社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
SP2005-167 (2006-03)

Implementation of Steady-State Suppression Using a Digital Signal Processor for Real-Time Processing:

Evaluation of the Processing in an Actual Hall

Kei TAKAHASHI†, Takahito GOTO†, Fumihiro TADOKORO†, Keiichi YASU† and Takayuki ARAI†

†Department of Electrical and Electronics Engineering, Sophia University, 7-1 Kioi-chou, Chiyoda-ku, Tokyo, 102-0094, Japan

E-mail: †kei-taka@sophia.ac.jp

Abstract In an actual hall, reverberation degrades speech intelligibility, which is the result of overlap-masking occurred when segments of an acoustic signal are affected by reverberation components of previous segments. Arai *et al*. (2001, 2002) have been proposed a pre-processing technique which suppresses steady-state portion of speech in order to prevent the result of overlap-masking. However, the originally proposed technique is not suitable for real-time processing. Therefore, Arai *et al*. (2003) suggested alternative technique based on the First Fourier Transform (FFT) with cepstral analysis in order to implement real-time processing.

In this study, we successfully implemented the real-time processing of steady-state suppression based on the FFT with a digital signal processor (DSP) (Goto et al., 2005). Also, the effectiveness of the real-time processing implemented with a DSP was evaluated for younger as well as elderly people in an actual reverberant environment.

Keywords Real-time processing, Digital signal processor, Reverberation, Speech enhancement, Steady-state suppression

1. Introduction

One of the major reasons reverberation degrades speech intelligibility in an actual hall is overlap-masking, which occurs when reverberant tails of previous portions of a sound affect subsequent segments [1]. Figure 1 shows overlap-masking for the word “October”, with the left panel showing the original speech signal with no reverberation and the right panel showing the corresponding reverberant signals. The reverberant signals were obtained by convoluting the original speech signals with an impulse response signal for a room having a 1.1 s reverberation time. Consonants that have weak energy, such as /k/, /t/, /b/, are masked by reverberation tails that have strong energy. This effect of overlap-masking degrades speech intelligibility in a reverberant environment.

To reduce the effect of overlap-masking, Arai *et al.* proposed the pre-processing technique known as steady-state suppression; a technique that improves speech intelligibility [2, 3]. In this technique, pre-processing suppresses a speech signal before it is affected by reverberation, reducing the influence of reverberation on the transmission path. In particular, pre-processing suppresses the steady-state portions of speech that have more energy but which are less crucial for speech perception in order to reduce the masking influence caused by the reverberation components of the previous portions. Arai *et al*. has proposed two ways to realize steady-state suppression to date [2, 3, 4]: the filter-bank method (FB method), which extracts the spectral envelope using a filter-bank to estimate the steady-state portion of speech, and the FFT method, which uses cepstral analysis to estimate the spectral transition. Both methods are briefly described in the following sections.

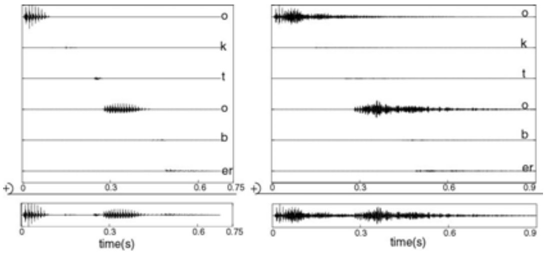
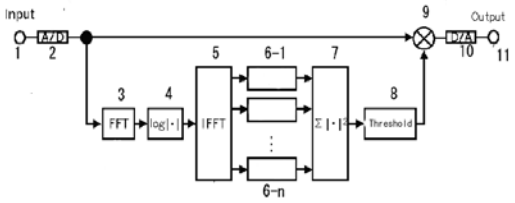


Figure 1: Overlap-masking for the word “October” (from [5])

1.1 Filter-bank method

The original method of steady-state suppression was implemented by a filter bank [2, 3]. First, an original signal was split into 1/3-octave bands. In each band the temporal envelope was extracted. After down-sampling, regression coefficients were calculated from the five adjacent values of the time trajectory of the logarithmic envelope of a sub-band. Then the mean square of the regression coefficients, *D*, was calculated. We used Furui’s *D* parameter to measure the spectral transition [6]. After up-sampling, we defined a speech portion as a steady state where *D* was less than a given threshold. Once a portion was considered as a steady state, the amplitude of the portion was multiplied by the factor of 0.4 (a suppression rate of 40%). We used this suppression rate because previous studies confirmed it was a reasonable degree of suppression [4] when the reverberation time was 1.1 s.

