 2$, $m_k = 4$, $m'_k = 6$, $n'_k = 22$, $B = 1024$ and a sampling frequency $F_s = 16000$, the number of floating-point operations per second (FLOPS) in a node for the proposed and the PK-GEVD-DANSE algorithm, using the expressions provided in Table I, are shown in Table II

VIII. Simulations

A. Simulation scenario

In order to assess the performance of the proposed algorithms under different situations, the scenarios shown in Fig. 1 are considered. The room size in both scenarios was set to $5\,\mathrm{m} \times 5\,\mathrm{m} \times 3\,\mathrm{m}$. For the scenario in Fig. 1a the impulse responses were $512\,\mathrm{samples}$ long with a reverberation time of $0.11\,\mathrm{s}$. For the scenario in Fig. 1b the impulse responses were $4096\,\mathrm{samples}$ long with a reverberation time of $1.1\,\mathrm{s}$. For both scenarios the sampling rate was $16\,\mathrm{kHz}$. Further details on the simulation parameters will be presented in Section VIII-B. The simulations aim at representing situations that may occur in real life, such as more than two nodes in the WASAN, stereo echo cancellation, long echo paths and reverberation times, double-talk and highly correlated signals being reproduced in two or more nodes. The simulations are grouped in batch processing, per-frame processing and adaptive processing simulations.

B. Batch-processing

This section outlines the batch simulations carried out using the MWF and PK-MWF algorithm presented in Sections III and IV respectively and the iterative GEVD-DANSE and pPK-GEVD-DANSE algorithm described in Sections V and VI, respectively. First, a comparison of these algorithms using the normalized mean squared error (NMSE) metric is shown in Fig. 2 for a WASAN consisting of 4 nodes, with $\{6, 3, 8, 3\}$ microphones and $\{1, 3, 2, 3\}$ loudspeakers respectively. For comparison, the NMSE that each node would achieve working in isolation using the PK-MWF algorithm is also shown. The desired, loudspeaker and noise signals were uncorrelated white noise signals. The loudspeaker and noise signals were continuously active while the desired signal had an ON/OFF behaviour. The NMSE was computed in the discrete time domain as

$$\text{NMSE} = 10\log_{10}\left(\frac{1}{T}\sum_{t=1}^{T}\frac{(d_k(t)-\hat{d}_k(t))^2}{(d_k(t))^2}\right) \quad \text{dB} \qquad (63)$$

where T is the sample duration of the signal and t the discrete time index. In frames where the desired signal $d_k(t)$ was zero, the power from previous frames was used for the normalization. It is observed in Fig. 2 that including the PK in both the centralized and distributed implementations reduces the NMSE in the estimation of the desired signal. In all nodes the distributed implementation outperforms each node working in isolation. This scenario deals with a large number of microphone and loudspeaker signals, and the activity patterns of the latter may not be very realistic. The scenario depicted

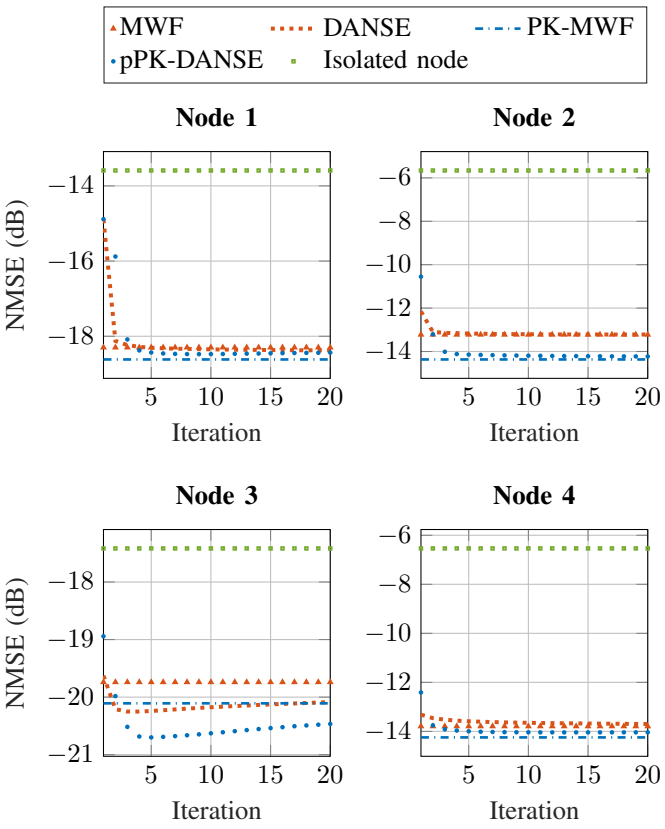

Fig. 2: NMSE for the MWF, GEVD-DANSE, PK-MWF and pPK-GEVD-DANSE algorithms at each node for a WASAN with 4 nodes, with $\{6, 3, 8, 3\}$ microphones signals and $\{1, 3, 2, 3\}$ loudspeakers, respectively.

