Distributed Weighted Prediction Error for Speech Dereverberation with Regularization by Denoising

Yibo Wang, Ziye Yang, Chengbo Chang, Jie Chen
Research and Development Institute of Northwestern Polytechnical University in Shenzhen, China
CIAIC, School of Marine Science and Technology, Northwestern Polytechnical University, China
{wangyibooo, yzy97, ccb}@mail.nwpu.edu.cn, dr.jie.chen@ieee.org

***Abstract*—Speech dereverberation addresses the degradation of speech quality caused by late reverberation. Although the weighted prediction error (WPE) method has demonstrated superior performance in mitigating reverberation, its centralized architecture results in substantial computational and communication overhead, particularly in distributed settings where each spatially separated node is equipped with a microphone array. This paper first formulates a novel distributed WPE optimization problem that fits into this network scenario. To further enhance the optimization process, we propose to integrate data-driven speech priors into the framework via a plug-and-play strategy. Hence, the proposed framework not only reduces the computation and communication complexity at individual nodes through effective inter-node collaboration but also improves performance under challenging acoustic conditions. Experimental evaluations confirm the framework's effectiveness in both noise-free and noisy distributed scenarios.**

***Index Terms*—Distributed speech dereverberation, the weighted prediction error method, deep speech priors, regularization by denoising**

I. Introduction

Speech signals recorded in enclosed environments are invariably affected by reverberation, which arises from the multiple reflections of sound waves off rigid surfaces [1], [2]. Consequently, despite the inherent difficulties of speech dereverberation, this research area has attracted considerable attention over recent decades [3], [4]. Among the established dereverberation techniques, the weighted prediction error (WPE) method is particularly notable, which employs a centralized strategy by aggregating observations from all microphones on a reference channel to estimate and subtract the late reverberation component from the speech signal [5].

Although the centralized strategy has demonstrated considerable potential, aggregating and processing signals on a reference channel presents substantial computational challenges. This issue is particularly pronounced in distributed scenarios. For example, in modern smart home environments or immersive online conference applications, microphones are often widely dispersed throughout a room, with each device linked to a computational unit that typically has limited processing capacity. To mitigate this issue, recent research has focused on developing the distributed WPE (DWPE) technique [6], allowing each node to operate with its own independent processing unit. In this approach, optimal estimation is achieved through efficient inter-node cooperation by exchanging compressed data rather than raw data. Consequently, the computational burden traditionally centralized at a fusion center is distributed among individual nodes. Despite the advantages of DWPE, its performance under complex acoustic conditions could be further enhanced.

Recently, the plug-and-play (PnP) strategy has garnered considerable attention in the signal processing community [7]–[9], which involves incorporating a deep denoising algorithm as a module within iterative optimization processes, thereby implicitly capturing deep priors learned from data. As such, the PnP strategy enables the seamless integration of physics-based methods with data-driven techniques by leveraging their complementary strengths. Several studies have validated the efficacy of this hybrid strategy in addressing the inverse problems in some speech processing applications [10]–[15].

Motivated by these advances, we propose incorporating data-driven priors into the conventional DWPE framework to address the distributed speech dereverberation task. Specifically, we employ the regularization-by-denoising (RED) strategy [16], a powerful variant of the PnP approach, to integrate a deep neural network (DNN)-based speech denoiser into the reformulated distributed dereverberation problem, thereby facilitating the parameter estimation process. This integration effectively captures the intricate speech priors inherent in the data, ensuring that the output at each node aligns with the structural characteristics typically observed in speech signals. Consequently, our framework reduces the computational and communication complexity at individual nodes through inter-node collaboration while enhancing performance in challenging acoustic environments.

II. System Model and Problem Formulation

A. System and Signal Model

Consider a speech sensing scenario in which an acoustic sensor network is deployed to capture speech signals in a distributed manner. As illustrated in Fig. 1(a), the network comprises M interconnected nodes, each equipped with multiple microphones to facilitate collaborative environmental sensing. Specifically, node i is outfitted with Q_i microphones, and the total number of microphones in the network is given

Y. Wang and Z. Yang contributed equally to this work. Corresponding author: J. Chen. The work was supported in part by Shenzhen Science and Technology Program JCYJ20230807145600001 and TCL science and technology innovation fund.

