IMPORTANCE OF SWITCH OPTIMIZATION CRITERION IN SWITCHING WPE DEREVERBERATION

Naoyuki Kamo, Rintaro Ikeshita, Keisuke Kinoshita, Tomohiro Nakatani

NTT Corporation

ABSTRACT

Weighted prediction error (WPE) is a fundamental dereverberation method to predict the late reverberation component of an observed signal based on linear prediction (LP). Recently, WPE was extended to Switching WPE (SwWPE), which optimizes (i) multiple LP filters and (ii) switching parameters to determine the best LP filter used for each time-frequency bin. Conventionally, these parameters are optimized based on the maximum likelihood (ML) criterion, but this is not optimal in terms of signal quality, such as signal-to-distortion ratio (SDR) and word error rate (WER) of automatic speech recognition. We thus propose a new SwWPE processing flow that enables us to optimize switching parameters based on an arbitrary optimization criterion. Using oracle clean signals, we demonstrate the potential performance of our new approach with an SDR maximization criterion, revealing that it can significantly improve the SDR and WER obtained by the conventional ML-based SwWPE. This motivates us to propose new SwWPE processing in which the switching parameters are externally estimated using a deep neural network (DNN) that is trained with an end-to-end SDR maximization criterion. The experimental result clearly demonstrates the improved SDR performance of the new approach compared to the conventional WPE and SwWPE.

***Index Terms*—** Dereverberation, linear prediction (LP), weighted prediction error (WPE), speech recognition

1. INTRODUCTION

When a speech signal is captured in a room with a distant microphone, the observed signal inevitably contains reverberation [1], which is known to degrade the performance of many audio signal processing systems such as ASR [2], and the audible quality [3]. To mitigate the reverberation effect in the observed signal, much research over the last few decades has focused on dereverberation algorithms [4, 5].

Among many dereverberation algorithms developed, weighted prediction error (WPE)-based dereverberation [6–9], which is based on linear prediction (LP) [10–13], has been shown to work well as a front end of ASR systems [14]. Its effectiveness has been proven based on many academic databases [15–18] and a commercial product [19]. However, WPE sometimes poses a problem for handling *noisy* reverberant speech, since in that case a LP filter estimated in WPE has to accomplish two tasks simultaneously, i.e., reducing reverberation while not amplifying the noise component, rendering the overall dereverberation processing less effective.

To address this problem, a novel switching mechanism was recently introduced to the WPE framework [20, 21]. This switching-WPE (hereafter, SwWPE) effectively utilizes *multiple* LP filters (as oppose to *one* in the original WPE), and adaptively use one of the filters in each time-frequency (TF) bin to achieve an optimal dereverberation even in conditions the original WPE could not appropriately handle. The LP filters and the switches that determine which filter to be used in each TF bin are optimized with a maximum likelihood (ML) criterion based on a probabilistic model of an observed signal. SwWPE has been shown to significantly outperform WPE in terms of the ASR performance as well as signal-level evaluation metrics [20, 21].

While the ML formulation employed in SwWPE is mathematically rigorous and is shown to work adequately in many cases, we argue that we can significantly broaden the application area of SwWPE if the system can be optimized with other optimization criteria; for example, the ML criterion clearly cannot be optimal for applications such as ASR [22, 23]. In fact, as will be shown later in this paper, the original SwWPE does not necessarily show improved performance in an ASR evaluation.

The contribution of this paper is twofold. First, we experimentally show that, just by employing a different algorithm for the switch optimization, we can significantly alter the behavior of SwWPE and improve its performance, for example in terms of ASR and signal-to-distortion ratio (SDR) [24]. This experiment should be considered as a proof of concept because the switch estimation algorithm therein utilizes reference/oracle information that is not available in most scenarios. Nevertheless, this experiment and the findings from it is essential in a sense that it clearly unveils the potential of the switching mechanism in the SwWPE, and significance of the optimization algorithm for the switch. Secondly, based on the aforementioned experimental findings, we develop a deep neural network (DNN)-based switch-optimization framework. By using the newly-developed framework, we can, for example, optimize the switch for the SDR maximization and notably achieve significant SDR improvement. As a byproduct of this SDR maximization, we shown that we can achieve better ASR performance in comparison with the original ML-based SwWPE in severe noisy conditions.