in Fig. 1a is now analyzed with no interframe filtering, i.e., $P = 1$. The performance of the algorithm was measured in terms of the echo return loss enhancement (ERLE), the signal-to-noise ratio (SNR) and the NMSE. The simulations were set up as follows. Firstly, the microphone and loudspeaker signals were simulated at each node using room impulse responses of 500 samples long simulated with the randomized image method described in [39] and a sampling frequency of $16\,\mathrm{kHz}$. The reflection coefficient of all surfaces in the room was set to 0.15 (for a reverberation time $T_{60} = 0.11\,\mathrm{s}$), and the random displacement of the image sources to $0.13\,\mathrm{m}$. The inter-microphone distance of the arrays was set to $20\,\mathrm{cm}$ for all the nodes. The microphone signals were created such that the signal-to-echo-and-noise ratio at microphone 1 in node 2 was $-5\,\mathrm{dB}$. Then, the corresponding vector $\mathbf{y}_k$ for each node was transformed to the STFT domain using a square-root Hann window of 512 samples using $50\ \%$ overlap. The correlation matrices in (22), (31) and $\{\hat{\mathbf{R}}_{\check{\mathbf{y}}_k\check{\mathbf{y}}_k}, \hat{\mathbf{R}}_{\check{\mathbf{n}}_k\check{\mathbf{n}}_k}\}$ in Section VI were computed by selecting the time frames where the desired speech signal was active and not active, respectively, based on an ideal VAD. An ideal VAD was used to isolate the influence of VAD errors. In practice VAD information may be

TABLE I: Computational complexity and communication cost at node k.

Category	Algorithm	Transmitted signals	Received signals	Computational complexity
Isolated node	PK-MWF	N/A	N/A	$O(Bm_k^3 + B(Pl_k)^3)$
Centralized - Integrated	MWF	$n_k B$	$(N-n_k)B$	$O(BN^3)$
Centralized - Cascade	PK-MWF	$n_k B$	$(N-n_k)B$	$O(BM^3 + B(PL)^3)$
Distributed - Integrated	GEVD-DANSE	B	$(K-1)B$	$O(Bn_k'^3)$
Distributed - Cascade	PK-GEVD-DANSE	$2B$	$2(K-1)B$	$O\left(Bm_k'^3 + B(Pl_k + K - 1)^3\right)$
Distributed - Cascade	NLMS-pPK-GEVD-DANSE	B	$(K-1)B$	$O\left(Bm_k'^3 + BPl_k n_k' m_k'\right)$
Distributed - Cascade	QRD-RLS-pPK-GEVD-DANSE	B	$(K-1)B$	$O\left(Bm_k'^3 + B\left(\frac{11}{2}(Pl_k)^2 + Pl_k(5m_k' + n_k' + \frac{7}{2}) + m_k' n_k'\right)\right)$

TABLE II: GigaFLOPS in a node using each distributed algorithm.

Algorithm	GigaFLOPS
PK-GEVD-DANSE	0.1935
NLMS-pPK-GEVD-DANSE	0.0745
QRD-RLS-pPK-GEVD-DANSE	0.0846

shared among the nodes [40], using a speaker-selective VAD [41] and/or estimating the speech presence probability in a distributed fashion [42]. All nodes in Fig. 1a had a loudspeaker reproducing a speech signal, which were simultaneously active only when the desired speech signal was not active. The second loudspeaker in node 3 was reproducing a music signal which was continuously active. The desired speech signal was produced by a speaker located in the centre of the room. A continuously active localized noise source was also included, producing babble noise.

pPK-GEVD-DANSE was run with simultaneous node updating with a relaxation factor $\alpha_{rS} = 0.9$, to guarantee convergence as suggested in [33]. The ERLE was computed with non-overlapping windows of 1024 samples. The average ERLE (over the time frames) and SNR are shown in Fig. 3. Both metrics were computed for the first microphone in each node. The SNR was computed by filtering the noise component at each microphone signal with the filter obtained for each implementation. The ERLE and SNR when each node works in isolation (ISO) are also shown. pPK-GEVD-DANSE is abbreviated to pPK-DANSE in the legends for brevity. The NMSE for the three algorithms at the first microphone of each node is shown in Fig. 4.