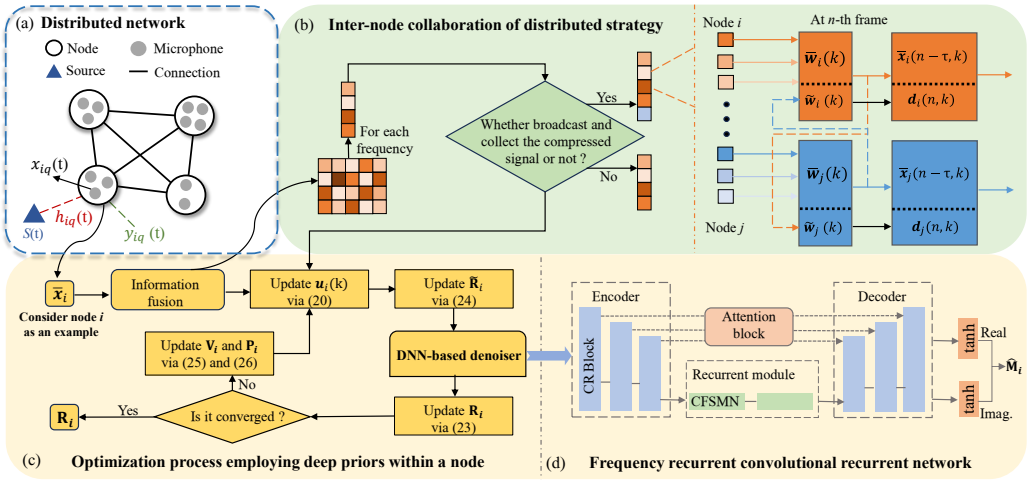

Fig. 1. Overview of the proposed method. (a) Network and signal model. (b) The inter-node collaboration of distributed strategy. (c) The optimization process with PnP to employ deep priors within a node. (d) The architecture of the incorporated DNN.

by $Q = \sum_{i=1}^{M} Q_i$. Note that under our scenario settings, each node has full access to signals captured by its own microphones; however, inter-node communication remains subject to bandwidth constraints.

Let $s(t)$ denote the source speech signal. The q-th microphone at node i receives the signal $x_{iq}(t)$, which can be modeled as:

$$x_{iq}(t) = h_{iq}(t) * s(t) + y_{iq}(t), \tag{1}$$

where $h_{iq}(t)$ represents the acoustic impulse response from the source to the q-th microphone at node i, the symbol $*$ represents the linear convolution operation, and $y_{iq}(t)$ is the zero-mean additive noise at the q-th microphone, assumed to be independent of the source speech signal $s(t)$.

The signal model described in equation (1) can be approximated in the short-time Fourier transform (STFT) domain as follows:

$$X_{iq}(n,k) = \sum_{j=0}^{J-1} H_{iq}(j,k) S(n-j,k) + Y_{iq}(n,k), \tag{2}$$

where n and k represent the time-frame and frequency-bin indices, respectively, and J denotes the convolutive order of $H_{iq}(n,k)$ across time frames. In this STFT domain, $X_{iq}(n,k)$, $S(n,k)$, and $Y_{iq}(n,k)$ represent the STFT counterparts of $x_{iq}(t)$, $s(t)$, and $y_{iq}(t)$, respectively.

B. Centralized WPE with Data-Driven Regularization

Now, consider the case where the signals from all microphones are aggregated in a centralized processing unit. In this configuration, multichannel linear prediction is widely employed for dereverberation, as it estimates the desired speech signal by minimizing the prediction error. Specifically, the approach utilizes signals that commence at a delay τ as regressors in the prediction model.

Define the regressor vector $\bar{\mathbf{x}}_i(n-\tau,k) \in \mathbb{C}^{LQ_i}$ at node i and time instant $n-\tau$ as follows:

$$\bar{\mathbf{x}}_i(n-\tau,k) = \mathrm{col}\left\{ \{X_{iq}(n-\tau-l,k)\}_{l=0}^{L-1} \right\}_{q=1}^{Q_i}, \tag{3}$$

where $\mathrm{col}\{\cdots\}$ represents the operation of stacking its arguments into a column vector, and L denotes the filter order. The desired speech signal at time n and frequency bin k can be estimated as:

$$\hat{S}(n,k) = X_{\mathrm{ref}}(n,k) - \bar{\mathbf{w}}^{\mathrm{H}}(k)\bar{\mathbf{x}}(n-\tau,k), \tag{4}$$

where $X_{\mathrm{ref}}(n,k)$ refers to the reference signal, which can be chosen arbitrarily from any microphone, and $\bar{\mathbf{x}}(n-\tau,k)$ is the aggregated regressor from all nodes in the network, given by:

$$\bar{\mathbf{x}}(n-\tau,k) = \mathrm{col}\left\{ \bar{\mathbf{x}}_1(n-\tau,k), \cdots, \bar{\mathbf{x}}_M(n-\tau,k) \right\}. \tag{5}$$