2. CONVENTIONAL SWITCHING WPE (SWWPE)

In this section, we define the dereverberation problem addressed in this paper and briefly review WPE [6, 7] and its extension SwWPE [20, 21]. Note that we deal with single-channel scenarios in this paper.

In the short-term Fourier transform (STFT) domain, observed signal $\boldsymbol{x} \coloneqq \{x_{f,t}\}_{f,t}$ using a single microphone is modeled as

$$x_{f,t} = \sum_{\ell=0}^{L_h} h_{f,\ell} s_{f,t-\ell} + n_{f,t} \in \mathbb{C}, \tag{1}$$

where $f = 1, \ldots, F$ and $t = 1, \ldots, T$ denote the frequency bin and time frame indexes, respectively. $s_{f,t}$ is the clean speech signal, $n_{f,t}$ is the background noise, and $(h_{f,\ell})_{\ell=0}^{L_h}$ with order L_h models the acoustic transfer function (ATF) between the speech source and

microphone. The goal of the dereverberation is to predict the late reverberantion component defined as $\sum_{\ell=D}^{L_h} h_{f,\ell} s_{f,t-\ell}$ with $0 < D \ll L_h$ and subtract it from $\boldsymbol{x}$ to obtain an enhanced signal $\boldsymbol{z} := \{z_{f,t}\}_{f,t}$:

$$z_{f,t} \approx x_{f,t} - \sum_{\ell=D}^{L_h} h_{f,\ell} s_{f,t-\ell} = \sum_{\ell=0}^{D-1} h_{f,\ell} s_{f,t-\ell} + n_{f,t}.$$

WPE predicts the late reverberation component based on LP with prediction filter $\boldsymbol{g}_f \in \mathbb{C}^L$ (L is the filter order) and achieves dereverberation as

$$z_{f,t} = x_{f,t} - \boldsymbol{g}_f^{\mathsf{H}} \overline{\boldsymbol{x}}_{f,t} \in \mathbb{C}, \tag{2}$$

$$\overline{\boldsymbol{x}}_{f,t} := [\, x_{f,t-D}, \ldots, x_{f,t-D-L+1} \,] \in \mathbb{C}^L. \tag{3}$$

Here, $\overline{\boldsymbol{x}}_{f,t}$ is obtained by stacking the past signals, $\cdot^{\mathsf{H}}$ denotes the Hermitian transpose, and $D > 0$ is a prediction delay, which is essential to prevent speech distortions [6, 10]. Let p_x and p_z denote the probability density functions of $\boldsymbol{x}$ and $\boldsymbol{z}$ respectively. Then, since we have $p_x(\boldsymbol{z}) = p_z(\boldsymbol{z})$ by relation (2), filter $\boldsymbol{g}_f$ can be estimated in a maximum likelihood (ML) sense if we model $\boldsymbol{z}$ to obey, for example, complex Gaussian distributions with the zero mean and time-frequency dependent variances $\boldsymbol{\lambda} := \{\lambda_{f,t}\}_{f,t}$:

$$-\log p(\boldsymbol{z}) = \sum_{f=1}^{F} \sum_{t=1}^{T} \left[\frac{|z_{f,t}|^2}{\lambda_{f,t}} + \log \lambda_{f,t} \right] + \text{const.} \tag{4}$$

One of WPE's major drawbacks is that filter $\boldsymbol{g}_f$ is time-invariant and cannot capture the spatial nonstationarity of the background noise, for example. To mitigate this model misspecification, the authors have recently proposed an extension of WPE, which we call Switching WPE (SwWPE) [20, 21]. Unlike WPE, SwWPE has N (≥ 2) LP filters $\boldsymbol{g}_{f,1}, \ldots, \boldsymbol{g}_{f,N} \in \mathbb{C}^L$ and performs dereverberation on the basis of

$$z_{f,t} = x_{f,t} - \sum_{i=1}^{N} \alpha_{f,t,i} \boldsymbol{g}_{f,i}^{\mathsf{H}} \overline{\boldsymbol{x}}_{f,t} \in \mathbb{C}, \tag{5}$$

$$\alpha_{f,t,i} \in \{0, 1\}, \quad \sum_{i=1}^{N} \alpha_{f,t,i} = 1, \tag{6}$$

where $\boldsymbol{\alpha} := \{\alpha_{f,t,i}\}_{f,t,i}$ is called a binary switch and is optimized to determine the *best* filter to use for each TF bin. Thanks to this switching mechanism, SwWPE can perform dereverberation in a time-varying way, unlike WPE.