It can be seen in all nodes that including the PK reduces the error in the estimation of the desired speech signal. In node 3 pPK-GEVD-DANSE and PK-MWF outperform MWF in terms of ERLE and SNR. Notice that node 3 is the furthest away from the desired speech source location, it is very close to the noise source location and has two different loudspeaker signals. pPK-GEVD-DANSE performs better, in all nodes, in terms of ERLE and SNR than a node working in isolation. The scenario depicted in Fig. 1b was simulated with echo paths of 4096 samples long and a $T_{60} \approx 1.1$ s. The use of previous frames is investigated for $P = 2$, 4, 8 and 16 in the same scenario. Fig. 5 shows the NMSE for pPK-GEVD-DANSE with different values of P and the PK-MWF, with frame size of 512 samples. It is observed that for all P, except $P = 64$, pPK-GEVD-DANSE outperforms PK-MWF ($P = 1$) at node 2, whereas in node 1 this happens for $P > 2$. In node 2, for

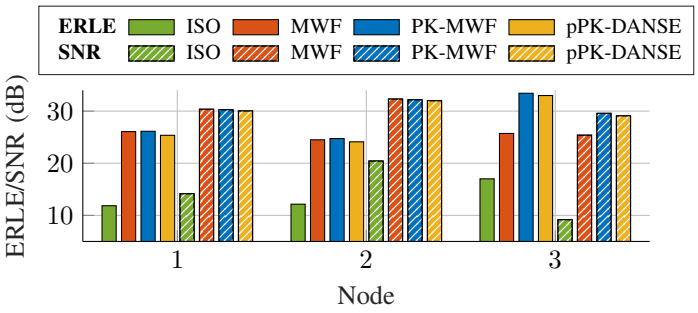

Fig. 3: Average ERLE and SNR computed at the first microphone of each node in Fig. 1a. The ERLE and SNR when the nodes work in isolation (ISO) are also shown.

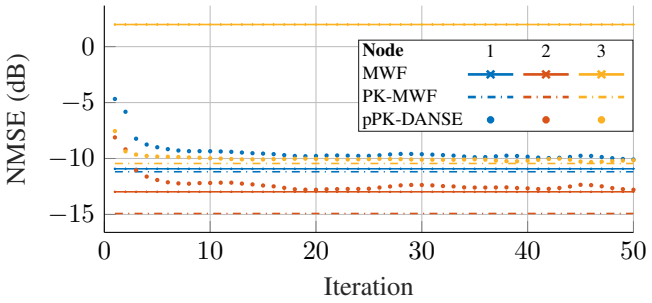

Fig. 4: NMSE at each of the nodes in Fig. 1a.

$P > 16$ the NMSE was not reduced further, which could be related to the presence of the babble noise and the increasing number of filter coefficients to be estimated. Figure 6 shows the NMSE for pPK-GEVD-DANSE with different values of P and the PK-MWF, with frame size of 1024 samples. A similar behaviour is observed where the best result is obtained with a filter which effectively has the same number of taps (i.e. $1024 \times \frac{P}{2}$) as the simulated echo path i.e. 4096.

Additionally, Table III and IV show the short-time objective intelligibility (STOI) [43] and perceptual evaluation of speech quality (PESQ) [44] metrics computed for the ISO, MWF, PK-MWF and pPK-DANSE algorithm for the scenarios depicted in Fig 1. For both metrics pPK-DANSE outperforms each node working in isolation (ISO), and PK-MWF outperforms MWF. In terms of STOI the score differences between MWF, PK-MWF and pPK-DANSE are not significant for both scenarios, except for Node 3 in scenario 1a, where the PK-MWF and pPK-DANSE outperform MWF. In terms of PESQ the differences in scores are larger, with pPK-DANSE outperforming

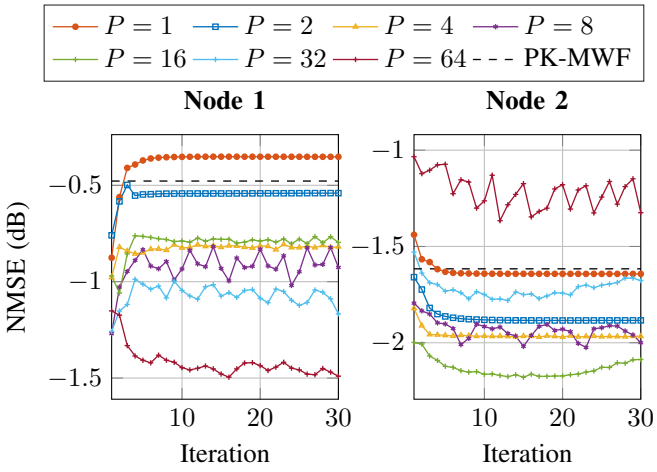

Fig. 5: NMSE for pPK-GEVD-DANSE at each node in Fig. 1b with $P = 1, 2, 4, 8, 16$ and a frame size of 512 samples.

