DEEP AD-HOC BEAMFORMING

Xiao-Lei Zhang and Zi-Chen Fan

Center for Intelligent Acoustics and Immersive Communications and
School of Marine Science and Technology, Northwestern Polytechnical University, China

ABSTRACT

Deep learning based speech enhancement methods face two problems. First, their performance is strongly affected by the distance between the speech source and the microphones. Second, unlike conventional methods, deep-learning-based multichannel methods do not show significant performance improvement over their single-channel counterpart. To address the above problem, we propose *deep ad-hoc beamforming*—the first deep-learning-based multichannel speech enhancement method in an ad-hoc microphone array. It serves for scenarios where the microphones are placed randomly in a room and work collaboratively. It aims to pick up speech signals with equally good quality in a range where the array covers. Its core idea is to reweight the estimated speech signals when conducting beamforming, where the weights produced by a neural network are an estimation of the signal-to-noise ratios at the microphone array. We conducted an experiment in a scenario where the location of the speech source is far-field, random, and blind to the microphones. Results show that our method outperforms representative deep-learning-based speech enhancement methods by a large margin.

Index Terms— Ad-hoc microphone array, deep learning, distributed microphone array, MVDR, speech enhancement.

1. INTRODUCTION

Deep neural network (DNN) based speech enhancement has shown its strong discriminative power in adverse acoustic environments. Current DNN-based techniques employ either a single microphone or a conventional multichannel array to pick up speech signals [1]. This paper focuses on the latter. DNN-based multichannel speech enhancement is a fast developing field that can be currently categorized to two research branches. The first branch extracts spatial features as the input of a DNN-based single-channel enhancement method [2]. The second branch, which we denote bravely as *deep beamforming*, estimates a monaural time-frequency (T-F) mask using a single-channel DNN so that the spatial covariance matrices of speech and noise can be derived for beamforming [3, 4].

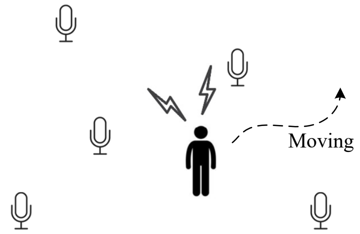


Fig. 1. Illustration of an ad-hoc microphone array.

Although deep beamforming and its application to robust speech recognition has been extensively studied since its first appearance in 2016 [3–14], including the aspects of acoustic features [10, 15], model training [11–14], mask estimations [5], post-processing [16], etc, its performance improvement over the DNN-based single-channel techniques is relatively limited, compared to the impressive performance improvement of the single-channel techniques over conventional statistical signal processing methods [17, 18]. This phenomenon may be caused by that deep beamforming is fundamentally still linear speech enhancement methods; and the spatial information gathered by a conventional microphone array is limited, compared to the strong discriminative power of DNN which is trained from large-scale historical data. Besides, both single-channel and multichannel techniques suffer performance drops when the distance between the speaker and the microphones increases. Finally, people have to carry the microphones and speak closely.

On the other side, the research on ad-hoc microphone arrays is an emerging direction [19–27]. As illustrated in Fig. 1, an ad-hoc microphone array is a set of randomly distributed microphones. The microphones collaborate with each other. Compared to traditional microphone arrays, an ad-hoc microphone array has the following two potentials. First, it has a chance to enhance a speaker's voice with equally good quality in a range where the array covers. Second, its performance is not limited to the physical size of application devices, e.g. cell-phones, gooseneck microphones, or smart speaker boxes. However, current research on ad-hoc microphone arrays is still at the very beginning. For example, some work has focused on the channel selection problem in an ideal scenario where perfect noise estimation and voice activity detection is available [20,26]. Although some work has tried to jointly conduct noise estimation and channel selection, it has to make many assumptions and carry out advanced mathematical formulations [21, 23].

