FAST NOISE COMPENSATION FOR SPEECH SEPARATION IN DIFFUSE NOISE

Rong Hu and Yunxin Zhao

Department of Computer Science
University of Missouri-Columbia, MO 65211 USA
rhq2c@mizzou.edu zhaoy@missouri.edu

ABSTRACT

In this paper, a fast noise compensation (FNC) algorithm is proposed for the adaptive decorrelation filtering (ADF) speech separation system in the presence of diffuse noise. The adaptation of ADF is a dynamic process, making noise effects at ADF outputs time-varying in nature. Such changing noise effects need to be tracked and adaptively removed. Under the assumption that acoustic paths are slow in change and utilizing the filtering structure of the separation model, noise compensation terms were adapted with an FFT-based fast algorithm. Experiments were based on both simulated and real recorded diffuse noises. Strong diffuse noise distracts part of the attention of ADF to do noise cancellation while separating speech, and FNC works by forcing ADF to stay focused on speech separation task. The proposed algorithm significantly improved the separation performance of ADF system in diffuse noise.

1. INTRODUCTION

Diffuse noise and interfering speech present double folds of challenges for hands-free automatic speech recognition (ASR) and speech communication. It is important to address effects of noise in speech separation for practical applications of blind source separation (BSS) and independent component analysis (ICA) techniques. Although general ideas in dealing with noisy mixtures are cast as bias removal [1], the compensation techniques and their effectiveness depend heavily on specific separation models and application scenarios. Currently, most studies are either of theoretical nature, or focus on easy noise conditions (e.g. [2]) or simplified mixing models (e.g. [3]). In [4], noise subtraction was used in both cases of sensor and real diffuse noise for convolutive BSS. However, research efforts along this line are still insufficient, and further development of effective online compensation algorithms are needed for convolutive speech mixtures in real environment.

In our previous work [5, 6], the separation model of adaptive decorrelation filtering (ADF) [7, 8] was significantly enhanced for speech mixtures in acceleration of convergence rate and reduction of steady-state filter estimation errors. For speech mixtures contaminated by white uncorrelated noises, a simple noise-adapted ADF algorithm [2] was proposed based on a time domain vector formulation. In practice, diffuse noises are not white and are highly correlated in low frequency. In [9], it was shown that correlated noise deteriorated performance more severely, and a subspace based noise reduction front-end was experimented to improve the working condition for speech separation, rather than compensating ADF algorithm itself. The objective of this paper is to extend the technique of noise-compensated ADF (NC-ADF) [2] to speech mixtures in diffuse noise and to derive a fast online algorithm for real-time applications.

2. ADF MODEL IN NOISE

The ADF noisy speech mixture separation system with filters, $\mathbf{g}_{ij} = [g_{ij}(0), ..., g_{ij}(N-1)]^T$, $(i, j{=}1, 2, i \neq j)$, is shown in Figure 1. We formulate the I/O relations of ADF as [2]

$$\mathbf{v}_n = \mathbf{G}(\tilde{\mathbf{y}} + \tilde{\mathbf{n}}), \tag{1}$$

where $\tilde{\mathbf{y}} = [\tilde{\mathbf{y}}_1^T(t), \tilde{\mathbf{y}}_2^T(t)]^T$ and $\tilde{\mathbf{n}} = [\tilde{\mathbf{n}}_1^T(t), \tilde{\mathbf{n}}_2^T(t)]^T$ are $(4N-2) \times 1$ vectors of clean mixture and noise respectively, with $\tilde{\mathbf{n}}_i = [n_i(t), \ldots, n_i(t{-}2N{+}2)]^T$, $\tilde{\mathbf{y}}_i = [y_i(t), \ldots, y_i(t{-}2N+2)]^T$, $(i = 1, 2)$, and the filter matrix

$$\mathbf{G} = \begin{bmatrix} [\mathbf{I}_N \;\; \mathbf{0}_{N\times(N-1)}] & \mathbf{G}_{12} \\ \mathbf{G}_{21} & [\mathbf{I}_N \;\; \mathbf{0}_{N\times(N-1)}] \end{bmatrix}, \tag{2}$$

is $2N \times (4N-2)$ with

$$\mathbf{G}_{ij} = \mathbf{Toeplitz}\left([g_{ij}(0), \mathbf{0}_{1\times(N-1)}]^T, [\mathbf{g}_{ij}^T, \mathbf{0}_{1\times(N-1)}]\right). \tag{3}$$