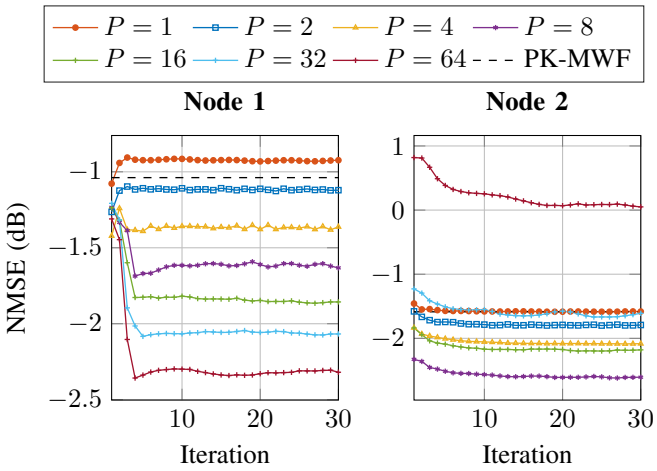

Fig. 6: NMSE for pPK-GEVD-DANSE at each node in Fig. 1b with $P = 1, 2, 4, 8, 16$ and a frame size of 1024 samples.

MWF in Node 3 in scenario 1a and in both nodes in scenario 1b. In all cases, PK-MWF has the highest score.

C. Per-frame processing

A per-frame processing approach is now used for pPK-GEVD-DANSE, where the correlation matrices are updated based on (20), and its NMSE is shown in Fig. 7 and compared to batch results for MWF and PK-MWF. The high NMSE values before frame 50 are due to the initial updating of the

TABLE III: STOI for all batch algorithms.

Node	ISO	MWF	PK-MWF	pPK-DANSE
Scenario 1a				
Node 1	0.70	0.87	0.90	0.86
Node 2	0.69	0.88	0.92	0.87
Node 3	0.68	0.79	0.89	0.87
Scenario 1b				
Node 1	0.32	0.46	0.49	0.49
Node 2	0.18	0.44	0.50	0.49

TABLE IV: PESQ for all batch algorithms.

Node	ISO	MWF	PK-MWF	pPK-DANSE
Scenario 1a				
Node 1	1.34	2.02	2.15	1.91
Node 2	1.34	2.11	2.33	2.03
Node 3	1.32	1.51	2.08	1.98
Scenario 1b				
Node 1	1.26	1.20	1.37	1.33
Node 2	1.10	1.17	1.33	1.32

correlation matrices. The closed-form expression for $\mathbf{F}_k$ was used and regularization was applied before inverting the matrix in (46). The forgetting factor for the correlation matrices was chosen such that data from 6 s ago is weighted with a factor of 0.1, based on the following expression

$$\beta = \exp\left(\frac{\ln(0.1)}{6\frac{F_s}{R/2}}\right) \tag{64}$$

where F_s is the sampling frequency and R the frame size. A per-frame approach is also used for $P > 1$ and it is compared to the batch solution with the same frame size in Fig. 8.

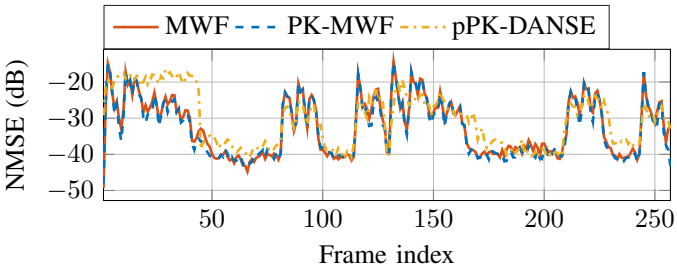

Fig. 7: NMSE at node 2 in Fig. 1b. Frame size of 4096 samples.

D. Adaptive processing

The adaptive simulations presented in this section show the results for the algorithms described in Sections VII-A and VII-B. The results are shown in terms of the NMSE for each frame index. The NMSE for the NLMS-based PK-MWF and pPK-GEVD-DANSE are shown in Fig. 9 and 10, respectively. The scenario used is shown in Fig. 1b. The first frames are used to update the *speech-plus-noise* and *noise-only* correlation matrices. The pPK-GEVD-DANSE takes a longer time to reduce the NMSE initially (around frame index 50) but after that it reaches lower values than the centralized implementation. It should be noted that the fusion vectors are computed and modified more often than in a batch-implementation which leads to some incorrect entries in the correlation matrices. However once they reach a stable point (around frame index 175), the algorithm performs similarly to the centralized algorithm.