LP filters $\boldsymbol{g} := \{\boldsymbol{g}_{f,i}\}_{f,i}$ and switch $\boldsymbol{\alpha}$ can jointly be optimized in an ML sense when assuming, e.g., (4) for the enhanced signal $\boldsymbol{z}$ (see [20, 21] for details). The optimization algorithm developed in [20, 21] alternately updates $(\boldsymbol{\alpha}, \boldsymbol{\lambda})$, and $\boldsymbol{g}$ as follows:

$$v_{f,t,i} = |x_{f,t} - \boldsymbol{g}_{f,i}^{\mathsf{H}} \overline{\boldsymbol{x}}_{f,t}|^2, \tag{7}$$

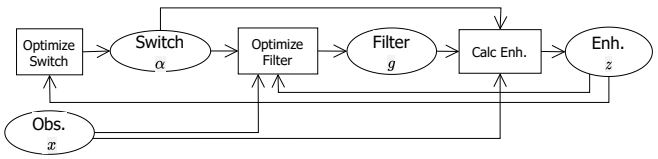
$$\alpha_{f,t,i} = \begin{cases} 1 & \text{if } i = \operatorname*{argmin}_j \{\, v_{f,t,j} \mid j = 1, \ldots, N \,\} \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

$$\lambda_{f,t} = \sum_{i=1}^{N} \alpha_{f,t,i} \cdot v_{f,t,i} = \min\{\, v_{f,t,1}, \ldots, v_{f,t,N} \,\}, \tag{9}$$

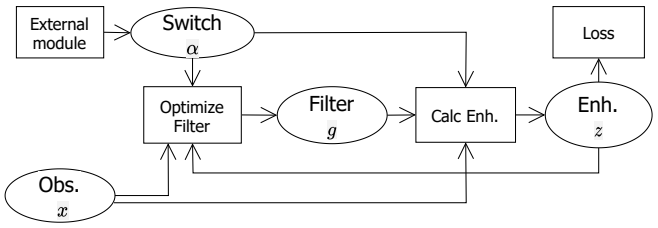
$$\boldsymbol{g}_{f,i} = \left(\sum_{t=1}^{T} \alpha_{f,t,i} \frac{\overline{\boldsymbol{x}}_{f,t} \overline{\boldsymbol{x}}_{f,t}^{\mathsf{H}}}{\lambda_{f,t}} \right)^{-1} \left(\sum_{t=1}^{T} \alpha_{f,t,i} \frac{\overline{\boldsymbol{x}}_{f,t} x_{f,t}^{\mathsf{H}}}{\lambda_{f,t}} \right). \tag{10}$$

3. SWWPE WITH EXTERNAL SWITCH ESTIMATION

With the conventional SwWPE, the switch $\boldsymbol{\alpha}$ is optimized based on the ML criterion or, more specifically, by the minimizing weighted power of dereverberated signals shown in the right-hand side of Eq. (4). This criterion, however, is not necessarily optimal for speech enhancement and ASR, for example. We thus propose a new SwWPE processing flow, shown in Fig. 1 (b) that can estimate the switches based on an arbitrary criterion. We explain the new flow in this section and examine it with different criteria in the following sections.