The output noise effects are described by $\mathbf{R}_{\mathbf{v}_n\mathbf{v}_n}$=$\mathbf{R}_{\mathbf{vv}}$+$\mathbf{R}_{\boldsymbol{\eta\eta}}$, where the $2N \times 1$ speech-only output $\mathbf{v} = [\mathbf{v}_1^T(t), \mathbf{v}_2^T(t)]^T$ and noise component $\boldsymbol{\eta} = [\boldsymbol{\eta}_1^T(t), \boldsymbol{\eta}_2^T(t)]^T$ all satisfy the correlation vector I/O relations, whose speech-only version is

$$\mathbf{r}_{v_i\mathbf{v}_j} = \mathbf{r}_{y_i\mathbf{y}_j} - \mathbf{G}_{ji}\mathbf{r}_{y_i\tilde{\mathbf{y}}_i} - \mathbf{R}_{\mathbf{y}_j\mathbf{y}_j}\mathbf{g}_{ij} + \mathbf{G}_{ji}\mathbf{R}_{\tilde{\mathbf{y}}_i\mathbf{y}_j}\mathbf{g}_{ij}, \tag{4}$$

$$\mathbf{r}_{v_i\mathbf{v}_i} = \mathbf{r}_{y_i\mathbf{y}_i} - \mathbf{G}_{ij}\mathbf{r}_{y_i\tilde{\mathbf{y}}_j} - \mathbf{R}_{\mathbf{y}_i\mathbf{y}_j}\mathbf{g}_{ij} + \mathbf{G}_{ij}\mathbf{R}_{\tilde{\mathbf{y}}_j\mathbf{y}_j}\mathbf{g}_{ij}. \tag{5}$$

In [2], based on minimization of the decorrelation objective functions $J_{ij} = \frac{1}{2}\mathbf{r}_{v_i\mathbf{v}_j}^T\mathbf{r}_{v_i\mathbf{v}_j}$, the positive definitive assumption of $\mathbf{R}_{\mathbf{y}_j\mathbf{v}_j}$, and approximation of cross correlation vectors $\mathbf{r}_{v_i\mathbf{v}_j}$ by instantaneous samples $v_i(t)\mathbf{v}_j(t)$, the basic

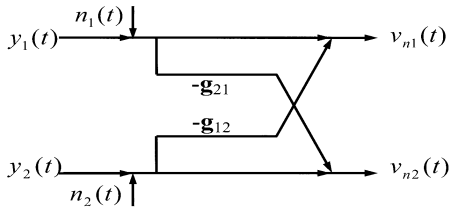

Fig. 1. ADF system for noisy speech mixtures

ADF algorithm for clean mixtures [7] was derived as

$$\mathbf{g}_{ij}(t) = \mathbf{g}_{ij}(t-1) + \mu(t) v_i(t) \mathbf{v}_j(t), \tag{6}$$

where the step size $\mu(t)$ can be chosen based on convergence analysis [8], or be combined with variable step size (VSS) schemes to accelerate convergence and reduce error (e.g., [5]).