1.2 FFT method



The original FB method is not so applicable to digital signal processing for real-time processing. Therefore, Arai *et al.* [4] proposed an alternative technique based on the Fast Fourier Transform (FFT) with cepstral analysis for real-time processing of steady-state suppression**.** Figure 2 shows the block diagram of FFT-based steady-state suppression. First, speech data was converted into a digital signal by an A/D converter with a sampling frequency of 16 kHz. The speech signal was divided into frames with a Hamming window. A single frame length was 20 ms and the frame shift was 10 ms. Thus, half of the signal in a frame was overlapped by half of a signal in adjacent frames. In the next step, FFT was taken for each frame signal and the logarithmic spectrum was calculated. Then, an inverse FFT was carried out on the obtained logarithmic spectrum to give the resulting cepstral coefficients. Generally, lower-order cepstral coefficients correspond to the spectral envelope in the frequency domain. Therefore, we used cepstral coefficients up to 30 to obtain the spectral envelope. Thereafter, the regression coefficient, or delta coefficient, was calculated for five continuous frames of cepstral coefficients which were in the same quefrency, as shown in Figure 3. We refer to this signal array as the “cepstral matrix”. In the next step, the mean square *D* of the regression coefficients obtained from a matrix was calculated for each frame. The *D* parameter is the same as Furui proposed in [6], and we used it to measure spectral transition as in the FB method. Once a threshold is set to a given value, we can automatically identify the steady-state portion of speech by comparing the value of *D* and the threshold. In a preliminary experiment, we set the threshold as 0.052, which we confirmed as a reasonable value. When the value of *D* was less than the threshold, the portion of speech was recognized as steady state and the amplitude of speech in this portion was suppressed 40%. Finally, processed signal was output using a D/A converter.

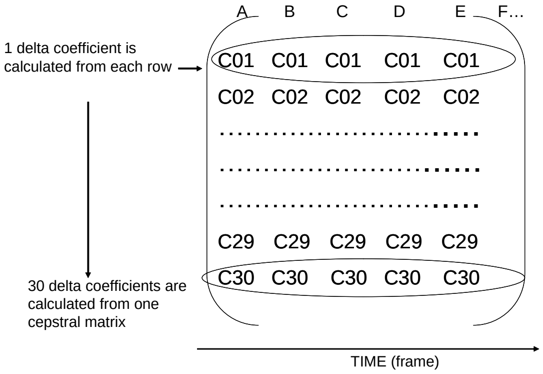

A
B
C
D
E
F...
1 delta coefficient is calculated from each row
C01 C01 C01 C01 C01
C02 C02 C02 C02 C02
C29 C29 C29 C29 C29
C30 C30 C30 C30 C30
30 delta coefficients are calculated from one cepstral matrix
TIME (frame)

Figure 3: Calculation flow to obtain the regression coefficient

We applied the FFT method for real-time processing of steady-state suppression with a digital signal processor (DSP) since the FFT-based method estimates spectral envelopes faster than the FB method.

2. Implementation of real-time processing (DSP method)

2.1 Changes required to implement the DSP method

Due to technical problems, we needed to change several points in order to implement the FFT method in a DSP for real-time processing. The two major problems were cepstral matrix shifting and sampling frequency. The most crucial change was to compute the cepstral matrix among five continuous frames of speech. In the FFT method shown in Figure 3, the matrix consisted of frames A to E, and the next matrix consisted of frames B to F; frames B to E were referred to again within this matrix shift. However, in the DSP method, we needed to shift the matrix from A-E to F-J with no overlap to minimize DSP load. Further, as the sampling frequency of DSP was 8 kHz, which is half of the original FFT method, cepstral coefficients were used up to 15 while we used up to 30 in the FFT method. We set the threshold 0.025 for the DSP method.