(a) ML-based Switching WPE (Conventional)



(b) Switching WPE with external switch estimation (Proposed)

Fig. 1: Processing flows

In the new flow, switches $\boldsymbol{\alpha} := \{\alpha_{f,t,i}\}_{f,t,i}$ are estimated by an external module and then utilized to estimate LP filters $\boldsymbol{g}$. A certain loss function $\mathcal{L}(z(\boldsymbol{\alpha}))$ is used for the estimation of $\boldsymbol{\alpha}$ as

$$\boldsymbol{\alpha}^\star = \operatorname*{argmin}_{\boldsymbol{\alpha}} \mathcal{L}(\boldsymbol{\alpha}) = \operatorname*{argmin}_{\boldsymbol{\alpha}} (-\mathrm{SDR}(\widehat{\mathsf{s}}(\boldsymbol{\alpha}), \mathsf{s}))$$

where $\boldsymbol{z}(\boldsymbol{\alpha}) := \{z_{f,t}\}_{f,t}$ denotes dereverberated signals obtained dependent on $\boldsymbol{\alpha}$ based on fixed dereverberation steps later described in Eqs. (12) and (13). With this function, we can flexibly introduce different criteria for optimizing the system output in an end-to-end manner. For generality, we use soft switches $\boldsymbol{\alpha}$ in the new flow unlike the conventional SwWPE, which uses hard switches taking only binary values. The soft switches are defined as

$$0 \leq \alpha_{f,t,i} \leq 1, \quad \sum_{i=1}^{N} \alpha_{f,t,i} = 1. \tag{11}$$

For estimation of the LP filters $\boldsymbol{g}$ given the estimated soft switches, we basically follow the steps in Eqs. (5) to (10) derived for the conventional SwWPE based on the ML criterion with hard switches. In the new flow, the soft switches are externally estimated and not updated during the steps. As a consequence, $\boldsymbol{g}$ and $\boldsymbol{z}(\boldsymbol{\alpha}) := \{z_{f,t}\}_{f,t}$ are iteratively and alternately updated by

$$z_{f,t} = x_{f,t} - \sum_{i=1}^{N} \alpha_{f,t,i} \boldsymbol{g}_{f,i}^{\mathsf{H}} \overline{\boldsymbol{x}}_{f,t} \in \mathbb{C}, \tag{12}$$

$$\boldsymbol{g}_{f,i} = \left(\sum_{t=1}^{T} \alpha_{f,t,i} \frac{\overline{\boldsymbol{x}}_{f,t} \overline{\boldsymbol{x}}_{f,t}^{\mathsf{H}}}{|z_{f,t}|^2} \right)^{-1} \left(\sum_{t=1}^{T} \alpha_{f,t,i} \frac{\overline{\boldsymbol{x}}_{f,t} x_{f,t}^{\mathsf{H}}}{|z_{f,t}|^2} \right). \tag{13}$$

4. EVALUATING EFFICACY OF SWITCHING MECHANISM USING ORACLE SWITCH

We experimentally show the potential performance of the proposed SwWPE processing (Fig. 1-(b)) by considering an ideal case where an oracle clean signal is available and switch $\boldsymbol{\alpha}$ can be obtained using

it in an oracle manner. To this end, we need to define an oracle switch (Section 4.1), which is not a trivial undertaking compared to, for example, defining an oracle mask commonly used in speech separation tasks [25].

4.1. Oracle switch: Definition and Computation

Let s be an oracle noise-free speech signal in the time domain and $\widehat{\mathsf{s}}(\boldsymbol{\alpha})$ be the enhanced signal obtained by the proposed SwWPE processing with switch $\boldsymbol{\alpha}$ output by the external module in Fig. 1-(b):

$$\widehat{\mathsf{s}}(\boldsymbol{\alpha}) \coloneqq \mathrm{STFT}^{-1}(\boldsymbol{z}(\boldsymbol{\alpha})). \tag{14}$$

Here, STFT^{-1} means the inverse STFT operation. In this paper, *an oracle switch* $\boldsymbol{\alpha}^\star$ is defined as the switch that minimizes the SDR loss as:

$$\boldsymbol{\alpha}^\star = \operatorname*{argmin}_{\boldsymbol{\alpha}} \mathcal{L}(\boldsymbol{\alpha}) = -\operatorname*{argmax}_{\boldsymbol{\alpha}} \mathrm{SDR}(\widehat{\mathsf{s}}(\boldsymbol{\alpha}), \mathsf{s}). \tag{15}$$