The QRD-based PK-MWF and pPK-GEVD-DANSE are shown in Fig. 11 and 12 using the scenario in Fig. 1b. The upper triangular matrix was initialized with a soft-constrained initialization factor as $\delta\mathbf{I}$, where $\mathbf{I}$ is an identity matrix with matching dimensions, according to the implementation. The scenario is time-invariant, i.e., the impulse responses do not

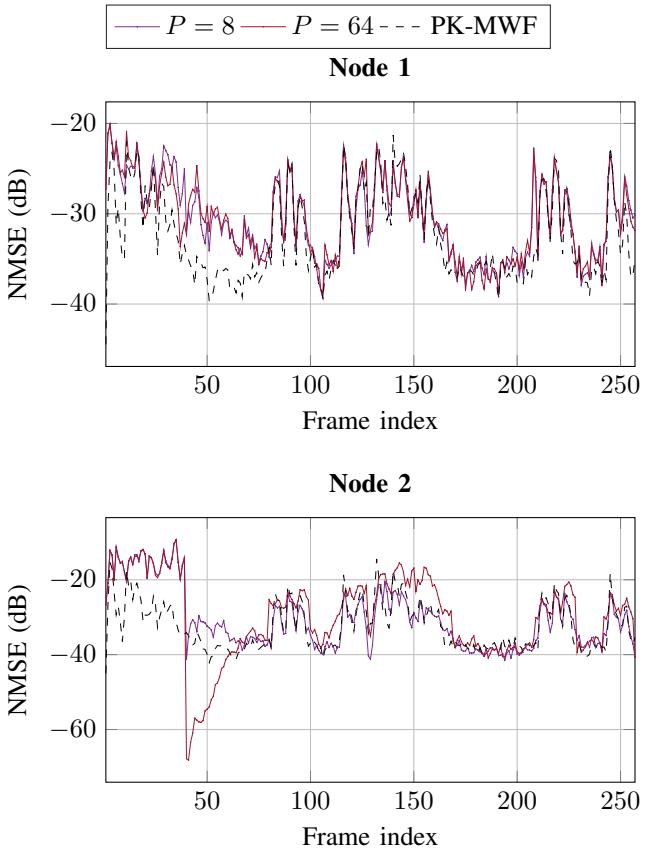

Fig. 8: NMSE for pPK-GEVD-DANSE at each of the nodes in Fig. 1b with $P = 8, 64$ and a frame size of 512 samples. The PK-MWF batch solution is shown as reference with same frame size

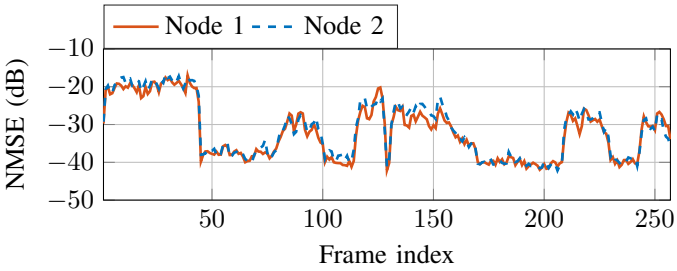

Fig. 9: NMSE at node 1 and 2 using the NLMS-based PK-MWF in scenario depicted in Fig. 1b using a step size $\mu_F = 0.02$, $\delta = 0.0154$ (the sum of the power of the loudspeaker signals) and a frame size of 4096 samples.

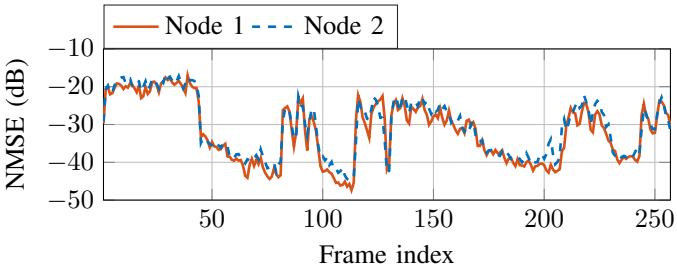

Fig. 10: NMSE at node 1 and 2 using the NLMS-based pPK-GEVD-DANSE in scenario depicted in Fig. 1b using a step size $\mu_F = 0.002$, $\delta = 0.0154$ and a frame size of 4096 samples.

change with time, hence the forgetting factor was set to 1 for the PK-MWF. For pPK-GEVD-DANSE, the forgetting factor was set to 0.97 due to the fact that the fusion vector in each node is updated over time. Similar results to the NLMS-based PK-MWF are observed for the QRD-based PK-MWF. The results for the QRD-based pPK-GEVD-DANSE algorithm are more stable than for the NLMS-based pPK-GEVD-DANSE. The NMSE values are more consistent and slightly lower in the desired speech signal parts than for the NLMS-based pPK-GEVD-DANSE. It should be noted that the tuning of the NLMS-based algorithms can take some time, which is not the case with the QRD-based algorithms, where only the forgetting factor and initialization of the upper triangular matrix ($\delta\mathbf{I}$) are needed.

To keep a low algorithmic delay, the frame size was reduced to 512 to estimate the 4096 samples long echo paths in the scenario in Fig. 1b and $P = 8$. The resulting NMSE for both adaptive implementations is shown in Fig. 13. A similar NMSE to that obtained with the respective PK-MWF using a frame size of 4096 is observed.