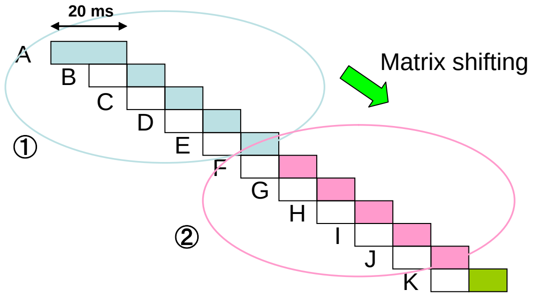


Figure 4: **Cepstral matrix shifting for the DSP method**

2.2 Hardware Setting

We required effective data buffering for real-time processing, because input data as well as output data needed to be continuous.

In other words, we needed a running computation thread while data was exchanged between input and output. For this reason, as shown in Figure 5, double buffering, or "ping-pong buffering", was considered reasonable to meet this need.

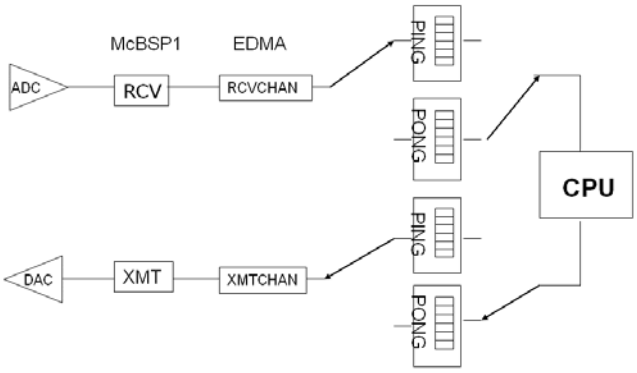


Figure 5: Schematic of ping-pong buffering

In this case, 60 ms is allowed for processing time because the length of the data buffer for both input and output is 480 samples with 8 kHz sampling frequency. Basically, the data buffer will be full within 60 ms and, while data is exchanged, the CPU can complete signal processing within 60 ms. Therefore, it takes 120 ms to output processed signals. Figure 6 shows the double buffering timing applied in this system.

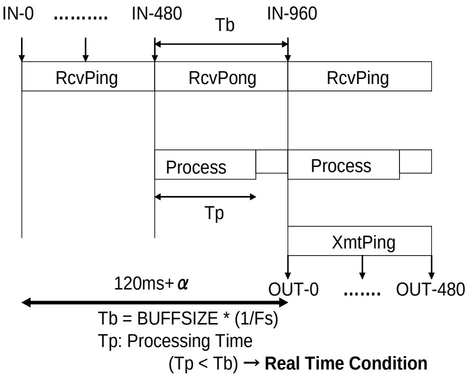
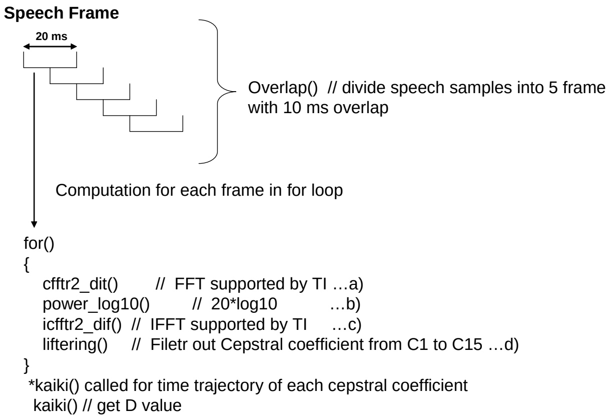


Figure 6: Double buffering timing

2.3 Software Setting

Some methods have been created for realizing steady-state suppression with DSP. In computer simulation done by MATLAB, all speech samples were read and divided into speech frames to analyze steady-state portions. However, in DSP realization, 480 continuous speech samples were saved in the received buffer at each time, and signal processing then proceeded 480 samples by 480 samples. The method defined in DSP is shown in Figure 7.


Speech Frame
20 ms
Overlap() // divide speech samples into 5 frame with 10 ms overlap
Computation for each frame in for loop

```
for()
{
   cfftr2_dit()      //  FFT supported by TI …a)
   power_log10()       //  20*log10          …b)
   icfftr2_dif()  //  IFFT supported by TI    …c)
   liftering()    //  Filetr out Cepstral coefficient from C1 to C15 …d)
}
 *kaiki() called for time trajectory of each cepstral coefficient
  kaiki() // get D value
```

Figure 7: Steady-state evaluation

This routine worked to return the *D* value (mean square of the regression coefficients obtained from a cepstral matrix). The methods explained above were working well, and we confirmed the specification of the methods with a specific test program. The final aspect to check was number of cycle times because the methods had to be completed within 60 ms in order to be implemented as real-time processing.