We implement the scale-invariant SDR [24, 26] as the SDR loss:

$$\mathrm{SDR}(\widehat{\mathsf{s}}, \mathsf{s}) \coloneqq 10\log_{10} \frac{\|\mathsf{s}_{\text{target}}\|^2}{\|\widehat{\mathsf{s}} - \mathsf{s}_{\text{target}}\|^2}, \quad \mathsf{s}_{\text{target}} \coloneqq \frac{\langle \widehat{\mathsf{s}}, \mathsf{s}\rangle}{\langle \mathsf{s}, \mathsf{s}\rangle}\mathsf{s}. \tag{16}$$

Here, $\|\cdot\|$ is the Euclidean norm and $\langle\cdot,\cdot\rangle$ is the inner product. Since obtaining the global minimizer of (15) is computationally intractable, an arbitrary local minimum of (15) is also regarded as *an oracle switch* here.

We next explain how to compute an oracle switch $\boldsymbol{\alpha}^\star$ in (15). Since switch $\boldsymbol{\alpha}_{f,t} \coloneqq (\alpha_{f,t,i})_{i=1}^N$ for each time-frequency bin is constrained to (11), we can reparameterize it to $\boldsymbol{\theta}_{f,t} \coloneqq (\theta_{f,t,i})_{i=1}^N \in \mathbb{R}^N$ using the softmax function:

$$\boldsymbol{\alpha}_{f,t} \coloneqq \mathrm{Softmax}(\boldsymbol{\theta}_{f,t}) = \left(\frac{\exp(\theta_{f,t,i})}{\sum_{j=1}^N \exp(\theta_{f,t,j})}\right)_{i=1,\ldots,N}. \tag{17}$$

Then, the SDR loss can be viewed as a function of $\boldsymbol{\theta} \coloneqq \{\boldsymbol{\theta}_{f,t}\}_{f,t}$. Since the loss is differentiable with respect to $\boldsymbol{\theta}$, it can be optimized using a gradient descent method given observed signal $\boldsymbol{x}$ and oracle clean signal s. From the optimized $\boldsymbol{\theta}$, we can obtain an estimation of $\boldsymbol{\alpha}^\star$ using (17). In Section 4.3, we will show the performance of the proposed SwWPE processing given this oracle switch $\boldsymbol{\alpha}^\star$.

4.2. Experimental condition

We carried out an experiment to show the potential performance of the proposed SwWPE processing given oracle switch $\boldsymbol{\alpha}^\star$ as compared with the conventional WPE and SwWPE.

Dataset: We simulated reverberant noisy speech signals using the *SIM* dataset of the REVERB challenge [14]. We randomly obtained speech and noise signals from the dataset. For the room impulse responses (RIR), we used both *near* and *far* RIRs in the dataset. Using these data, we generated 200 observed signals as $\mathsf{x} = \mathsf{h} * \mathsf{c} + \mathsf{n}$, where h, c, and n are the RIR, clean speech, and noise signals, respectively. $\mathsf{h} * \mathsf{c}$ means the convolution of h and c. We set the input signal-to-noise (SNR) ratio to 0, 5, 10, or 20 dB:

$$\text{SNR [dB]} \coloneqq 10\log_{10} \frac{\|\mathsf{h}_{32} * \mathsf{c}\|^2}{\|\mathsf{n}\|^2} \in \{0, 5, 10, 20\}, \tag{18}$$

where h_{32} is obtained from h by extracting the consecutive samples with a length of 32 ms (512 points) from its peak point. We define the reference signal used in the SDR loss (15) by $\mathsf{s} \coloneqq \mathsf{h}_{32} * \mathsf{c}$.

Evaluation criteria: We measured (i) SDR using BSS Eval v4 (BSS-SDR)[1] [27] and (ii) WER of ASR. To evaluate WER, we used an ASR backend of the ESPnet[2] [28] developed for the REVERB challenge. To the best of our knowledge, this ASR system, based on a transformer-based model [29], can be considered as the best baseline for the REVERB challenge.

Other conditions: The values of LP filters were initialized to be zero. The sampling rate was 16 kHz and the STFT window length and shift lengths were 512 (32 ms) and 256 (16 ms), respectively. The Hann window was used for STFT. The number of iterations for optimizing the LP filters was set to 3 for all methods.

4.3. Experimental result

Table 1 shows the BSS-SDR and WER results obtained by the conventional WPE/SwWPE and the proposed SwWPE processing with the help of the oracle switch.