In this paper, we propose *deep ad-hoc beamforming* (DAB)—the first DNN-based multichannel speech enhancement method for ad-hoc microphone arrays. DAB is a supervised method. It revises the signal model of the DNN-based minimum variance distortionless

Corresponding author: Xiao-Lei Zhang

Email: xiaolei.zhang@nwpu.edu.cn, fanzc@mail.nwpu.edu.cn

This work was supported in part by the National Natural Science Foundation of China (NSFC) funding scheme under Project No. 61671381, in part by the Project of the Science, Technology, and Innovation Commission of Shenzhen Municipality under grant No. JCYJ20170815161820095, and in part by the Shaanxi Natural Science Basic Research Program under grant No. 2018JM6035.

response (MVDR) beamforming [3, 4] by reweighting the channels according to the quality of the received signals, where the weights of the channels are produced from a channel-reweighting model trained by supervised off-line learning. Experimental results show that DAB significantly outperforms the DNN-based single-channel enhancement and DNN-based MVDR beamforming in scenarios where a speaker moves randomly in a room.

2. PROBLEM FORMULATION

Suppose that there is only one target speaker. The physical model for the received signals by a standard microphone array of N microphones is assumed to be

$$\mathbf{y}(t,f) = \mathbf{c}(f)s(t,f) + \mathbf{n}(t,f) \tag{1}$$

where $s(t,f)$ is the short-time Fourier transform (STFT) value of the target clean speech at time t and frequency f, $\mathbf{c}(f)$ is the acoustic transfer function from the speaker to the array which is an M-dimensional complex number, and $\mathbf{y}(t,f)$ and $\mathbf{n}(t,f)$ are the received noisy speech and noise respectively. If we denote $\mathbf{x}(t,f) = \mathbf{c}(f)s(t,f)$, then (1) can be rewritten as

$$\mathbf{y}(t,f) = \mathbf{x}(t,f) + \mathbf{n}(t,f). \tag{2}$$

An MVDR beamformer finds a linear estimator $\mathbf{w}_{\text{opt}}(f)$ to filter $\mathbf{y}(t,f)$ by the following equation:

$$\hat{s}(t,f) = \mathbf{w}_{\text{opt}}^H(f)\mathbf{y}(t,f). \tag{3}$$

where $\hat{s}(t,f)$ is an estimate of $s(t,f)$, and $(\cdot)^H$ denotes the conjugate transpose. Although the DNN-based MVDR beamforming [3,4] does not need to know the pattern of the array, the way of viewing all microphones equally important may not be the best. Because the distances between the speaker and the microphones in an ad-hoc microphone array vary in a large range as shown in Fig. 1, the quality of the received signals may vary dramatically accordingly. From [3, 4], we know that, if the DNN-based time-frequency (T-F) masking is not accurate enough which is a situation that the ad-hoc microphone array meets, then $\mathbf{w}_{\text{opt}}(f)$ can be problematic. Hence, the signal model for ad-hoc microphone arrays should be rectified.

3. DEEP AD-HOC BEAMFORMING

The core idea of DAB is to filter $\mathbf{y}(t,f)$ by a channel-reweighting vector $\mathbf{p} = [p_1, \ldots, p_M]^T$ before the MVDR beamforming, such that the channels that output low quality speech signals can be depressed. A system overview is shown in Fig. 2.

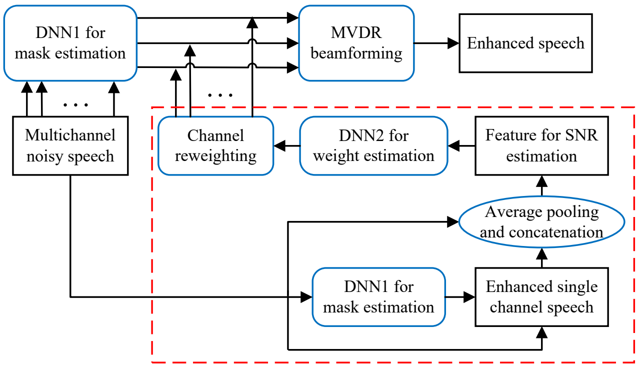
3.1. System overview

The signal model considered in this paper is

$$\mathbf{y}_{\mathbf{p}}(t,f) = \mathbf{p} \odot \mathbf{y}(t,f) = \mathbf{p} \odot \mathbf{x}(t,f) + \mathbf{p} \odot \mathbf{n}(t,f) \tag{4}$$