1. **Overlap( ) = 4338 cycles**
2. **Cfftr2_dit( ) = 4885 cycles**
3. **Power_log10( ) = 391819 cycles**
4. **Cifftr2_dif( ) = 4648 cycles**
5. **liftering( ) = 507 cycles**
6. **kaiki( ) = 1437 cycles**

The function defined in a) to d) was in a for loop, so the cycle time had to be 5 times more. Considering this, the cycle time for the entire process was 2015070 cycles. This cycle time covered the analysis time for five continuous speech frames. Summing up the cycle time for data exchange from the received buffer to the transmitted buffer with or without suppression, the final value was 2018736 cycles. Because the operating frequency of TMS320C6713 DSP is 225 MHz, the process time is computed as total cycle time divided by the operating frequency of the device. Finally, we achieved a process time of about 9.0 ms and this was markedly lower than the maximum process time of 60 ms. Thus, steady-state suppression using the DSP method works stably on the chip level.

3. Experiments

To evaluate the DSP method, we conducted two experiments. These experiments differed only in the use of younger or elderly listeners. The main purpose of this experiment was to examine whether we could obtain the same performance for the DSP and FB methods. We were not able to compare the two methods directly because of the different sampling frequency. Therefore, we evaluated the methods by the following step-by-step comparison:

i) FB method (sampling frequency of 16 kHz) vs. unprocessed condition (sampling frequency of 16 kHz),

ii) FFT method (16 kHz) and FB method (16 kHz),

iii) FFT method (8 kHz) and FFT method (16 kHz), and

iv) FFT method (8 kHz) and DSP method (8 kHz).

3.1 Experiments for younger listeners

3.1.1 Subjects

Thirty-eight younger subjects (19 males and 19 females, aged 18 to 37 years, average age 22.8 years) participated in the experiment. All were native Japanese speakers.

3.1.2 Stimuli

The original speech samples consisted of 14 nonsense Consonant-Vowel (CV) syllables embedded in a Japanese carrier phrase. The vowel was /a/ and the consonants were /p/, /t/, /k/, /b/, /d/, /g/, /s/, /ʃ/, /h/, /dz/, /ʒ/, /tʃ/, /m/, and /n/. We used two sets of original speech samples: one set uttered by a trained male speaker (ATR Speech Database of Japanese) and the other set uttered by an untrained male speaker. The followings show six conditions of processing:

- unprocessed (16 kHz),
- unprocessed (8 kHz),
- processed by the FB method (16 kHz)
- processed by the FFT method (16 kHz)
- processed by the FFT method (8 kHz)
- processed by the DSP method (8 kHz)

Eighty-four stimuli were prepared for each speaker (14 CVs x 6 processing conditions) in one set. The stimuli were presented randomly.

3.1.3 Procedure

The experiment was conducted in the largest lecture hall at Sophia University, with a capacity of 822 people (reverberation time: 1.3 s). Each subject sat in the back-center area of the hall. An empty seat was left between adjacent subjects. Stimuli were presented through four built-in loudspeakers in the lecture hall (Electro-Voice TS-9040D-LX). The experiment comprised two sessions: for the first session, we played the set of stimuli uttered by the trained speaker and for the second session, we used the untrained-speaker set. The sound level was adjusted to the subjects' comfort level before each session began. In each session, a stimulus was presented only once for each trial. After listening to each stimulus once, subjects were expected to select one of 14 CVs in the Japanese Kana orthography appearing on an answer sheet provided to them. They were given five seconds to record their selection.

3.1.4 Results

Tables 1 and 2 show the mean correct rates among 38 subjects of the first (trained speaker) and second (untrained speaker) sessions, respectively, in terms of the six processing conditions.

