ICSV14
Cairns • Australia
9-12 July, 2007

14th International Congress on Sound & Vibration

MULTIDELAY ADAPTIVE FILTERS FOR ACTIVE NOISE CONTROL

Xiaojun Qiu[1] and Colin H. Hansen[2]

[1]Key Laboratory of Modern Acoustics and Institute of Acoustics,
Nanjing University, China 210093
[2]School of Mechanical Engineering, University of Adelaide
South Australia 5005, Australia
xjqiu@nju.edu.cn

Abstract

There are two ways to implement frequency domain filtered x LMS (FXLMS) algorithms. The first carries out both control signal generation and control filter updating in the frequency domain, and the second involves implementation of the control filtering in the time domain and updating of the coefficients of the control filters in the frequency domain. Most active noise control literature is focused on the second approach as the first one introduces a delay of at least one FFT block size for the control filter generation, which is usually not acceptable for active noise control. However, the second implementation has a limitation on its maximum computational complexity reduction due to its delayless requirement and it also needs quite a large on-chip memory for its FFT. The multidelay adaptive filter to be discussed in this paper is intended to solve some of the above problems. The multidelay adaptive filter has a flexible structure, which partitions a long filter into many shorter sub-filters so that a much smaller FFT size can be used to reduce the delay and memory requirement while maintaining the low computational complexity and faster convergence properties of the frequency domain algorithm.

1. INTRODUCTION

One of the limitations of current active noise control (ANC) systems is the limited bandwidth and space over which they operate. To increase the upper limiting frequency and to extend the attenuation zone of ANC systems, a higher system sampling rate and multiple channel systems often have to be used. A large number of control filter weights and a large number of control channels bring a significant amount of computational load. There are many different approaches that can be taken to solve the problem, such as using a decentralized system, applying the modal method, implementing the distributed adaptive algorithms with a network control structure and applying subband techniques [1].

An alternative solution to this problem is to use frequency domain adaptive filters. The frequency domain algorithm can reduce the computational complexity significantly by exploiting the Fast Fourier Transformation (FFT), and increase the convergence speed of the algorithm by de-correlating the input signals. There are two ways to implement the frequency domain Filtered x LMS algorithms (FXLMS). The first carries out both control signal generation and control filter updating in the frequency domain, and the second involves implementation of the control filtering in the time domain and updating of the coefficients of the control filters in the frequency domain [2-3]. Most active noise control literature is concerned with the second approach as the first one introduces a delay of at least one FFT block size for the control filter generation, which is usually not acceptable for active noise control. However, the second approach has limitations on its maximum computational complexity reduction due to its delayless requirement and large on-chip memory requirement for its FFT.

The multidelay adaptive filter (MDF) is a flexible structure, which partitions a long filter into many shorter sub-filters so that much smaller FFT size can be used to reduce the delay and memory requirement while maintaining the low computational complexity and faster convergence properties of the frequency domain algorithm. The multidelay adaptive filter was first proposed by Soo and Pang [4] to solve practical implementation problems of the frequency domain algorithm for acoustic echo cancellation [4]. The MDF was sometimes also called the partitioned block frequency domain adaptive filter (PBFDAF) [5]. To completely eliminate the delay of the MDF algorithm while maintaining its low computation complexity, Bendel *et al.* [6] proposed the delayless MDF filter by using a time-frequency hybrid approach. Recently, Buchner *et al.* [7] analyzed the traditional MDF algorithm, and found that while the processing delay can be significantly reduced with the MDF structure, its convergence speed might be decreased for strongly correlated signals due to the missing of the correlations between those shorter blocks. An extended MDF algorithm with a fast implementation algorithm was proposed to solve the problem [7].

This paper will apply the multidelay adaptive filter in ANC, which has to take into account the effects of the cancellation path. The remainder of the paper is organized as follows. The multidelay algorithms for active noise control and its convergence properties are presented and discussed in Section 2, and their computational complexity is analyzed in Section 3. Section 4 concludes with a discussion of the advantages of applying multidelay algorithms in active noise control.