IX. Conclusion

It has been shown that the GEVD-DANSE algorithm from [24] can be adopted for distributed integrated AEC and NR, and that the PK-GEVD-DANSE algorithm from [27] can be adopted for distributed cascade AEC and NR in a WASAN. In addition, the communication cost of the PK-GEVD-DANSE algorithm has been reduced resulting in each node in the network broadcasting only 1 fused signal (instead of 2 signals) to the other nodes. The resulting algorithm, referred to as the pruned PK-GEVD-DANSE (pPK-GEVD-DANSE) algorithm, then effectively combines the lowest possible communication cost (as low as in the GEVD-DANSE algorithm) with a lowest possible computational complexity in each node (further reduced from the PK-GEVD-DANSE computational complexity), within the class of algorithms considered in this paper. The performance of the algorithms has been verified in terms of AEC quantified with the ERLE, as well as in terms of NR quantified with the SNR. As in most cascaded approaches, it has been shown that the AEC stage can be implemented using an NLMS- or QRD-RLS-based algorithm.

References

[1] W. Herbordt, W. Kellermann, and S. Nakamura, "Joint optimization of acoustic echo cancellation and adaptive beamforming," in *Topics in Acoustic Echo and Noise Control*, E. Hänsler and G. Schmidt, Eds. Springer, 2006, pp. 19–50.

[2] J. Benesty, J. R. Jensen, M. G. Christensen, and J. Chen, *Speech Enhancement: A Signal Subspace Perspective*. Elsevier, 2014.

[3] E. Böhmler, J. Freudenberger, and S. Stenzel, "Combined echo and noise reduction for distributed microphones," in *Proc. 2011 Joint Workshop Hands-Free Speech Commun. and Microphone Arrays (HSCMA '11)*. Edinburgh, United Kingdom, 2011, pp. 98–103.

[4] S. Gustafsson, R. Martin, and P. Vary, "Combined acoustic echo control and noise reduction for hands-free telephony," *Signal Process.*, vol. 64, no. 1, pp. 21–32, 1998.

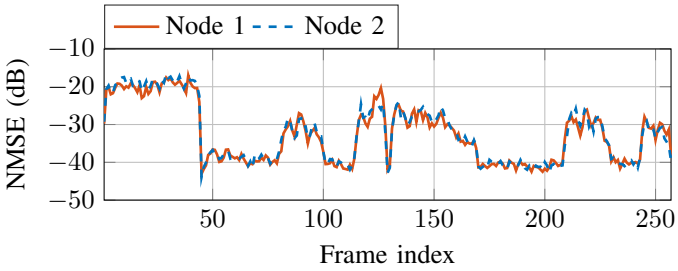

Fig. 11: NMSE at node 1 and 2 using the QRD-based PK-MWF in scenario depicted in Fig. 1b with forgetting factor $\gamma = 1$.

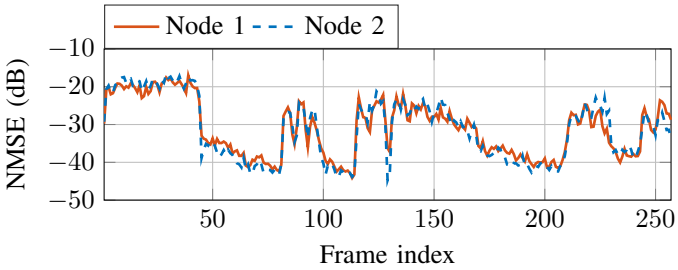

Fig. 12: NMSE at node 1 and 2 using the QRD-based pPK-GEVD-DANSE in scenario depicted in Fig. 1b with forgetting factor $\gamma = 0.97$.