where $\odot$ is the dot-product operator and $\mathbf{p}$ is the output of a channel-reweighting model $g(\cdot)$ (denoted as DNN2 in Fig. 2), see Section 3.3 for the details about $g(\cdot)$. We denote the covariance matrices of $\mathbf{y}(t,f)$, $\mathbf{x}(t,f)$, and $\mathbf{n}(t,f)$ over time as $\mathbf{\Phi}_{\mathbf{yy}}(f)$, $\mathbf{\Phi}_{\mathbf{xx}}(f)$, and $\mathbf{\Phi}_{\mathbf{nn}}(f)$, respectively. Assuming that $\mathbf{x}(t,f)$ and $\mathbf{n}(t,f)$ are uncorrelated, then we have $\mathbf{\Phi}_{\mathbf{yy}}(f) = \mathbf{\Phi}_{\mathbf{xx}}(f) + \mathbf{\Phi}_{\mathbf{nn}}(f)$, so as to $\mathbf{P} \odot \mathbf{\Phi}_{\mathbf{yy}}(f) = \mathbf{P} \odot \mathbf{\Phi}_{\mathbf{xx}}(f) + \mathbf{P} \odot \mathbf{\Phi}_{\mathbf{nn}}(f)$, where $\mathbf{P} = \mathbf{p}\mathbf{p}^T$. We further denote

$$\mathbf{\Phi}_{\mathbf{p},\mathbf{vv}}(f) = \mathbf{P} \odot \mathbf{\Phi}_{\mathbf{yy}}(f), \quad \forall \mathbf{v} \in \{\mathbf{y}, \mathbf{x}, \mathbf{n}\}. \tag{5}$$



DNN1 for mask estimation
MVDR beamforming
Enhanced speech
...
Multichannel noisy speech
...
Channel reweighting
DNN2 for weight estimation
Feature for SNR estimation
Average pooling and concatenation
DNN1 for mask estimation
Enhanced single channel speech

Fig. 2. Deep ad-hoc beamforming.

Following [3,4], we derive $\mathbf{w}_{\text{opt}}(f)$ and $\hat{s}(t,f)$ as

$$\mathbf{w}_{\text{opt}}(f) = \frac{\widehat{\mathbf{\Phi}}_{\mathbf{p},\mathbf{nn}}^{-1}(f)\hat{\mathbf{c}}(f)}{\hat{\mathbf{c}}^H(f)\widehat{\mathbf{\Phi}}_{\mathbf{p},\mathbf{nn}}^{-1}(f)\hat{\mathbf{c}}(f)} \tag{6}$$

$$\hat{s}(t,f) = \mathbf{w}_{\text{opt}}^H(f)\mathbf{y}_{\mathbf{p}}(t,f) \tag{7}$$

where $\widehat{\mathbf{\Phi}}_{\mathbf{p},\mathbf{nn}}(f)$ is an estimate of $\mathbf{\Phi}_{\mathbf{p},\mathbf{nn}}(f)$:

$$\widehat{\mathbf{\Phi}}_{\mathbf{p},\mathbf{nn}}(f) = \frac{1}{\sum_t \eta(t,f)} \sum_t \eta(t,f)\mathbf{y}_{\mathbf{p}}(t,f)\mathbf{y}_{\mathbf{p}}^H(t,f) \tag{8}$$

and $\hat{\mathbf{c}}(f)$ is an estimate of $\mathbf{c}(f)$, which is the first principal component of an estimated covariance matrix $\widehat{\mathbf{\Phi}}_{\mathbf{p},\mathbf{xx}}(f)$:

$$\widehat{\mathbf{\Phi}}_{\mathbf{p},\mathbf{xx}}(f) = \frac{1}{\sum_t \xi(t,f)} \sum_t \xi(t,f)\mathbf{y}_{\mathbf{p}}(t,f)\mathbf{y}_{\mathbf{p}}^H(t,f) \tag{9}$$

with $\eta(t,f)$ and $\xi(t,f)$ defined as the product of individual estimated T-F masks:

$$\eta(t,f) = \prod_{i=1}^{N}(1 - \hat{m}_i(t,f)), \quad \xi(t,f) = \prod_{i=1}^{N}\hat{m}_i(t,f) \tag{10}$$

where $\hat{m}_i(t,f)$ is an estimate of the ideal T-F mask produced by a regression model $h(\cdot)$ (denoted as DNN1 in Fig. 2) at the i-th channel. See Section 3.2 for the details about $h(\cdot)$.