The filter weight vector $\bar{\mathbf{w}}^{\mathrm{H}}(k) \in \mathbb{C}^{LQ}$ is constructed as:

$$\bar{\mathbf{w}}^{\mathrm{H}}(k) = \mathrm{col}\left\{ \bar{\mathbf{w}}_1, \cdots, \bar{\mathbf{w}}_M \right\}, \tag{6}$$

where $\bar{\mathbf{w}}_i \in \mathbb{C}^{LQ_i}$ is the weight vector associated with the regressor $\bar{\mathbf{x}}_i(n-\tau,k)$ at node i.

The WPE method aims to determine the filter weight vector by minimizing the following cost function:

$$\mathcal{J}_{\mathrm{WPE}}\left(\{\bar{\mathbf{w}}(k)\}_{k=1}^{K} \right) = \sum_{k=1}^{K} \sum_{n=1}^{N} \left[\frac{|\hat{S}(n,k)|^2}{\sigma(n,k)} + \log\left(\pi\sigma(n,k)\right) \right], \tag{7}$$

where $\hat{S}(n,k)$ is defined as in equation (4), $\sigma(n,k)$ represents the estimated speech variance at frame n and frequency bin k, and N is the number of frames used as regressors for prediction.

Since $\hat{S}(n,k)$ is regarded as the desired speech, it is advantageous to incorporate a regularizer to enforce speech priors on $\hat{S}(n,k)$:

$$\mathcal{J}_{\mathrm{WPE\_Reg}}\left(\{\bar{\mathbf{w}}(k)\}_{k=1}^{K} \right) = \mathcal{J}_{\mathrm{WPE}}\left(\{\bar{\mathbf{w}}(k)\}_{k=1}^{K} \right) + \beta_0 \mathcal{J}_{\mathrm{Reg}}\left(\hat{\mathbf{R}} \right), \tag{8}$$

where β_0 is a parameter that balances the trade-off, $\mathcal{J}_{\mathrm{Reg}}$ represents a regularization term, and $\hat{\mathbf{R}} \in \mathbb{R}^{N \times K}$ is the speech

time-frequency matrix with its (n,k)-th element given by $\hat{R}(n,k)$ [1],

$$\hat{R}(n,k) = X_{\text{ref}}(n,k) - \overline{\mathbf{w}}^{\text{H}}(k)\overline{\mathbf{x}}(n-\tau,k) - V(n,k) \quad (9)$$

where $V(n,k)$ represents the additive noise at time frame n and frequency bin k. Without designing an explicit regularizer, we learn priors from speech data and incorporate them into the optimization process, using the regularization by denoising (RED) strategy as follows:

$$\mathcal{J}_{\text{Reg}}\left(\hat{\mathbf{R}}\right) = \frac{1}{2}\langle\hat{\mathbf{R}}, \hat{\mathbf{R}} - \Omega(\hat{\mathbf{R}})\rangle, \quad (10)$$

where $\Omega(\cdot)$ represents a deep-neural network based speech denoiser. This formulation serves as an efficient regularizer, exhibiting beneficial derivative characteristics under reasonable assumptions.

C. Distributed dereverberation

In a distributed configuration, each node i operates in parallel to fulfill the same role. Specifically, each node independently estimates $\overline{\mathbf{w}}_i$ and $\hat{R}(n,k)$ based on its local microphone signals $x_{iq}(t)$ (for $q = 1, \cdots, Q_i$), and exchanges only reduced information with other nodes, without aggregating their raw data.