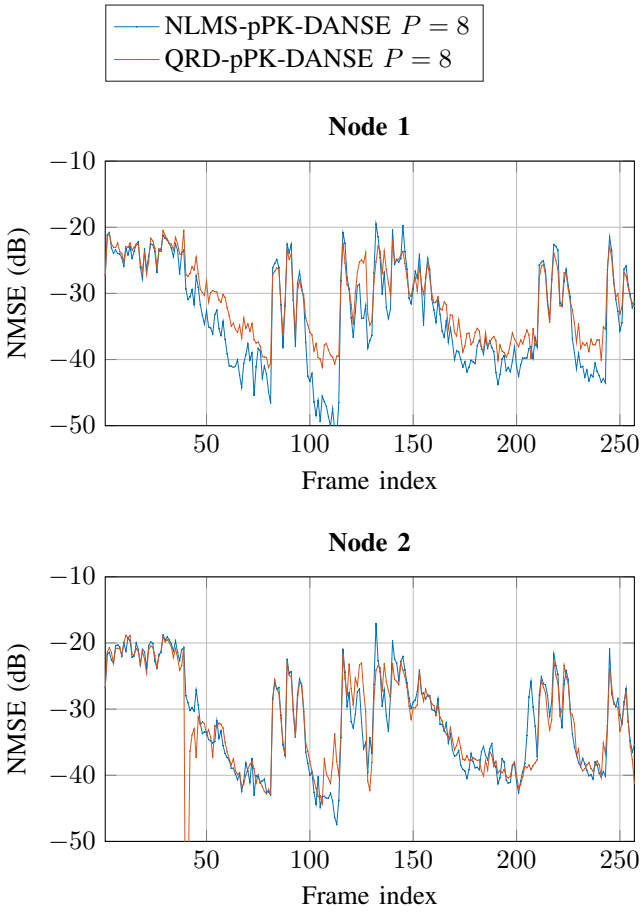

Fig. 13: NMSE for the NLMS-based and QRD-based pPK-GEVD-DANSE algorithms at each of the nodes in Fig. 1b with $P = 8$ and a frame size of 512 samples. For the NLMS-based algorithm the step size $\mu_F = 0.001$, $\delta = 10^{-2}$. For the QRD-based algorithm $\gamma = 0.996$.

[5] M. M. Sondhi, "An adaptive echo canceller," *Bell Syst. Tech. J.*, vol. 46, no. 3, pp. 497–511, 1967.

[6] M. M. Sondhi, "The history of echo cancellation," *IEEE Signal Process. Mag.*, vol. 23, no. 5, pp. 95–102, 2006.

[7] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, S. L. Gay *et al.*, *Advances in Network and Acoustic Echo Cancellation*. Springer, 2001.

[8] W. Kellermann, "Strategies for combining acoustic echo cancellation and adaptive beamforming microphone arrays," in *Proc. 1997 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '97)*, 1997, pp. 219–222 vol.1.

[9] G. Enzner and P. Vary, "Frequency-domain adaptive Kalman filter for acoustic echo control in hands-free telephones," *Signal Process.*, vol. 86, no. 6, pp. 1140–1156, 2006.

[10] K. Sridhar, R. Cutler, A. Saabas, T. Parnamaa, M. Loide, H. Gamper, S. Braun, R. Aichner, and S. Srinivasan, "ICASSP 2021 acoustic echo cancellation challenge: Datasets, testing framework, and results," in *Proc. 2021 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '21)*, 2021, pp. 151–155.

[11] A. Fazel, M. El-Khamy, and J. Lee, "CAD-AEC: Context-aware deep acoustic echo cancellation," in *Proc. 2020 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '20)*. Barcelona, Spain (virtual conference), 2020, pp. 6919–6923.

[12] G. Rombouts and M. Moonen, "An integrated approach to acoustic noise and echo cancellation," *Signal Process.*, vol. 85, no. 4, pp. 849–871, 2005.

[13] W. Herbordt and W. Kellermann, "Frequency-domain integration of acoustic echo cancellation and a generalized sidelobe canceller with improved robustness," *European Trans. Telecommun.*, vol. 13, no. 2, pp. 123–132, 2002.

[14] S. J. Park, C. G. Cho, C. Lee, and D. H. Youn, "Integrated echo and noise canceler for hands-free applications," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 49, no. 3, pp. 188–195, 2002.

[15] A. Cohen, A. Barnov, S. Markovich-Golan, and P. Kroon, "Joint beamforming and echo cancellation combining QRD based multichannel AEC and MVDR for reducing noise and non-linear echo," in *Proc. 26th European Signal Process. Conf. (EUSIPCO '18)*. Rome, Italy, 2018, pp. 6–10.

[16] W. Kellermann, "Joint design of acoustic echo cancellation and adaptive beamforming for microphone arrays," in *Proc. 1997 Int. Workshop Acoustic Echo Noise Control (IWAENC '97)*, 1997, pp. 81–84.

[17] P. Meyer, S. Elshamy, and T. Fingscheidt, "A multichannel Kalman-based Wiener filter approach for speaker interference reduction in meetings," in *Proc. 2020 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '20)*. Barcelona, Spain (virtual conference), 2020, pp. 451–455.

[18] K. Nathwani, "Joint acoustic echo and noise cancellation using spectral domain Kalman filtering in double-talk scenario," in *Proc. 2018 Int. Workshop Acoustic Signal Enhancement (IWAENC '18)*. Tokyo, Japan, 2018, pp. 1–330.

[19] H. Zhang, K. Tan, and D. Wang, "Deep learning for joint acoustic echo and noise cancellation with nonlinear distortions." in *Proc. Interspeech '19*. Graz, Austria, 2019, pp. 4255–4259.

[20] H. Seo, M. Lee, and J.-H. Chang, "Integrated acoustic echo and background noise suppression based on stacked deep neural networks," *Appl. Acust.*, vol. 133, pp. 194–201, 2018.

[21] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer Networks*, vol. 51, no. 4, pp. 921–960, 2007.