3. NOISE COMPENSATION FOR ADF

From the noise-component version of relation (4), the estimate of noise contribution to ADF output cross-correlation is

$$\hat{\mathbf{r}}_{\eta_i \boldsymbol{\eta}_j} = \hat{\mathbf{r}}_{n_i \mathbf{n}_j} - \mathbf{G}_{ji} \hat{\mathbf{r}}_{n_i \tilde{\mathbf{n}}_i} - \hat{\mathbf{R}}_{\mathbf{n}_j \mathbf{n}_j} \mathbf{g}_{ij} + \mathbf{G}_{ji} \hat{\mathbf{R}}_{\tilde{\mathbf{n}}_i \mathbf{n}_j} \mathbf{g}_{ij}. \tag{7}$$

As filters $\mathbf{g}_{ij}$ evolve, correlation characteristics of output noise also changes, even when input noises are stationary. Compensation can be done by removing noise bias from the noisy objective function $J_{n_{ij}} = \frac{1}{2} \mathbf{r}^T_{v_{\mathbf{n}_i} \mathbf{v}_{n_j}} \mathbf{r}_{v_{\mathbf{n}_i} \mathbf{v}_{n_j}}$. This corresponds to subtracting the cross-correlation vector estimate $\hat{\mathbf{r}}_{\eta_i \boldsymbol{\eta}_j}(t)$ from ADF adaptation as

$$\mathbf{g}_{ij}(t) = \mathbf{g}_{ij}(t-1) + \mu(t)(v_{ni}(t) \mathbf{v}_{nj}(t) - \hat{\mathbf{r}}_{\eta_i \boldsymbol{\eta}_j}(t)) \tag{8}$$

The difference from [2] is that here we do not simplify (7) by the assumption of white sensor noise (uncorrelated), but rather consider real diffuse noises with strong correlations.

Due to non-stationarity of speech, individual competing sources are not equally excited at short time intervals. In [5], an analysis was made on the effect of unequal excitation of sources on ADF estimation error and a step-size scaling technique was proposed by using unequal discount factors controlled by short-term output powers. It can be interpreted as a kind of soft-decision switch that senses the relative strength of individual sources. However, such a method is noise-sensitive because noise-corrupted estimate of source strengths will lead to inaccurate computation of VSS. Therefore, noise compensation should also be performed for VSS.

Here we use the error-reducing VSS similar to [5] that balances adaptation between unequally excited sources

$$\mu_{ij}(t) = \mu(t) \cdot \hat{\sigma}^2_{v_j}(t) / \hat{\sigma}^2_{av}(t), \tag{9}$$

where the normalizing gain factor $\mu(t)$ was given by [8]

$$\mu(t) = \gamma / \big(N(\sigma^2_{y_{n1}}(t) + \sigma^2_{y_{n2}}(t)) \big), \tag{10}$$

with $\sigma^2_{y_{ni}}(t)$ the short-term power of the i-th input. The average noise-free output power $\hat{\sigma}^2_{av}(t)$ is

$$\hat{\sigma}^2_{av}(t) = \big(\hat{\sigma}^2_{v_1}(t) + \hat{\sigma}^2_{v_2}(t) \big) / 2 \tag{11}$$

The VSS compensation is done by subtracting noise power from that of noisy ADF output

$$\hat{\sigma}^2_{v_j} = \hat{r}_{v_j}(0) = \hat{r}_{v_{n_j}}(0) - \hat{r}_{\eta_j}(0), \tag{12}$$

with the output noise power estimated as

$$\hat{r}_{\eta_j}(0) = \hat{r}_{n_j}(0) - 2 \mathbf{g}^T_{ji} \hat{\mathbf{r}}_{n_j \mathbf{n}_i} + \mathbf{g}^T_{ji} \mathbf{R}_{\mathbf{n}_i \mathbf{n}_i} \mathbf{g}_{ji}, \tag{13}$$

which follows from the I/O relation (5) for noise.

4. FAST UPDATE OF COMPENSATION TERMS

The noise compensation terms derived above require matrix-vector multiplications and are not suitable for real time implementation. To speed up the NC-ADF, we first reduce the update rate for compensation terms. Then, FFT-based computations of noise cross-correlation vectors are applied, utilizing the Toeplitz structure of correlation and system matrices.