III. Distributed WPE with data-driven priors

A. Data compression and inter-node information sharing

An effective approach for estimating the required node-specific parameters and signals is to incorporate learnable compressors at each node. In this framework (as illustrated in Fig. 1(b)), each node leverages its local raw data in conjunction with compressed information received from connected nodes to perform prediction. Consequently, the prediction process at node i can be expressed as follows:

$$\hat{S}_i(n,k) = X_{\text{ref}}(n,k) - \left[\overline{\mathbf{w}}_i^{\text{H}}(k) \,|\, \tilde{\mathbf{w}}_i^{\text{H}}(k)\right] \begin{bmatrix} \overline{\mathbf{x}}_i(n-\tau,k) \\ \mathbf{d}_i(n-\tau,k) \end{bmatrix} \quad (11)$$

where $\mathbf{d}_i(n-\tau,k) \in \mathbb{C}^{M-1}$ is a vector of compressed data from other nodes. and $\tilde{\mathbf{w}}_i \in \mathbb{C}^{M-1}$ is the weights associated with the exchanged compressed signal $\mathbf{d}_i(n-\tau,k)$ respectively.

Now, let us specify the construction of the compressed data via a linear projection

$$c_j(n-\tau,k) = \mathbf{g}_j^{\text{H}}(k)\overline{\mathbf{x}}_j(n-\tau,k). \quad (12)$$

Each nodes then broadcasts this compressed information to its neighbors. For each node i, the compressed data vector $\mathbf{d}_i(n-\tau,k)$ is constructed by

$$\mathbf{d}_i(n-\tau,k) = \text{col}\{c_j(n-\tau,k)\}_{j\neq i}. \quad (13)$$

It is worth noting that a particularly effective strategy for determining $\mathbf{g}_j(k)$ is to set

$$\mathbf{g}_j(k) = \overline{\mathbf{w}}_i(k). \quad (14)$$

[1]Similar notation will be applied to other bold capital letters, such as $\mathbf{R}$, $\mathbf{V}$, and $\mathbf{P}$.

Although the optimal $\overline{\mathbf{w}}_i(k)$ is initially unknown, $\mathbf{g}_j(k)$ can be updated using the current estimate of $\overline{\mathbf{w}}_i(k)$ during the iterative process, as will be detailed in subsequent sections. **Remark:** Under the efficient inter-node cooperation scheme, each node i broadcasts only N frames signals to other nodes rather than $Q_i \times N$ frames per frequency bin, thereby substantially reducing the overall communication complexity across the network.

B. Distributed problem formulation on each node

In light of the previously described data compression and sharing mechanism, we relax the centralized problem in (8) to derive a local cost function for distributed dereverberation incorporating data-driven priors:

$$\begin{aligned}\mathcal{J}_i\left(\overline{\mathbf{w}}_i(k), \tilde{\mathbf{w}}_i(k), \boldsymbol{\sigma}_i(k), \hat{\mathbf{R}}_i, \hat{\mathbf{V}}_i\right) &= \sum_{k=1}^{K}\sum_{n=1}^{N}\log\pi\sigma_i(n,k) \\ &+ \frac{1}{\sigma_i(n,k)}\left|X_{\text{ref},i}(n,k) - \left[\overline{\mathbf{w}}_i^{\text{H}}(k)|\tilde{\mathbf{w}}_i^{\text{H}}(k)\right]\begin{bmatrix}\overline{\mathbf{x}}_i(n-\tau,k)\\ \mathbf{d}_i(n-\tau,k)\end{bmatrix}\right|^2 \\ &+ \frac{\beta_0}{2}\langle\hat{\mathbf{R}}_i, \hat{\mathbf{R}}_i - \Omega(\hat{\mathbf{R}}_i)\rangle\end{aligned} \quad (15)$$

with

$$\begin{aligned}\hat{R}_i(n,k) = X_{\text{ref},i}(n,k) - \left[\overline{\mathbf{w}}_i^{\text{H}}(k)|\tilde{\mathbf{w}}_i^{\text{H}}(k)\right]\begin{bmatrix}\overline{\mathbf{x}}_i(n-\tau,k)\\ \mathbf{d}_i(n-\tau,k)\end{bmatrix} \\ - V_i(n,k).\end{aligned} \quad (16)$$

Here, $V(n,k)$ denotes the additive noise in the modeling and processing. In contrast to the centralized formulation, this distributed solution is characterized by three key features: i) the incorporation of compressed information from neighboring nodes; ii) the use of a local reference signal, and iii) the enhancement of speech properties in the local desired speech estimate, $\hat{\mathbf{R}}_i$.