Table 1 Correct rates for younger listeners (trained speaker)

	DSP method	FFT method	FB method	Unprocessed
Sampling frequency 8kHz	37.6%	45.9%	—	44.4%
Sampling frequency 16kHz	—	64.7%	60.3%	57.0%

Table 2 Correct rates for younger listeners (untrained speaker)

	DSP method	FFT method	FB method	Unprocessed
Sampling frequency 8kHz	34.0%	31.8%	—	28.7%
Sampling frequency 16kHz	—	45.9%	57.3%	47.9%

3.1.5 Discussion

We discuss the results based on the evaluation steps listed in Section 3 from i) to iv).

i) In the case of the untrained speaker, FB method 16 kHz significantly improved speech intelligibility by 9.4 % (p<0.001) compared to Unprocessed 16 kHz. In contrast, in the case of the trained speaker, although FB method 16 kHz improved speech intelligibility by 3.0% compared to Unprocessed 16 kHz, this improvement was not statistically significant. Thus, "FB method ≥ Unprocessed".

ii) In the case of the trained speaker, the processed signals did not significantly improve speech intelligibility. In the case of the untrained speaker, FFT method 16 kHz significantly decreased speech intelligibility by 11.4 % (p=0.006) compared to FB method. Thus, "FB method ≥ FFT method".

iii) In either the case of the trained or untrained speaker, FFT method 16 kHz yielded higher intelligibility than FFT method 8 kHz (trained speaker: 19.8%, untrained speaker: 14.1%). Because these results show the same gap in speech intelligibility between Unprocessed 16 kHz and Unprocessed 8 kHz (trained speaker: 13.6%, untrained speaker: 19.2%), the gap in speech intelligibility between FFT methods (16 kHz and 8 kHz) might be caused by the different sampling frequency. In other words, the intelligibility drop might be due to the loss of higher frequency components above 4 kHz that are still important for speech intelligibility.

iv) In the case of the untrained speaker, the processed signals did not significantly improve speech intelligibility. In the case of the trained speaker, DSP method 8 kHz significantly decreased speech intelligibility by 8.3% (p=0.016) compared to FFT method 8 kHz. These results indicate "FFT method ≥ DSP method".

Given these results, there might not in fact be a large difference in the algorithms between FFT 16 kHz and FFT 8 kHz; rather the difference in sampling frequency between two methods might cause the decreasing effect. We obtained the following order for the methods in terms of performance:

FB method ≥ FFT method ≥ DSP method

3.2 Experiments for elderly listeners

3.2.1 Subjects

Twenty-three elderly subjects (7 males and 16 females, aged 68 to 92 years, average age 73.5 years) participated in the experiment. All were native Japanese speakers with sufficient physical capabilities to join the experiment and no dementia.

3.2.2 Stimuli

We used the same procedure as described in Section 3.1.2.

3.2.3 Procedure

We used the same procedure as described in Section 3.1.3.

3.2.4 Results

Tables 3 and 4 show the mean correct rates among 23 subjects for the first (trained speaker) and second (untrained speaker) sessions, respectively, in terms of the six processing conditions. For statistical analysis, correct rates were transformed by the arcsine function [7].

Table 3 Correct rates for elderly listeners (trained speaker)

	DSP method	FFT method	FB method	Unprocessed
Sampling frequency 8kHz	19.6%	29.2%	—	26.4%
Sampling frequency 16kHz	—	42.6%	42.2%	34.8%

Table 4 Correct rates for elderly listeners (untrained speaker)

	DSP method	FFT method	FB method	Unprocessed
Sampling frequency 8kHz	26.4%	20.8%	—	21.1%
Sampling frequency 16kHz	—	31.4%	41.9%	33.9%

3.2.5 Discussion

We discuss the results based on the evaluation steps listed in Section 3 from i) to iv).

i) In the case of the untrained speaker, as for the younger listeners' results, FB method 16 kHz significantly improved speech intelligibility by 8.1% (p<0.05) than Unprocessed 16 kHz. In the case of the trained speaker, FB method 16 kHz improved speech intelligibility by 7.5 % compared to Unprocessed, although this improvement was not statistically significant. These results indicate "FB method ≥ Unprocessed".

ii) In the case of the trained speaker, again as for the younger listeners' results, these processed signals were not significantly improved. In the case of the untrained speaker, FFT method 16 kHz significantly decreased speech intelligibility by 10.5% (p<0.05) compared to the FB method. These results indicate "FB method ≥ FFT method".