Observations on the adaptation procedure show that the change of ADF filters is small within short time intervals (e.g., $< 30ms$). In short-term, ADF parameters can be regarded as fixed. This approximation makes it possible to update compensation terms for only every K time samples.

The output bias estimate (7) can be written as

$$\hat{\mathbf{r}}_{\eta_i \boldsymbol{\eta}_j} = \hat{\mathbf{r}}_{n_i \mathbf{n}_j} - \mathbf{a}_{ij} - \mathbf{b}_{ij} + \mathbf{c}_{ij}, \tag{14}$$

with

$$\mathbf{a}_{ij} = \mathbf{G}_{ji} \hat{\mathbf{r}}_{n_i \tilde{\mathbf{n}}_i}, \tag{15}$$

$$\mathbf{b}_{ij} = \hat{\mathbf{R}}_{\mathbf{n}_j \mathbf{n}_j} \mathbf{g}_{ij}, \tag{16}$$

$$\mathbf{c}_{ij} = \mathbf{G}_{ji} \tilde{\mathbf{d}}_{ij}, \tag{17}$$

$$\tilde{\mathbf{d}}_{ij} = \hat{\mathbf{R}}_{\tilde{\mathbf{n}}_i \mathbf{n}_j} \mathbf{g}_{ij}. \tag{18}$$

The fast computation of $\mathbf{a}_{ij}$ and $\mathbf{c}_{ij}$ share the same structure. Applying basic algebra to Toeplitz matrix $\mathbf{G}_{ji}$, each component, $a_{ij}(k), k = 0, ..., N-1$, of vector $\mathbf{a}_{ij}$ can be expressed as the linear convolution (denoted as $*$)

$$a_{ij}(k) = g_{ji}(n) * \tilde{\xi}^a_{ij}(n)|_{n=2N-2-k}, \tag{19}$$

where the $(2N-1)$-point sequence

$$\tilde{\xi}^a_{ij}(n) = \tilde{r}_{n_i n_i}(2N-2-n). \tag{20}$$

Similarly, components of $\mathbf{c}_{ij}$ are obtained by

$$c_{ij}(k) = g_{ji}(n) * \tilde{\xi}^c_{ij}(n)|_{n=2N-2-k}, \tag{21}$$

$$\tilde{\xi}^c_{ij}(n) = \tilde{d}_{ij}(2N-2-n). \tag{22}$$

The vectors $\mathbf{b}_{ij}$ and $\tilde{\mathbf{d}}_{ij}$ also have similar structure

$$b_{ij}(k) = g_{ij}(n) * \tilde{\xi}^b_{ij}(n)|_{n=k+N-1}, \tag{23}$$

$$\tilde{\xi}^b_{ij}(n) = \tilde{r}_{n_j n_j}(|n-N+1|), \tag{24}$$

$$\tilde{d}_{ij}(k) = g_{ij}(n) * \tilde{\xi}^d_{ij}(n)|_{n=k+N-1}, \tag{25}$$

$$\tilde{\xi}^d_{ij}(n) = \tilde{r}_{n_i n_j}(N-1-n). \tag{26}$$

The only exception is that, unlike other N-point sequences $a_{ij}(k)$, $b_{ij}(k)$, and $c_{ij}(k)$, the length of $\tilde{d}_{ij}(k)$ in (25) is $2N-1$. All of them can be computed using fast convolutions based on N_F-point FFTs ($N_F > 2N-1$). Specially, $\tilde{d}_{ij}(k)$ can actually be decomposed into two N-point sub-sequences and be computed with two N_F-point FFT-IFFT modules. In this way, every sequence only needs to be zero-padded to length N_F, because only N-point result sequences are required in each module. Other points with aliasing are discarded.