C. Problem solving

For brevity, we define the following composite vectors:

$$\mathbf{u}_i(k) = \text{col}\{\overline{\mathbf{w}}_i(k), \tilde{\mathbf{w}}_i(k)\}, \quad (17)$$

$$\mathbf{y}_i(n-\tau,k) = \text{col}\{\overline{\mathbf{x}}_i(n-\tau,k), \mathbf{d}_i(n-\tau,k)\}. \quad (18)$$

With a given $\boldsymbol{\sigma}_i(k)$, we consider the following (scaled) augmented Lagrange function in order to minimize (15) with the equality constraint (16):

$$\begin{aligned}\mathcal{L}\Big(\mathbf{u}_i(k)_{k=1}^K, \hat{\mathbf{R}}_i, \mathbf{V}_i, \mathbf{P}_i\Big) &= \mathcal{J}_i\left(\overline{\mathbf{w}}_i(k), \tilde{\mathbf{w}}_i(k), \boldsymbol{\sigma}_i(k), \hat{\mathbf{R}}_i, \hat{\mathbf{V}}_i\right) \\ &+ \frac{\rho}{2}\sum_{k=1}^{K}\sum_{n=1}^{N}\Big(\big|[X_{\text{ref},i}(n,k) - \mathbf{u}_i{}^{\text{H}}\mathbf{y}_i(n-\tau,k)] \\ &- V_i(n,k) - \hat{R}_i(n,k) + P_i(n,k)\big|^2 - |P_i(n,k)|^2\Big),\end{aligned} \quad (19)$$

In this context, $P_i(n,k)$ represents the scaled dual variable in i-th node, and ρ denotes the penalty parameter. The ADMM method decomposes the optimization of the problem in equation (19) into solving separate subproblems for each iteration index ℓ as follows.

TABLE I

PERFORMANCE UNDER VARIOUS DISTRIBUTED CONDITIONS, WHERE L REPRESENTS THE FILTER ORDER REQUIRED BY THE DIFFERENT COMPARISON METHODS. THE BEST RESULTS ARE IN BOLD.

Methods	Node 1					Node 2					Node 3				
	L	Noise-free		Noisy (0 dB)		L	Noise-free		Noisy (0 dB)		L	Noise-free		Noisy (0 dB)	
		PESQ ↑	STOI ↑	PESQ ↑	STOI ↑		PESQ ↑	STOI ↑	PESQ ↑	STOI ↑		PESQ ↑	STOI ↑	PESQ ↑	STOI ↑
Unprocessed	-	1.671	0.504	1.246	0.488	-	1.615	0.600	1.153	0.499	-	1.827	0.572	1.235	0.498
WPE	120	2.776	0.832	1.265	0.507	80	2.275	0.778	1.168	0.506	160	2.760	0.854	1.251	0.522
WPE-a	400	2.455	0.843	1.183	0.499	400	2.467	0.774	1.211	0.515	400	2.459	0.833	1.290	0.529
PnPWPE	120	2.543	0.770	1.753	0.566	80	3.112	**0.898**	**1.583**	0.572	160	2.781	0.842	1.886	0.562
DWPE	123	2.217	0.626	1.270	0.504	83	2.911	0.734	1.286	0.584	163	2.950	0.833	1.351	0.499
Proposed	123	**3.276**	**0.886**	**2.123**	**0.708**	83	**3.167**	0.805	1.465	**0.601**	163	**3.248**	**0.917**	**2.647**	**0.708**

Methods	Node 1					Node 2					Node 3				
	L	Noisy (10 dB)		Noisy (20 dB)		L	Noisy (10 dB)		Noisy (20 dB)		L	Noisy (10 dB)		Noisy (20 dB)	
		PESQ ↑	STOI ↑	PESQ ↑	STOI ↑		PESQ ↑	STOI ↑	PESQ ↑	STOI ↑		PESQ ↑	STOI ↑	PESQ ↑	STOI ↑
Unprocessed	-	1.562	0.535	1.687	0.549	-	1.516	0.567	1.510	0.596	-	1.585	0.552	1.737	0.567
WPE	120	1.613	0.584	1.898	0.651	80	1.571	0.592	1.688	0.649	160	1.627	0.607	1.981	0.685
WPE-a	400	1.611	0.584	1.864	0.676	400	1.561	0.584	1.782	0.623	400	1.613	0.588	1.962	0.673
PnPWPE	120	2.145	0.661	2.431	0.735	80	1.808	0.661	2.028	**0.725**	160	2.169	0.677	**2.447**	0.757
DWPE	123	2.181	0.580	2.165	0.695	83	1.899	0.632	2.021	0.678	163	1.842	0.640	1.841	0.672
Proposed	123	**2.372**	**0.775**	**2.435**	**0.778**	83	**2.288**	**0.746**	**2.463**	0.706	163	**2.875**	**0.728**	2.395	**0.790**