As we expected, the proposed SwWPE processing given the oracle switch significantly outperformed the conventional SwWPE in terms of SDR, which clearly shows the great potential of this new approach. Moreover, by increasing the number of LP filters, i.e., N, the SDR scores were greatly improved with the new approach. This trend cannot be seen with the conventional SwWPE, thus highlighting the novelty of the proposed SwWPE processing.

As for the WER scores, the proposed SwWPE processing given the oracle switch clearly provided better WER than the conventional methods in adverse environments, but the improvement is not significant. This may be because the new method seeks to minimize the negative SDR but is not correlated to WER. This observation urges us to use the ASR loss for optimizing an oracle switch, but we will leave this for future research.

5. SWWPE WITH DNN SWITCH ESTIMATION

In the previous section, we demonstrated that the proposed processing can provide much higher performance than the conventional SwWPE. Here, we propose to estimate the switch using DNN.

5.1. DNN-based switch estimation with SDR loss

A switch estimation DNN with parameter W is defined as

$$\boldsymbol{\alpha} = \mathrm{Softmax}(\mathrm{DNN}_W(\boldsymbol{x})). \tag{19}$$

The DNN parameter W is trained using training data and fixed during testing. Here we explain how to train the DNN parameter W. The enhanced signal is expressed in the time domain as

$$\widehat{\mathsf{s}}(W) \coloneqq \mathrm{STFT}^{-1}(\boldsymbol{z}(\boldsymbol{\alpha})) = \mathrm{STFT}^{-1}(\boldsymbol{z}(\mathrm{Softmax}(\mathrm{DNN}_W(\boldsymbol{x})))),$$

which is a function of W and $\boldsymbol{x}$. When using the SDR loss function as in (15), i.e., $\mathcal{L}(W) = -\mathrm{SDR}(\widehat{\mathsf{s}}(W), \mathsf{s})$, we can backpropagate the SDR loss error and optimize W.

5.2. Experimental condition

We compared the performance of the new method with switches estimated by DNN to the conventional WPE and SwWPE. We used the same evaluation data and ASR backend as in Section 4.2. The training data for the switch estimation DNN was created in a similar manner to the evaluation data. The DNN architecture of (19)

[1] https://github.com/sigsep/sigsep-mus-eval

[2] https://github.com/espnet/espnet/tree/master/egs2/reverb/asr1

Table 1: BSS-SDR [dB] and WER [%] obtained by each method

		Input SNR	0 dB	5 dB	10 dB	20 dB	0 dB	5 dB	10 dB	20 dB
Method	Switch estimation	N	SDR [dB] (higher is better)				WER [%] (lower is better)			
Observation	-	-	-1.13	2.72	5.46	7.70	23.7	11.4	8.25	7.05
WPE	-	-	-0.71	3.27	6.20	8.70	23.8	11.35	8.25	7.05
SwWPE (Fig. 1-(a))	ML-based	2	-0.37	3.72	6.89	10.0	23.9	**11.1**	**7.95**	**6.75**
		3	-0.25	3.81	6.83	9.58	26.8	11.5	8.10	6.90
SwWPE (Fig. 1-(b))	DNN	2	1.42	5.38	8.24	11.2	24.3	12.0	8.25	7.05
		3	**1.57**	**5.71**	**8.76**	**11.9**	**21.8**	11.5	8.35	7.10
	Oracle	2	6.51	9.60	12.3	15.0	10.7	8.35	7.55	6.85
		3	10.1	13.1	15.6	18.1	7.55	6.85	6.75	6.6

was 2 layers BLSTM with 512 nodes for each layer. The input feature of the DNN consisted of the amplitude and phase spectra of an observed signal.

For the proposed SwWPE processing with the DNN switch estimation module, the number of iterations for optimizing filter $\boldsymbol{g}$ was set to one each for the training and evaluation stages. For the conventional WPE and SwWPE, the number was set to three. Other experimental conditions were the same as those in Section 4.2.