3.2. Single-channel T-F masking

Suppose the STFT feature is F-dimensional. We denote

$$\hat{\mathbf{m}}_i(t) = [\hat{m}_i(t,1), \ldots, \hat{m}_i(t,F)]^T \tag{11}$$

$$\tilde{\mathbf{y}}_i(t) = [|y|_i(t,1), \ldots, |y|_i(t,F)]^T \tag{12}$$

where $|y|_i(t,f)$ is the amplitude spectrogram of $\mathbf{y}(t,f)$ at the i-th channel. $\hat{\mathbf{m}}_i(t)$ is produced by DNN2 via

$$\hat{\mathbf{m}}_i(t) = h(\tilde{\mathbf{y}}_i(t)) \tag{13}$$

In the training stage of $h(\cdot)$, we construct a corpus $\mathcal{X}_1$ containing the amplitude spectrograms of single-channel noisy speech and its corresponding noise and clean speech components, which are denoted as $|y|(t,f)$, $|s|(t,f)$, and $|n|(t,f)$ respectively. $h(\cdot)$ takes the ideal ratio mask (IRM) as the training target:

$$\text{IRM}(t,f) = \frac{|s|^2(t,f)}{|s|^2(t,f) + |n|^2(t,f)}, \quad \forall f = 1, \ldots, F \tag{14}$$

3.3. Channel-reweighting

Given a test utterance of U frames, we first merge all noisy frames and the estimated clean speech respectively by average pooling:

$$\bar{\tilde{\mathbf{y}}}_i = \frac{1}{U}\sum_{t=1}^{U}\tilde{\mathbf{y}}_i(t), \quad \bar{\hat{\mathbf{s}}}_i = \frac{1}{U}\sum_{t=1}^{U}\hat{\mathbf{s}}_i(t), \quad \forall i = 1,\ldots,M \tag{15}$$

where $\hat{\mathbf{s}}_i(t) = \hat{\mathbf{m}}_i(t) \odot \tilde{\mathbf{y}}_i(t)$, and then concatenate $\bar{\tilde{\mathbf{y}}}_i$ and $\bar{\hat{\mathbf{s}}}_i$ as the input of $g(\cdot)$:

$$p_i = g\left(\left[\bar{\tilde{\mathbf{y}}}_i^T, \bar{\hat{\mathbf{s}}}_i^T\right]^T\right), \quad \forall i = 1,\ldots,M \tag{16}$$

We train $g(\cdot)$ by supervised learning. In the training stage of $g(\cdot)$, we take the ground-truth SNR in the time domain as the target of $g(\cdot)$. Specifically, suppose that a noisy speech sequence in the time domain and its corresponding clean speech and noise components are $\{y_{\mathrm{time}}(t)\}_t$, $\{s_{\mathrm{time}}(t)\}_t$, and $\{n_{\mathrm{time}}(t)\}_t$ respectively, then the training target of $g(\cdot)$ is:

$$p_{\mathrm{opt}} = \sum_t \frac{|s|_{\mathrm{time}}(t)}{|n|_{\mathrm{time}}(t)} \tag{17}$$

We need to construct another training corpus $\mathcal{X}_2$ excluded from $\mathcal{X}_1$ to train $g(\cdot)$, so as to prevent overfitting. We first take the noisy speech in $\mathcal{X}_2$ as the input of DNN1, and then use the estimated clean speech produced by DNN1 as part of the training data $\hat{\mathbf{s}}$. Because $g(\cdot)$ deals with segment-level features, $\mathcal{X}_2$ should be much larger than $\mathcal{X}_1$ in practice. This paper uses DNN as $g(\cdot)$ given the scalability of DNN, though many regression models can be used as well.