1) Step 1: Optimization with respect to $\mathbf{u}_i(k)$. The optimization w.r.t. $\mathbf{u}_i(k)$ is a separable least square problem with it solution given by

$$\mathbf{u}_i^{(\ell+1)}(k) = [\mathbf{R}_{\mathbf{y}_i}{}^{(\ell+1)}(k)]^{-1}\mathbf{p}_{\mathbf{y}_i}^{(\ell+1)}(k), \tag{20}$$

where $\mathbf{R}_{\mathbf{y}_i}^{(\ell+1)}(k) = \sum_{n=1}^{N} \frac{\mathbf{y}_i(n-\tau,k)\mathbf{y}_i^{\mathrm{H}}(n-\tau,k)}{\lambda_i^{(\ell+1)}(n,k)}$ and $\mathbf{p}_{\mathbf{y}_i}^{(\ell+1)}(k) = \sum_{n=1}^{N} \frac{\mathbf{y}_i(n-\tau,k)\tilde{X}_i^{(\ell+1)}(n,k)}{\lambda_i^{(\ell+1)}(n,k)}$. In the above solution, $\lambda_i^{(\ell+1)}(n,k) = \frac{2\sigma_i^{(\ell)}(n,k)}{2+\rho\sigma_i^{(\ell)}(n,k)}$ and $\tilde{X}_i{}^{(\ell+1)}(n,k)$ is given by

$$\begin{aligned}\tilde{X}_i{}^{(\ell+1)}(n,k) =& X_{\mathrm{ref},i}(n,k) - \frac{\rho}{2}\lambda_i^{(\ell+1)}(n,k)\big[\hat{R}_i^{(\ell)}(n,k) \\ &+ V_i^{(\ell)}(n,k) - P_i^{(\ell)}(n,k)\big].\end{aligned} \tag{21}$$

Within each frequency band, the vector $\overline{\mathbf{w}}_i(k)$ is extracted from $\mathbf{u}_i(k)$ and subsequently employed in (4) to construct the matrix $\hat{\mathbf{S}}_i$, which is used in subsequent processing steps. Moreover, $\overline{\mathbf{w}}_i(k)$ serves as the compressor in (14). By utilizing $\hat{S}_i(n,k)$, we can estimate $\sigma_i(n,k)$ as follows:

$$\sigma_i^{(\ell+1)}(n,k) = |\hat{S}_i{}^{(\ell)}(n,k)|^2. \tag{22}$$

2) Step 2: Optimization w.r.t. $\mathbf{R}_i$. In the context of RED, optimization problem (19) can be solved by

$$\mathbf{R}_i^{(\ell+1,a)} = \mu\tilde{\mathbf{R}}_i{}^{(\ell+1,a)} + (1-\mu)\Omega(\tilde{\mathbf{R}}_i{}^{(\ell+1,a)}), \tag{23}$$

with

$$\tilde{\mathbf{R}}_i^{(\ell+1)} = \hat{\mathbf{S}}_i^{(\ell+1)} - \mathbf{V}_i^{(\ell)} + \mathbf{P}_i^{(\ell)}. \tag{24}$$

$\mu = \frac{\rho}{\rho+\beta_0}$ is a scalar parameter. $a = 1, \cdots, A$ denotes the index for the inner iterations. Here, the denoiser $\Omega(\cdot)$ defined in Eq. (24) is implemented using a pre-trained Frequency Recurrent Convolutional Recurrent Network (FRCRN) [17], as shown in Fig. 2(d), chosen due to the flexibility of the proposed framework.