In practical implementation, DC-components of these vectors are removed. Triangular windows of length N, $w(n) = (N-n)/N, n = 0, ..., N-1$, are also applied on both compensation vector estimates and ADF adaptation vectors.

From (12), (13), and (16), the noise-free ADF output powers used in VSS compensation are estimated by

$$\hat{\sigma}^2_{v_j} \approx \mathbf{v}^T_{n_j}\mathbf{v}_{n_j}/N - 2\mathbf{g}^T_{ji}\hat{\mathbf{r}}_{n_j\mathbf{n}_i} + \mathbf{g}^T_{ji}\mathbf{b}_{ji}. \quad (27)$$

For the case when K=N and FFT length N_F=$2N$, the computation of $2N$-point FFTs can be distributed to the time interval of length N. The complexity of fast noise compensation (FNC) is $O(N+\log N)$ for each time sample, as compared with the complexity $O(N^2)$ of direct NC.

5. EXPERIMENTS

5.1. Experimental Data and Setup

Speech mixtures were generated from clean sources in TIMIT database and real acoustic impulse responses measured in a room with reverberation time $T_{[60]}$=$0.3sec$ from two microphones (13 and 15) in a circular microphone array of radius 15cm [10]. The target speech had 40 sentences from 4 speakers (faks0, felc0, mdab0, mreb0) approximately $2m$ away, from the microphones.

Both simulated and real recorded diffuse noises were tested. For simulated case, noise signals were designed to be speech-shaped by the procedure

$$n_1(t) = \beta_1 \sum_{k=1}^{P_1} a_k^{(1)} n_1(t-k) + (1-\beta_1)n_2(t) + \varepsilon_1(t), \quad (28)$$

$$n_2(t) = \beta_2 \sum_{k=1}^{P_2} a_k^{(2)} n_2(t-k) + (1-\beta_2)n_1(t) + \varepsilon_2(t), \quad (29)$$

where $\varepsilon_i(t)$'s are white Gaussian excitations, β_1=0.65, β_2=0.6, P_1=2, P_2=3, and $a_k^{(i)}$'s are linear prediction coefficients (LPC) estimated from clean TIMIT data. Real diffuse noises were recorded in a computer lab with a pair of omnidirectional stereo microphones placed $15cm$ apart on a conference table in the middle of the lab, where the air-conditioning and ventilation system and 8 desktop workstations were working simultaneously. With stationary or slow-changing assumption, we use the 5-second segment of noise-only data preceding the 1st speech sentence to estimate properties of input noise. Figure 2 illustrates the estimate of cross power spectra for both types of noise.