iii) In either the case of the trained or untrained speaker, as for the younger listeners' results, FFT method 16 kHz yielded higher intelligibility than FFT method 8 kHz (trained speaker: 13.4%, untrained speaker: 10.6%). Because these results show the same gap in speech intelligibility between Unprocessed 16 kHz and Unprocessed 8 kHz (trained speaker: 8.4%, untrained speaker: 12.8%), the gap in speech intelligibility between the FFT methods (16 kHz and 8 kHz) might be caused by the different sampling frequency. That is, the decrease in intelligibility might be due to the loss of higher frequency components above 4 kHz that are still important for speech intelligibility.

iv) In the case of the untrained speaker, as for the younger listeners' results, these processed signals were not significantly improved. In the case of the trained speaker, DSP method 8 kHz significantly decreased speech intelligibility by 5.6% (p<0.05) compared to FFT method 8 kHz. These results indicate "FFT method ≥ DSP method".

Given the results, we suggest that there is not a large difference in the algorithms between FFT 16 kHz and FFT 8 kHz; rather the difference in sampling frequency between two methods might cause the decreasing effect. Therefore, we obtain the following order of the methods in terms of performance:

FB method ≥ FFT method ≥ DSP method

The results for elderly listeners show the same tendency as for younger listeners.

4. Conclusions

In this study, we implemented real-time processing of steady-state suppression using DSP and evaluated the effectiveness of the processing. We found that the current DSP method as yet offers no advantage over the FB method. This is due, in part, to the threshold value, differences in sampling frequency, and differences in cepstral matrix shifts of the algorithms between the FFT and DSP methods. For future work, we plan to change the sampling frequency from 8 kHz to 44.1 kHz or 48 kHz, although this change might result in a heavy calculation load for DSP. In addition, we plan to implement the FFT method using DSP without any compromise; however, the FFT algorithm would need to be processed at less than 60 ms in DSP for real-time processing due to differences in matrix shift. In the present study, we achieved real-time processing of steady-state suppression, which marks a real advancement in our research. We plan to continue using the steady-state suppression technique to further investigate how to create a barrier-free environment for speech perception.

5. Acknowledgements

This research was supported in part by Grants-in-Aid for Scientific Research (A-2, 16203041) from the Japan Society for the Promotion of Science. We are grateful to all of the subjects for their cooperation in this study. We thank Texas Instruments Japan for technical support of DSP implementation.

6. Reference

[1] R. H, Bolt and A. D, MacDonald, “Theory of speech masking by reverberation,” *J. Acoust. Soc. Am.*, Vol. 21, pp. 577-580, 1949.

[2] T. Arai, K. Kinoshita, N. Hodoshima, A. Kusumoto and T. Kitamura, ``Effects of suppressing steady-state portions of speech on intelligibility in reverberant environments," *Proc. Autumn Meet. Acoust. Soc. Jpn.*, Vol. 1, pp. 449-450, 2001

[3] T. Arai, K. Kinoshita, N. Hodoshima, A. Kusumoto and T. Kitamura, “Effects on suppressing steady-state portions of speech on intelligibility in reverberant environments,” Acoustical Science and Technology, Vol. 23, pp. 229-232, 2002.

[4] T. Arai, N. Hodoshima, T. Goto, T. Inoue, N. Ohata, K. Kinoshita, A. Kusumoto, “Pre-Processing technique to prevent degradation of speech intelligibility in reverberant environments” Trans. Tech. Comm. Psycho. Physiol. Acoust., *The Acoustical Society of Japan*, Vol.33, No.5, H2003-59, pp. 341-346, 2003.

[5] N. Hodoshima, T. Goto, N. Ohata, T. Inoue and T. Arai, “The effect of pre-processing for improving speech intelligibility in the Sophia University lecture hall,” *Proc. of the International Congress on Acoustics*, Vol. III, pp. 2389-2392, Kyoto, 2004.

[6] S. Furui, “On the role of spectral transition for speech perception,” *J. Acoust. Soc. Am.*, Vol. 80(4), pp. 1016-1025, 1986.

[7] G. Studebaker, “A rationalized arcsine transform,” *J. Speech and Hearing Research*, Vol. 28, pp. 455-462, 1985.