3) Step 3: Optimization w.r.t. $\mathbf{V}_i$. The solution to this optimization problem can be directly expressed as:

$$\mathbf{V}_i^{(\ell+1)} = \hat{\mathbf{S}}_i{}^{(\ell+1)} - \mathbf{R}_i^{(\ell+1)} + \mathbf{P}_i^{(\ell)}. \tag{25}$$

4) Step 4: Optimization w.r.t. $\mathbf{P}_i$. The update of this dual variable follows the standard procedure:

$$\mathbf{P}_i^{(\ell+1)} = \mathbf{P}_i^{(\ell)} + \hat{\mathbf{S}}_i{}^{(\ell+1)} - \mathbf{V}_i^{(\ell+1)} - \mathbf{R}_i^{(\ell+1)}. \tag{26}$$

As illustrated in Fig. 2(c), the variables $\mathbf{u}_i(k)$, $\mathbf{R}_i$, $\mathbf{V}_i$, and $\mathbf{P}_i$ are iteratively updated until convergence. The final value of $\mathbf{R}_i$ will then be taken as the estimated speech.

IV. EXPERIMENTAL RESULTS

Experimental settings: To evaluate our proposed method, we constructed an enclosed environment containing one sound source and four distributed nodes, as illustrated in Fig. 2. The room measured approximately 13 m×6.5 m×3.5m, while the sound source was positioned at (9.60 m, 4.55 m, 1.92 m). The four nodes in our experiment were equipped with 6, 4, 8, and 2 microphones, respectively. Room impulse responses (RIRs) were synthesized using the image method [18]. Clean speech from the Wall Street Journal dataset (WSJ0) [19] was then convolved with these RIRs to generate distributed, reverberant speech. The reverberation time (T_{60}) was set to approximately 790 ms. To simulate a noisy distributed scenario, white Gaussian noise was added to the convolved speech at signal-to-noise ratios (SNRs) of 0, 10 and 20 dB. Finally, all test utterances were segmented into 4-second intervals with a sampling rate of 16 kHz.

Method comparison and evaluation: In the experiments, we evaluated four distinct comparative methods. Specifically, `WPE` and `WPE-a` served as baseline methods, both belonging to the vanilla WPE algorithm: the former utilizes microphones within an individual node, whereas the latter leverages all

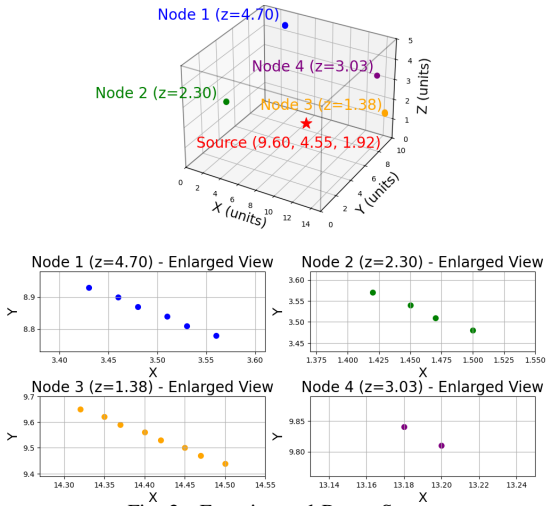

Fig. 2. Experimental Room Setup.

20 microphones distributed across the entire network. Additionally, `PnPWPE` exclusively incorporated data-driven priors, while `DWPE` employed a distributed processing strategy. To objectively quantify the experimental results, we adopted two widely recognized evaluation metrics in the speech dereverberation task: Perceptual Evaluation of Speech Quality (PESQ) [20] and Short-Time Objective Intelligibility (STOI) [21].

Implementation details: All comparison methods were implemented in the short-time Fourier transform (STFT) domain using a Hann window, with a frame length of 32 ms and 75% overlap. Regarding parameter settings, the filter order L was set to 20 and the time delay τ was set to 5. The trade-off parameter ρ was fixed at 0.1, while the scalar μ was set to 0.5 initially and increased in increments of 0.01. We also set the inner iteration A to 1 to expedite the optimization process. Furthermore, for both `DWPE` and the proposed method, nodes exchange information every other iteration [6].

Results analysis: Table I summarizes the comparative results for nodes 1 to 3 across all evaluation metrics. These results indicate that the `proposed` method outperforms the comparison methods under most experimental scenarios. Specifically, in terms of the PESQ metric, the `proposed` method achieves improvements of 0.733 and 1.059 over `PnPWPE` and `DWPE`, respectively, under noise-free conditions, and improvements of 0.370 and 0.853 under noisy conditions with an SNR of 0 dB. Furthermore, the `proposed` algorithm outperforms `WPE-a` despite using a smaller filter order. These observations underscore the effectiveness of incorporating deep data priors within the distributed dereverberation network.