4. EXPERIMENTS

4.1. Datasets

We simulated a room where a speaker moves randomly. The farest distance from the speaker to a microphone in the room is limited to at most 20 meters. The clean speech propagates at a speed of 343 meters per second. Its amplitude is fading at a rate of $1/r$ where r is the distance from the speech source. It is corrupted by additive noise. We assumed that (i) the space has weak or no reverberation, (ii) the additive noise is diffuse noise whose energy distributes evenly across the entire space, and (iii) the additive noise between two locations is uncorrelated. The average SNR level at a place of 1 meter away from the speaker was set to 15 dB. Based on the above setting, the relationship between the SNR level and the distance from the speech source is shown in Fig. 3.

The clean speech was generated from the "tr05_org" corpus of CHiME-4, which are 16 bit stereo WAV files sampled at 16 kHz. The additive noise was the babble and factory1 noise from the NOISEX-92 database respectively. For each noise scenario, we selected 500, 5000, and 30 clean utterances from the clean corpus to construct the databases for training DNN1, DNN2, and test, respectively. We further split the noise recordings to three parts as the noise sources for training DNN1, DNN2 and test, respectively. For training DNN1, we synthesized 500 noisy utterances every other meter in a distance range of $[1, 20]$ meters from the location of the speaker, which amounts to 10,000 utterances totally. For training DNN2, we synthesized 100,000 noisy utterances distributed uniformly from 1 to 20 meters from the location of the speaker. We have conducted many tests, see Section 4.2 for their detailed descriptions. For each test, we produced 30 clean utterances from the speech source, and recorded their noisy counterparts at any necessary location of a microphone for evaluation.

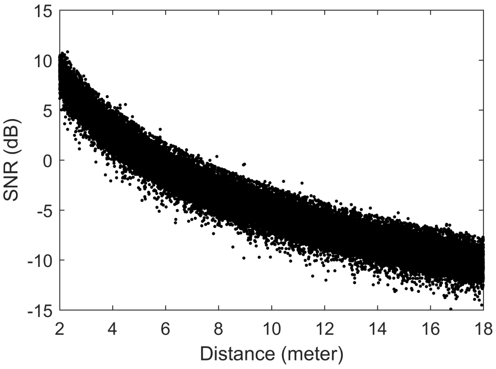


Fig. 3. SNR of the received noisy speech signals at all locations of microphones.

4.2. Experimental settings

We generated an ad-hoc microphone array of 4 microphones for DAB. The microphones are distributed randomly in a distance range of $[a_1, a_2]$ meters from the speaker; the average distance between the microphones and the speaker is b meters, where (a_1, a_2, b) are three experimental parameters. We repeated the above experiment on DAB 20 times. We compared DAB with a DNN-based single-channel speech enhancement method that takes the IRM as the target (DS) and a DNN-based MVDR beamforming with a linear microphone array (DB-LMA) of 4 microphones, where the distance between two neighboring microphones of the LMA is 10 centimeters. The microphone (array) is distributed randomly in a distance range of $[a_1, a_2]$ meters from the speaker. Different from [3], DB-LMA takes the IRM as the training target of the DNN-based noise estimation. We repeated the experiments on DS and DB-LMA 20 times. The average distance between the microphone (array) and the speaker over the 20 independent runs was controlled to be b meters. We adopted two evaluation scenarios, whose (a_1, a_2, b) were set to $(2, 14, 8)$ and $(2, 18, 10)$ respectively. For each evaluation scenario, we reported the average performance of the comparison methods over the 20 runs in terms of short-time intelligibility measure (STOI). We also reported the average STOI scores of the noisy speech recorded by all microphones. The higher the STOI score is, the better the performance is.