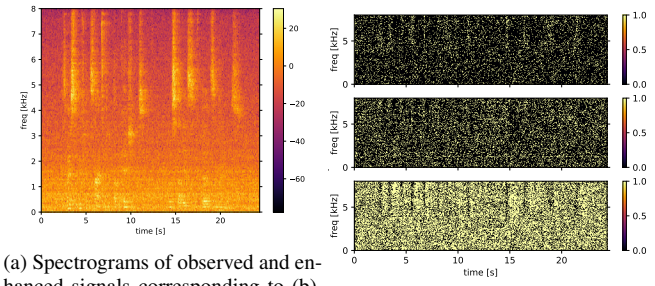
5.3. Experimental result

Table 1 again shows the BSS-SDR and WER results for each method. As we expected, the proposed SwWPE processing with the DNN switch estimation module outperformed the conventional WPE and SwWPE in terms of SDR, implying the novelty of the proposed approach of introducing the SDR maximization criterion for estimating the switch. One might guess that DNN was successfully trained to estimate switches similar to the oracle switch, but this is not the case because, as shown in Fig. 2 the switch estimated by the DNN was completely different from the oracle switch. In fact, our new method slightly degraded the WER scores given by SwWPE when the input SNR was 5 dB or higher, while the oracle method consistently improved the WERs. This issue may be resolved by training the DNN based on the ASR loss instead of the SDR loss, which we will leave to our future work.

5.4. Exemplary visualization of estimated switch

Figure 2 shows an exemplary image of the switch estimated by each method for an observed signal with 0 dB of SNR. The switch estimated by the DNN (c) roughly captures the speech and noise segments of the observed signal (the top image of Fig. 2 (a)). More concretely, the sum of the top and second plots of (c) roughly corresponds to the speech segments and the bottom plot corresponds to the noise segments. This phenomenon is interesting because the switch estimation DNN used to estimate (c) is trained based on the SDR maximization with the proposed SwWPE processing (Fig. 1 (b)) and not on estimating the power spectrums of speech and noise signals. In fact, it is difficult to understand such structures for the switch obtained by the conventional SwWPE (b), which is based on weighted power minimization. This difference in the switches estimated by the two methods results in a difference in the output enhanced signals, which can be considered as the reason why the proposed method was able to improve the SDR and WER scores for the 0-dB SNR condition (although we cannot confirm any significant difference in the spectrograms of the enhanced signals between the two methods in Fig. 2 (a)).

On the other hand, the oracle switch (d) is completely different from the switch estimated by DNN and appears to be almost random. Contrary to this, however, it gave the highest SDR and WER scores in the 0-dB SNR condition. As a research direction to further improve the proposed SwWPE processing, one can pursue a new optimization criterion to enable DNNs to estimate switches similar to the oracle switch, and we remain it as an important future work.



(a) Spectrograms of observed and enhanced signals corresponding to (b), (c), and (d) from top to bottom

(b) Conventional SwWPE (§2)

(c) DNN switch (§5)

(d) Oracle switch (§4)

Fig. 2: An exemplary comparison of estimated switch $\boldsymbol{\alpha}$ for $N = 3$. The yellow color indicates 1 and the black indicates 0. Since switch $\boldsymbol{\alpha}$ is estimated independently for each frequency bin, we sorted order of $\boldsymbol{\alpha}$ so that colors go from black to yellow from top to bottom.

6. CONCLUSION

The conventional SwWPE optimizes the switching parameters based on the weighted power minimization criterion, which is not necessarily optimal for such applications as speech enhancement and ASR. We therefore proposed a new SwWPE processing flow in which the parameters can be optimized in an arbitrary criterion. Using oracle speech signals, we demonstrated that SwWPE has the potential to provide significantly better performance than the conventional WPE and SwWPE in terms of SDR and WER if the switching parameters can be optimized effectively. We thereby proposed a switch estimation DNN that can be trained based on the end-to-end SDR maximization loss. The experimental results clearly showed the improved SDR performance of the new approach compared to the conventional methods.

7. REFERENCES

[1] H. Kuttruff, *Room acoustics*, CRC Press, 6 edition, 2019.

[2] R. Haeb-Umbach, S. Watanabe, T. Nakatani, M. Bacchiani, B. Hoffmeister, M. L. Seltzer, H. Zen, and M. Souden, "Speech processing for digital home assistants: Combining signal processing with deep-learning techniques," *IEEE Signal processing magazine*, vol. 36, no. 6, pp. 111–124, 2019.