V. Conclusion

In this paper, we proposed an effective method for integrating data-driven speech priors to enhance the performance of distributed dereverberation method. Specifically, we adopted the PnP framework with variable splitting of ADMM, namely the RED strategy, to facilitate the optimization of DWPE. Experimental results show that the proposed method significantly improves distributed speech dereverberation performance in both noise-free and noisy conditions.

References

[1] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B. Juang, "Speech dereverberation based on maximum-likelihood estimation with time-varying gaussian source model," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 8, pp. 1512–1527, 2008.

[2] S. R. Chetupalli and T. V. Sreenivas, "Late reverberation cancellation using bayesian estimation of multi-channel linear predictors and student's t-source prior," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 6, pp. 1007–1018, 2019.

[3] R. Ikeshita, N. Kamo, and T. Nakatani, "Blind signal dereverberation based on mixture of weighted prediction error models," *IEEE Signal Process Lett.*, vol. 28, pp. 399–403, 2021.

[4] Z. Wang, "Usdnet: Unsupervised speech dereverberation via neural forward filtering," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 32, pp. 3882 – 3895, 2024.

[5] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Speech dereverberation based on variance-normalized delayed linear prediction," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 7, pp. 1717–1731, 2010.

[6] Z. Yang, M. Zhang, and J. Chen, "Distributed speech dereverberation using weighted prediction error," *Signal Process.*, vol. 225, pp. 109577, 2024.

[7] J. Chen, M. Zhao, X. Wang, C. Richard, and S. Rahardja, "Integration of physics-based and data-driven models for hyperspectral image unmixing: A summary of current methods," *IEEE Signal Process Mag.*, vol. 40, no. 2, pp. 61–74, 2023.

[8] X. Wang, J. Chen, Q. Wei, and C. Richard, "Hyperspectral image super-resolution via deep prior regularization with parameter estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 1708–1723, 2021.

[9] R. Ahmad, C. A Bouman, G. T Buzzard, S. Chan, S. Liu, E. T Reehorst, and P. Schniter, "Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery," *IEEE Signal Process Mag.*, vol. 37, no. 1, pp. 105–116, 2020.

[10] C. Chang, Z. Yang, and J. Chen, "Plug-and-play mvdr beamforming for speech separation," in *Proc. 2024 IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2024, pp. 1346–1350.

[11] K. Matsumoto and K. Yatabe, "Determined BSS by combination of iva and dnn via proximal average," in *Proc. 2024 IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2024, pp. 871–875.

[12] H. Hu, Z. Yang, J. Chen, and L. Zhang, "Speech dereverberation with deconvolution regularized by denoising," in *Asia Pacific Signal, Inf. Process. Assoc. Annual Summit Conf. (APSIPA ASC)*. IEEE, 2024, pp. 1–6.

[13] Z. Yang, J. Chen, C. Richard, and J. Li, "Plug-and-play wpe guided by deep spectrum estimation for speech dereverberation," in *Int. Conf. Signal Process., Commun. Comput. (ICSPCC)*. IEEE, 2024, pp. 1–5.

[14] Z. Yang, W. Yang, K. Xie, and J. Chen, "Integrating data priors to weighted prediction error for speech dereverberation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 2024.

[15] Z. Yang, W. Yang, K. Xie, and J. Chen, "Speech dereverberation using weighted prediction error with priors learnt from data," in *Proc. European Signal Process. Conf. (EUSIPCO)*. IEEE, 2023, pp. 356–360.

[16] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (RED)," *SIAM J. Imaging Sci.*, vol. 10, no. 4, pp. 1804–1844, 2017.

[17] S. Zhao, B. Ma, K. N. Watcharasupat, and W.-S. Gan, "Frcrn: Boosting feature representation using frequency recurrence for monaural speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2022, pp. 9281–9285.

[18] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943–950, 1979.

[19] J. S. Garofolo, "Csr-i (wsj0) complete ldc93s6a," *Linguistic Data Consortium, Philadelphia*, vol. 83, 1993.

[20] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 2, pp. 749–752, 2001.

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