We set the frame length and frame shift to 32 and 16 milliseconds respectively, and extracted 257-dimensional STFT features. We used the same DNN1 for DAB, DS, and DB-LMA. The parameter setting of DNN1 is as follows. DNN1 is a standard feedforward DNN. It contains two hidden layers. Each hidden layer has 1024 hidden units. The activation functions of the hidden units and output units are rectified linear unit and sigmoid function, respectively. The number of epochs was set to 30. The batch size was set to 512. The scaling factor for the adaptive stochastic gradient descent was set to 0.0015, and the learning rate decreased linearly from 0.08 to 0.001. The momentum of the first 5 epochs was set to 0.5, and the momentum of other epochs was set to 0.9. A contextual window was used to expand each input frame to its context along the time axis. The window size was set to 7. The parameter setting of DNN2 for DAB was as follows. It contains one hidden layer with 1024 hidden units. The number of epochs was set to 10. The batch size was set to 128. All other parameters were set to the same values as DNN1. The ground-truth SNR p_{opt} was normalized globally to the range $[0, 1]$ so as to fit the output range of the sigmoid function.

Table 1. Comparison results in two evaluation scenarios $(2, 14, 8)$ and $(2, 18, 10)$ respectively in terms of STOI.

(a_1, a_2, b)	Noise type	Noisy	DS	DB-LMA	DAB
$(2, 14, 8)$	Babble	0.6967	0.7529	0.7680	**0.8417**
	Factory	0.7045	0.7727	0.8077	**0.8853**
$(2, 18, 10)$	Babble	0.6523	0.6990	0.6971	**0.8007**
	Factory	0.6518	0.7196	0.7560	**0.8119**

Table 2. Comparison results between the DAB without channel-reweighting and the DAB with the channel-reweighting.

(a_1, a_2, b)	Noise type	without reweighting	with reweighting
$(2, 14, 8)$	Babble	0.8142	**0.8417**
	Factory	0.8789	**0.8853**
$(2, 18, 10)$	Babble	0.7634	**0.8007**
	Factory	0.8045	**0.8119**

4.3. Main results

Table 1 lists the comparison results between DS, DB-LMA, and the proposed DAB in the two evaluation scenarios $(2, 14, 8)$ and $(2, 18, 10)$. From the table, we observe that DAB outperforms DS and DB-LMA by a large margin, while DB-LMA is only slightly better than DS on average. This phenomenon indicates that the main advantage of DNN-based multichannel speech enhancement over DS is in the setting of ad-hoc microphone arrays. We also see that the STOI scores of all comparison methods drop significantly when the evaluation scenario becomes difficult, i.e. from $(2, 14, 8)$ to $(2, 18, 10)$. The performance decrease of DAB suggests that the proposed channel-reweighting algorithm, which takes all microphones into consideration, still has much room to get improved, since some microphones that are too far away from the speaker may be completely noisy.

4.4. On the effectiveness of channel-reweighting

Table 2 lists the comparison results between the DAB without channel-reweighting and that with the proposed channel-reweighting. From the table, we see that the DAB with the channel-reweighting algorithm significantly outperforms that without channel-reweighting in the babble noise environment, and slightly outperforms the latter in the factory noise environment. This phenomenon indicates that the DNN2-based channel-reweighting becomes more and more important when the environment is getting difficult.

To study how much room we can further improve beyond the proposed DAB, we pick up the “best” single-channel result among the results produced from the 4 microphones respectively for each test utterance, as well as the STOI score of the noisy utterance received by the “best” single-channel microphone. The two methods are denoted as sDAB* and noisy* respectively. Because the best microphone for each test utterance cannot be identified perfectly by DNN2, the two referenced methods may be understood as “upperbound” and “lowerbound” of DAB, instead of two realistic methods.

Table 3 lists the comparison results between DAB, sDAB* and noisy*. From the table, we see that DAB produces higher STOI scores than noisy*, which indicates the effectiveness of DAB. Comparing Table 3 with Tables 1 and 2, we see that all comparison methods including the DAB without channel-reweighting are no better than noisy*, which emphasizes the importance of the channel-reweighting. However, the proposed DAB is still worse than sDAB*, which indicates that there is still much room we can improve. Moreover, we believe that, if more advanced channel-reweighting methods are developed, DAB can outperform sDAB* significantly, which is our future work.

Table 3. Comparison results between DAB, sDAB*, and noisy*.