[22] Y. Zeng and R. C. Hendriks, "Distributed delay and sum beamformer for speech enhancement via randomized gossip," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 22, no. 1, 2013.

[23] R. Heusdens, G. Zhang, R. C. Hendriks, Y. Zeng, and W. B. Kleijn, "Distributed MVDR beamforming for (wireless) microphone networks using message passing," in *Proc. 2012 Int. Workshop Acoustic Signal Enhancement (IWAENC '12)*. Aachen, Germany, 2012.

[24] 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.

[25] E. Ceolini and S. C. Liu, "Combining deep neural networks and beamforming for real-time multi-channel speech enhancement using a wireless acoustic sensor network," in *Proc. 2019 Int. Workshop Mach. Learning Signal Process. (MLSP '19)*. Pittsburgh, PA, USA, 2019, pp. 1–6.

[26] S. Ruiz, T. van Waterschoot, and M. Moonen, "Distributed combined acoustic echo cancellation and noise reduction using GEVD-based distributed adaptive node specific signal estimation with prior knowledge," in *Proc. 28th European Signal Process. Conf. (EUSIPCO '20)*. Amsterdam, The Netherlands, 2021.

[27] R. Van Rompaey and M. Moonen, "Distributed adaptive node-specific signal estimation in a wireless sensor network with partial prior knowledge of the desired source steering vector," in *Proc. 27th European Signal Process. Conf. (EUSIPCO '19)*. A Coruña, Spain, 2019.

[28] M. H. Bahari, A. Bertrand, and M. Moonen, "Blind sampling rate offset estimation for wireless acoustic sensor networks through weighted least-squares coherence drift estimation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 25, no. 3, pp. 674–686, 2017.

[29] S. Markovich-Golan, S. Gannot, and I. Cohen, "Blind sampling rate offset estimation and compensation in wireless acoustic sensor networks with application to beamforming," in *Proc. 2012 Int. Workshop Acoustic Signal Enhancement (IWAENC '12)*. VDE, 2012, pp. 1–4.

[30] A. Chinaev, P. Thüne, and G. Enzner, "A double-cross-correlation processor for blind sampling rate offset estimation in acoustic sensor networks," in *Proc. 2019 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '19)*. IEEE, 2019, pp. 641–645.

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

[32] J. Benesty, J. Chen, Y. A. Huang, and S. Doclo, "Study of the Wiener filter for noise reduction," in *Speech Enhancement*, J. Benesty, S. Makino, and J. Chen, Eds. Springer, 2005, pp. 9–41.

[33] A. Bertrand and M. Moonen, "Robust distributed noise reduction in hearing aids with external acoustic sensor nodes," *EURASIP J. Adv. Signal Process.*, vol. 2009, p. 12, 2009.

[34] F. Jabloun and B. Champagne, "Signal subspace techniques for speech enhancement," in *Speech Enhancement*, J. Benesty, S. Makino, and J. Chen, Eds. Springer, 2005, pp. 135–159.

[35] S. Doclo, S. Gannot, M. Moonen, and A. Spriet, "Acoustic beamforming for hearing aid applications," in *Handbook on Array Processing and Sensor Networks*, S. Haykin and K. R. Liu, Eds. Wiley Online Library, 2010, pp. 269–302.

[36] J. Szurley, A. Bertrand, and M. Moonen, "Improved tracking performance for distributed node-specific signal enhancement in wireless acoustic sensor networks," in *Proc. 2013 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '13)*. Vancouver, Canada, 2013, pp. 336–340.

[37] S. Haykin, *Adaptive Filter Theory*. Prentice-Hall, Inc., 1996.

[38] G. Rombouts and M. Moonen, "QRD-based unconstrained optimal filtering for acoustic noise reduction," *Signal Process.*, vol. 83, no. 9, pp. 1889–1904, 2003.

[39] E. De Sena, N. Antonello, M. Moonen, and T. van Waterschoot, "On the modeling of rectangular geometries in room acoustic simulations," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 23, no. 4, pp. 774–786, 2015.

[40] V. Berisha, H. Kwon, and A. Spanias, "Real-time implementation of a distributed voice activity detector," in *Proc. 4th IEEE Workshop Sensor Array Multichannel Process. (SAM '06)*. Waltham, MA, USA, 2006, pp. 659–662.

[41] S. Maraboina, D. Kolossa, P. Bora, and R. Orglmeister, "Multi-speaker voice activity detection using ICA and beampattern analysis," in *Proc. 14th European Signal Process. Conf. (EUSIPCO '06)*. Florence, Italy, 2006.

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

[43] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *Proc. 2010 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '10)*. IEEE, 2010, pp. 4214–4217.

[44] ITU-T Rec. P.862, "Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," International Telecommunication Union, Geneva, Switzerland, 2001.