[3] I. J. Tashev, *Sound capture and processing: practical approaches*, John Wiley & Sons, 2009.

[4] P. A. Naylor and N. D. Gaubitch, *Speech dereverberation*, Springer Science & Business Media, 2010.

[5] J. Allen, D. Berkley, and J. Blauert, "Multimicrophone signal-processing technique to remove room reverberation from speech signals," *The Journal of the Acoustical Society of America*, vol. 62, no. 4, pp. 912–915, 1977.

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

[7] T. Yoshioka and T. Nakatani, "Generalization of multi-channel linear prediction methods for blind MIMO impulse response shortening," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 10, pp. 2707–2720, 2012.

[8] A. Jukić, T. van Waterschoot, T. Gerkmann, and S. Doclo, "Multi-channel linear prediction-based speech dereverberation with sparse priors," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 9, pp. 1509–1520, 2015.

[9] 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, Language Process.*, vol. 27, no. 6, pp. 1007–1018, 2019.

[10] K. Kinoshita, M. Delcroix, T. Nakatani, and M. Miyoshi, "Suppression of late reverberation effect on speech signal using long-term multiple-step linear prediction," *IEEE Trans. Audio, Speech, Language Process.*, vol. 17, no. 4, pp. 534–545, 2009.

[11] K. Abed-Meraim, E. Moulines, and P. Loubaton, "Prediction error method for second-order blind identification," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 694–705, 1997.

[12] D. T. Slock, "Blind fractionally-spaced equalization, perfect-reconstruction filter banks and multichannel linear prediction," in *Proc. ICASSP*, 1994, vol. 4, pp. IV/585–IV/588.

[13] D. Gesbert and P. Duhamel, "Robust blind channel identification and equalization based on multi-step predictors," in *Proc. ICASSP*, 1997, vol. 5, pp. 3621–3624.

[14] K. Kinoshita, M. Delcroix, S. Gannot, E. A. Habets, R. Haeb-Umbach, W. Kellermann, V. Leutnant, R. Maas, T. Nakatani, B. Raj, et al., "A summary of the REVERB challenge: state-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, 2016.

[15] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: Analysis and outcomes," *Computer Speech and Language*, vol. 46, pp. 605–626, 2017.

[16] E. Vincent, S. Watanabe, A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech and Language*, vol. 46, pp. 535–557, 2017.

[17] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," in *Proc. Interspeech*, 2018, pp. 1561–1565.

[18] S. Watanabe, M. Mandel, J. Barker, and E. Vincent, "CHiME-6 Challenge: Tackling multispeaker speech recognition for unsegmented recordings," *arXiv:2004.09249*, 2020.

[19] B. Li, T. N. Sainath, A. Narayanan, J. Caroselli, M. Bacchiani, A. Misra, I. Shafran, H. Sak, G. Pundak, K. K. Chin, et al., "Acoustic modeling for Google Home," in *Proc. Interspeech*, 2017, pp. 399–403.

[20] 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.

[21] R. Ikeshita, K. Kinoshita, N. Kamo, and T. Nakatani, "Online speech dereverberation using mixture of multichannel linear prediction models," *IEEE Signal Process. Lett.*, vol. 28, pp. 1580–1584, 2021.

[22] T. Ochiai, S. Watanabe, T. Hori, and J. R. Hershey, "Multichannel end-to-end speech recognition," in *ICML*, 2017, pp. 2632–2641.

[23] S.-W. Fu, T.-W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 9, pp. 1570–1584, 2018.

[24] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[25] D. Wang, "On ideal binary mask as the computational goal of auditory scene analysis," in *Speech separation by humans and machines*, pp. 181–197. Springer, 2005.

[26] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR – half-baked or well done?," in *Proc. ICASSP*, 2019, pp. 626–630.

[27] F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 signal separation evaluation campaign," in *Proc. LVA/ICA*, 2018, pp. 293–305.

[28] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-end speech processing toolkit," in *Proc. Interspeech*, 2018, pp. 2207–2211.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, vol. 30, pp. 5998–6008.