(a_1, a_2, b)	Noise type	DAB	sDAB*	Noisy*
$(2, 14, 8)$	Babble	0.8417	0.8782	0.8227
	Factory	0.8853	0.9140	0.8617
$(2, 18, 10)$	Babble	0.8007	0.8437	0.7818
	Factory	0.8119	0.8712	0.8025

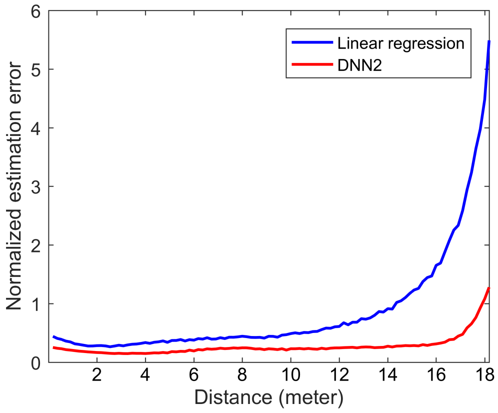


Fig. 4. Comparison of the estimation errors of channel weights between DNN2 and linear regression.

To verify that DNN2 is a valid choice of the channel-reweighting model, we compared DNN2 with linear regression. The comparison result is shown in Fig. 4, where the normalized estimation error is defined as $|p_{\text{opt}} - p|/p_{\text{opt}}$. From the figure, we see that DNN2 yields much lower estimation error than linear regression. The estimation error of linear regression quickly becomes large along with the increase of SNR, while DNN2 keeps the estimation error in a low level steadily.

5. CONCLUSIONS

In this paper, we have proposed deep ad-hoc beamforming, which is the first deep beamforming method for ad-hoc microphone arrays. The difference between DAB and deep beamforming is that DAB employs another DNN, i.e. DNN2, to produce the channel weights for a reweighted aggregation of the single-channel signals picked up by an ad-hoc microphone array. Empirically, we have found that the advantage of the deep beamforming over DNN-based single-channel speech enhancement is released to a maximum extent in the ad-hoc microphone arrays. We have also verified that channel-reweighting is important to DAB, and the DNN2-based channel-reweigting is effective. To summarize, the proposed DAB is still premature. We believe that many advanced methods can be developed to fully mine the potential of deep learning based multi-channel speech enhancement in ad-hoc microphone arrays.

6. REFERENCES

[1] DeLiang Wang and Jitong Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[2] Yi Jiang, DeLiang Wang, RunSheng Liu, and ZhenMing Feng, "Binaural classification for reverberant speech segregation using deep neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 2112–2121, 2014.

[3] Jahn Heymann, Lukas Drude, and Reinhold Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *Proc. ICASSP*, 2016, pp. 196–200.

[4] Takuya Higuchi, Nobutaka Ito, Takuya Yoshioka, and Tomohiro Nakatani, "Robust mvdr beamforming using time-frequency masks for online/offline asr in noise," in *Proc. ICASSP*, 2016, pp. 5210–5214.

[5] Hakan Erdogan, John R Hershey, Shinji Watanabe, Michael I Mandel, and Jonathan Le Roux, "Improved mvdr beamforming using single-channel mask prediction networks.," in *Proc. Interspeech*, 2016, pp. 1981–1985.

[6] Bo Li, Tara N Sainath, Ron J Weiss, Kevin W Wilson, and Michiel Bacchiani, "Neural network adaptive beamforming for robust multichannel speech recognition.," in *Proc. Interspeech*, 2016, pp. 1976–1980.

[7] Lukas Pfeifenberger, Matthias Zöhrer, and Franz Pernkopf, "Dnn-based speech mask estimation for eigenvector beamforming," in *Proc. ICASSP*, 2017, pp. 66–70.

[8] Suliang Bu, Yunxin Zhao, Meiyuh Hwang, and Sining Sun, "A probability weighted beamformer for noise robust asr," *Proc. Interspeech*, pp. 3048–3052, 2018.

[9] Zhong-Qiu Wang and DeLiang Wang, "On spatial features for supervised speech separation and its application to beamforming and robust asr," in *Proc. ICASSP*, 2018, pp. 5709–5713.