2. THE MULTIDELAY ALGORITHM FOR ACTIVE NOISE CONTROL

Figure 1 shows a block diagram of FIR filtering based on the MDF. The weights of the FIR filter are $\mathbf{w}(n) = \left[w_0(n),\ w_1(n),\ ...,\ w_{L-1}(n)\right]^T$, where superscript ^{T} denotes transposition of a vector or a matrix. $x(n)$ is the input signal, $y(n)$ is the output signal, and n is the time sample index. The following sub filters can be obtained by partitioning $\mathbf{w}(n)$ into K segments, each of length $N = L/K$,

$$\mathbf{w}_{kN}(n) = \left[w_{kN}(n),\ w_{kN+1}(n),\ ...,\ w_{kN+N-1}(n)\right]^T,\ k = 0,\ ...,\ K-1 \tag{1}$$

As the processing block size is N, a $2N$ point FFT is applied to remove the effects of the circular convolution by using the overlap save method [2-7]. Appending N point zeros at the end of the weights of the kth sub filter, and applying a $2N$ point FFT, results in the following:

$$\mathbf{W}_k(m) = FFT_{2N}\left[w_{kN}(n),\ w_{kN+1}(n),\ ...,\ w_{kN+N-1}(n),\ 0,\ 0,\ ...,\ 0\right]^T,\ k = 0,\ ...,\ K-1 \tag{2}$$

where $m=n/N$ is the block index. For a block of inputs $\mathbf{x}_N(m)=\left[x(n),\ x(n+1),\ ...,\ x(n+N-1)\right]^T$, the N filtering outputs $\mathbf{y}_N(m)=[y(n),\ y(n+1),\ ...,\ y(n+N-1)]^T$ can be written as:

$$\mathbf{y}_N(m)=[\mathbf{0}_N\ \ \mathbf{I}_N]\sum_{k=0}^{K-1} IFFT\{diag[\mathbf{X}_{2N}(m-k)]\mathbf{W}_k(m)\} \tag{3}$$

where $[\mathbf{0}_N\ \ \mathbf{I}_N]$ is an $N\times 2N$ matrix that consists of a concatenation of an $N\times N$ zero matrix $\mathbf{0}_N$, and an $N\times N$ identity matrix $\mathbf{I}_N$. $\mathbf{X}_{2N}(m)=FFT_{2N}\{[\mathbf{x}_N^T(m-1)\ \mathbf{x}_N^T(m)]^T\}$, and $diag[\cdot]$ defines a $2N\times 2N$ diagonal matrix with its ith diagonal term equal to the ith term of the $2N$ vector. It can be seen from Fig. 1 and Eq. (1) that the MDF uses a smaller FFT size $2N$ instead of $2L$, resulting in a shorter delay of N instead of L and less memory requirements compared with the original frequency domain filtering. The original full block frequency domain filtering is a special case of the MDF with $N=L$.