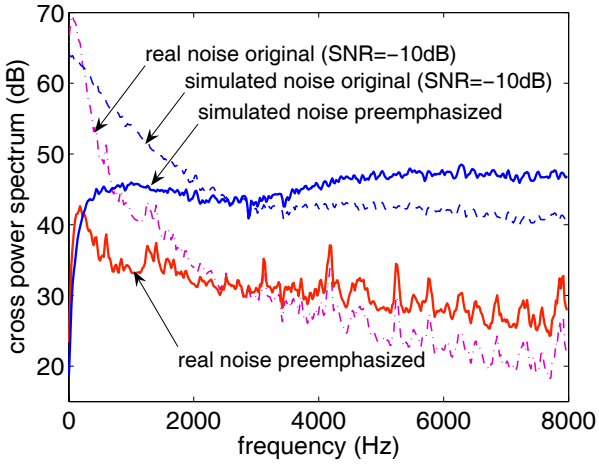

Fig. 2. Cross power spectrum of diffuse noise

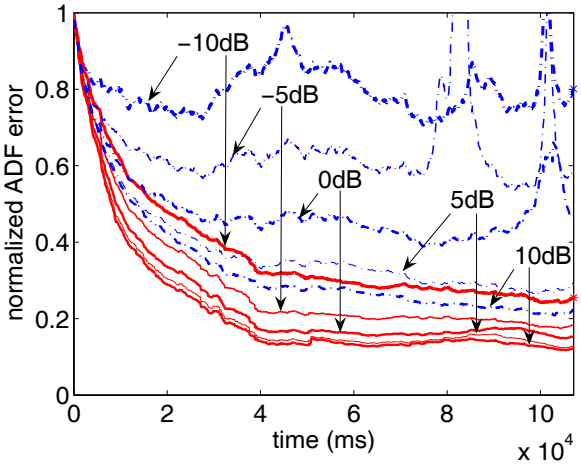

Fig. 3. Comparison of convergence performance in real noise (dash-dot: baseline ADF; solid: FNC-ADF)

The basic setup for ADF were N=400 and γ=0.01. Preemphasis ($1\text{-}z^{-1}$) was applied to mixtures to remove the 6-dB/octave tilt of speech long-term spectrum and reduce eigenvalue disparity to achieve better convergence [6]. As a common technique in ASR, preemphasis enhances perceptually important speech components. It also alters noise properties at ADF input. In fact, the simulated speech-shaped noise spectrum was flattened resulting in a loss of signal-to-noise ratio (SNR) of approximately 3dB; preemphasis on recorded diffuse noise retained a significant amount of coloration and correlation, and it increased SNR by 12dB while suppressing strongly correlated low frequency components (see Figure 2). In the following, we evaluate SNR and target-to-interference-ratio (TIR) results for preemphasized components, ignoring perceptually unimportant components. For FNC-ADF, the NC vector update rate was reduced by the factor of K=N, FFT length (N_F) was 1024. VSS was not applied to baseline

original SNR(dB)	preemphsized SNR(dB)	baseline (v_1, v_2)	FNC-ADF (v_1, v_2)
3	$(0.2, -1.3)$	$(1.7, 2.1)$	$(7.6, 8.6)$
9	$(6.2, 4.7)$	$(3.0, 3.9)$	$(9.7, 10.0)$
15	$(12.2, 10.7)$	$(4.7, 5.6)$	$(10.8, 10.5)$
21	$(18.2, 16.7)$	$(6.3, 6.8)$	$(11.3, 10.7)$
27	$(24.2, 22.7)$	$(7.5, 7.6)$	$(11.5, 10.7)$

Table 1. Gain in TIR (dB) (simulated speech-shaped noise)

original SNR(dB)	preemphsized SNR(dB)	baseline (v_{n_1}, v_{n_2})	FNC-ADF (v_{n_1}, v_{n_2})
3	$(0.2, -1.3)$	$(-0.3, -1.5)$	$(-1.4, -3.6)$
9	$(6.2, 4.7)$	$(5.3, 3.4)$	$(4.5, 2.4)$
15	$(12.2, 10.7)$	$(10.8, 8.6)$	$(10.3, 8.3)$
21	$(18.2, 16.7)$	$(16.3, 14.0)$	$(16.3, 14.3)$
27	$(24.2, 22.7)$	$(22.0, 19.7)$	$(22.2, 20.3)$

Table 2. Output SNR (dB) (simulated speech-shaped noise)

original SNR(dB)	preemphsized SNR(dB)	baseline (v_1, v_2)	FNC-ADF (v_1, v_2)
-12	$(0.2, 0.3)$	$(3.1, 3.9)$	$(7.5, 8.7)$
-6	$(6.2, 6.3)$	$(4.2, 5.6)$	$(9.9, 10.1)$
0	$(12.2, 12.3)$	$(6.3, 7.7)$	$(10.9, 10.6)$
6	$(18.2, 18.3)$	$(7.7, 7.9)$	$(11.4, 10.9)$
12	$(24.2, 24.3)$	$(8.1, 8.1)$	$(11.6, 10.9)$