[10] Zhong-Qiu Wang and DeLiang Wang, "All-neural multichannel speech enhancement," *Proc. Interspeech*, pp. 3234–3238, 2018.

[11] Xiong Xiao, Shengkui Zhao, Douglas L Jones, Eng Siong Chng, and Haizhou Li, "On time-frequency mask estimation for mvdr beamforming with application in robust speech recognition," in *Proc. ICASSP*, 2017, pp. 3246–3250.

[12] Yan-Hui Tu, Jun Du, Lei Sun, and Chin-Hui Lee, "Lstm-based iterative mask estimation and post-processing for multichannel speech enhancement," in *Proc. APSIPA ASC*, 2017, pp. 488–491.

[13] Takuya Higuchi, Keisuke Kinoshita, Nobutaka Ito, Shigeki Karita, and Tomohiro Nakatani, "Frame-by-frame closed-form update for mask-based adaptive mvdr beamforming," in *Proc. ICASSP*, 2018, pp. 531–535.

[14] Ying Zhou and Yanmin Qian, "Robust mask estimation by integrating neural network-based and clustering-based approaches for adaptive acoustic beamforming," in *Proc. ICASSP*, 2018, pp. 536–540.

[15] Tomohiro Nakatani, Nobutaka Ito, Takuya Higuchi, Shoko Araki, and Keisuke Kinoshita, "Integrating dnn-based and spatial clustering-based mask estimation for robust mvdr beamforming," in *Proc. ICASSP*, 2017, pp. 286–290.

[16] Xueliang Zhang, Zhong-Qiu Wang, and DeLiang Wang, "A speech enhancement algorithm by iterating single-and multi-microphone processing and its application to robust asr," in *Proc. ICASSP*, 2017, pp. 276–280.

[17] Yuxuan Wang and DeLiang Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 7, pp. 1381–1390, 2013.

[18] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 1, pp. 7–19, 2015.

[19] Richard Heusdens, Guoqiang Zhang, Richard C Hendriks, Yuan Zeng, and W Bastiaan Kleijn, "Distributed mvdr beamforming for (wireless) microphone networks using message passing," in *Proc. IWAENC*, 2012, pp. 1–4.

[20] Yuan Zeng and Richard C Hendriks, "Distributed delay and sum beamformer for speech enhancement via randomized gossip," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 1, pp. 260–273, 2014.

[21] Matt O'Connor, W Bastiaan Kleijn, and Thushara Abhayapala, "Distributed sparse mvdr beamforming using the bi-alternating direction method of multipliers," in *Proc. ICASSP*, 2016, pp. 106–110.

[22] Matt O'Connor and W Bastiaan Kleijn, "Diffusion-based distributed mvdr beamformer," in *Proc. ICASSP*, 2014, pp. 810–814.

[23] Vincent Mohammad Tavakoli, Jesper Rindom Jensen, Mads Græsbøll Christensen, and Jacob Benesty, "A framework for speech enhancement with ad hoc microphone arrays," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 6, pp. 1038–1051, 2016.

[24] Suhanya Jayaprakasam, Sharul Kamal Abdul Rahim, and Chee Yen Leow, "Distributed and collaborative beamforming in wireless sensor networks: Classifications, trends, and research directions," *IEEE Commun. Surveys & Tutorials*, vol. 19, no. 4, pp. 2092–2116, 2017.

[25] Vincent M Tavakoli, Jesper R Jensen, Richard Heusdens, Jacob Benesty, and Mads G Christensen, "Distributed max-sinr speech enhancement with ad hoc microphone arrays," in *Proc. ICASSP*, 2017, pp. 151–155.

[26] Jie Zhang, Sundeep Prabhakar Chepuri, Richard Christian Hendriks, and Richard Heusdens, "Microphone subset selection for mvdr beamformer based noise reduction," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 3, pp. 550–563, 2018.

[27] Andreas I Koutrouvelis, Thomas W Sherson, Richard Heusdens, and Richard C Hendriks, "A low-cost robust distributed linearly constrained beamformer for wireless acoustic sensor networks with arbitrary topology," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 8, pp. 1434–1448, 2018.