Table 3. Gain in TIR (dB) (real diffuse noise)

original SNR(dB)	preemphsized SNR(dB)	baseline (v_{n_1}, v_{n_2})	FNC-ADF (v_{n_1}, v_{n_2})
-12	$(0.2, 0.3)$	$(3.8, 2.4)$	$(0.2, -1.8)$
-6	$(6.2, 6.3)$	$(6.1, 5.9)$	$(6.2, 4.1)$
0	$(12.2, 12.3)$	$(12.3, 11.4)$	$(12.1, 10.0)$
6	$(18.2, 18.3)$	$(17.9, 16.4)$	$(18.1, 15.9)$
12	$(24.2, 24.3)$	$(23.4, 21.7)$	$(24.0, 21.8)$

Table 4. Output SNR (dB) (real diffuse noise)

ADF without noise compensation since it degraded performance for noisy mixtures.

5.2. Convergence and Separation Performance

ADF filter estimation error during adaptation was measured to quantify convergence performance. Using FNC-ADF, the steady-state error was significantly reduced compared with baseline AFD. Figure 3 gives an example for the case of real recorded lab noise. It can be seen that, as noise levels increase, the advantage of FNC-ADF over baseline becomes more significant.

Separation performances are evaluated by the gains in TIR, defined as $\text{TIR}_{output} - \text{TIR}_{input}$. In Tables 1 and 3, the TIR gains of FNC-ADF outperform those of the baseline for both types of noises, at the cost of only slight decrease in SNR, as shown in Tables 2 and 4. It is interesting to observe that at severe conditions, e.g., for SNR=-12 dB (original), baseline ADF actually increased SNR. This is consistant with the analysis in [9] that in correlated noise, baseline ADF tends to cancel out some noise. Tables 2 and 4 tell us that FNC algorithm can force ADF to focus on separation, rather than noise cancellation. Examination of ADF filter impulse responses also verified this explanation.

6. CONCLUSIONS

The FNC algorithm significantly improved the separation performance of ADF for speech mixtures in diffuse noise. Future work will include tests to incorporate recursive tracking methods for non-stationary noises, and to apply noise reduction post filtering methods to current technique for the improvement of both system TIR and SNR.

7. REFERENCES

[1] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent component analysis*, John Wiley, 2001.

[2] R. Hu and Y. Zhao, "Adaptive decorrelation filtering algorithm for speech source separation in uncorrelated noises," in *ICASSP*, 2005, vol. I, pp. 1113–1116.

[3] R. Balan, J. Rosca, and S. Richard, "Scalable non-square blind separation in the presence of noise," in *ICASSP*, 2003, vol. 5, pp. 293–296.

[4] R. Aichner, H. Buchner, and W. Kellermann, "Convolutive blind source separation for noisy mixtures," in *Proc. CFA/DAGA*, 2004, http://www.lnt.de/LMS/publications/web/lnt2004_2.pdf.

[5] R. Hu and Y. Zhao, "Variable step size adaptive decorrelation filtering for competing speech separation," in *Eurospeech'05*, Sept. 2005, vol. I, pp. 2297–2300.

[6] Y. Zhao, R. Hu, and X. Li, "Speedup convergence and reduce noise for enhanced speech separation and recognition," *IEEE Trans. SAP*, to be published, July, 2006.

[7] E. Weinstein, M. Feder, and A. V. Oppenheim, "Multichannel signal separation by decorrelation," *IEEE Trans. SP*, vol. 43, pp. 405–413, Oct. 1993.

[8] K. Yen and Y. Zhao, "Adaptive co-channel speech separation and recognition," *IEEE Trans. SAP*, vol. 7, pp. 138–151, 1999.

[9] K. Yen, J. Huang, and Y. Zhao, "Co-channel speech separation in the presence of correlated and uncorrelated noises," in *ESCA Eurospeech'99*, 1999, pp. 2587–2589.

[10] "RWCP sound scene database in real acoustic environments," ATR Spoken Language Translation Research Lab, Japan, 2001.