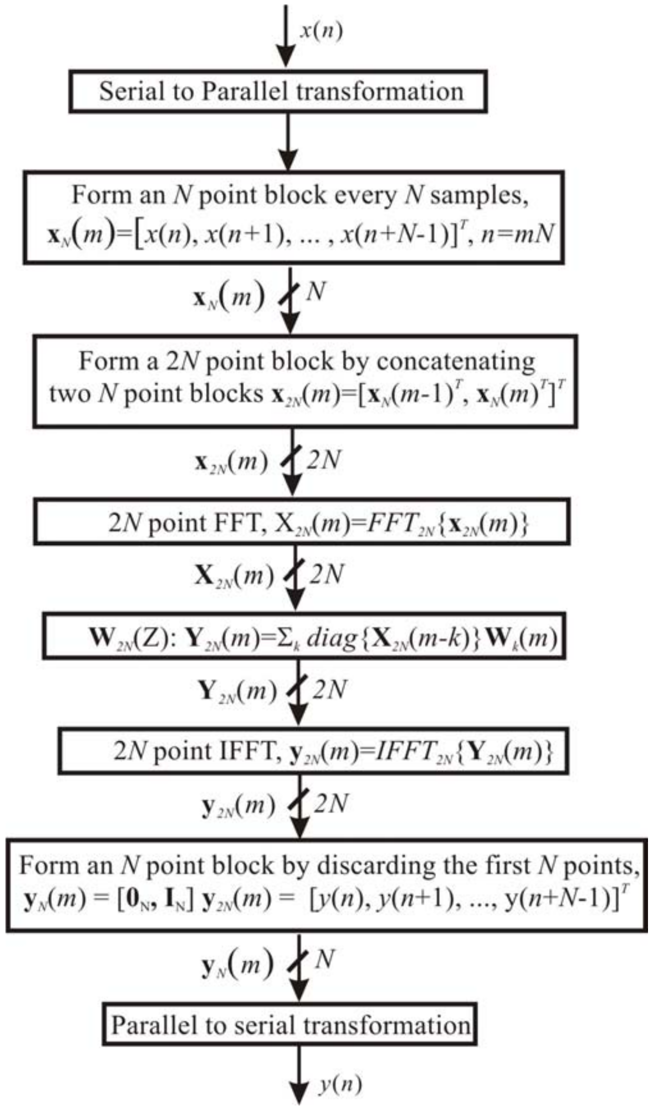

Figure 1. Block diagram of FIR Filtering based on the MDF.

Figure 2 shows a block diagram of the LMS algorithm based on the MDF. For a block of error signals $\mathbf{e}_N(m)=\left[e(n),\ e(n+1),\ ...,\ e(n+N-1)\right]^T$, inserting N point zeros in front of it, and applying a $2N$ point FFT, the following is obtained,

$$\mathbf{E}_{2N}(m) = FFT_{2N}\left[0,\ 0,\ ...,\ 0,\ e(n),\ e(n+1),\ ...,\ e(n+N-1)\right]^{T} \tag{4}$$

The weight update equations for partition index $k = 0,\ ...,\ K-1$ are

$$\mathbf{W}_{k}(m+1) = \mathbf{W}_{k}(m) + 2\mu FFT_{2N}\{\begin{bmatrix}\mathbf{I}_{N} & \mathbf{0}_{N}\\ \mathbf{0}_{N} & \mathbf{0}_{N}\end{bmatrix} IFFT_{2N}\{diag[\mathbf{X}^{*}_{2N}(m-k)]\mathbf{E}_{2N}(m)\}\} \tag{5}$$

where $(\cdot)^{*}$ denotes the complex conjugate, and μ is the convergence coefficient.

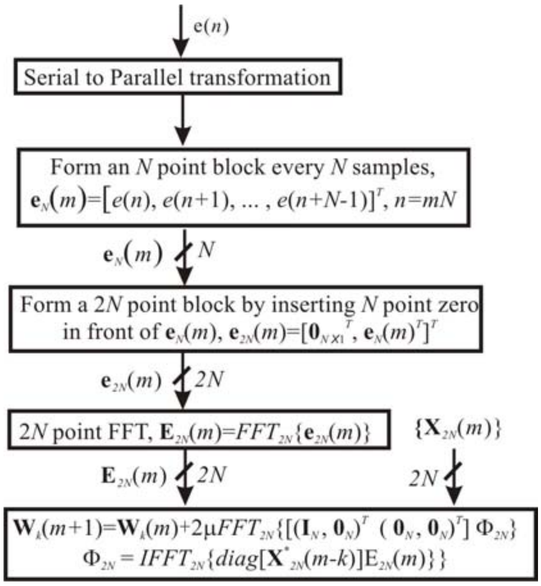

Figure 2. Block diagram of the LMS algorithm based on the MDF.

Figure 3 shows a block diagram of the FXLMS algorithm based on the MDF for ANC systems. $x(n)$ is the reference signal from the noise source and $P(z)$ is the primary path transfer function of the physical acoustic system between the primary noise $p(n)$ and $x(n)$. The actual control signal at the position of the error sensor results from filtering the output of the controller, $y(n)$, with the physical cancellation path transfer function $C_o(z)$, which is modelled by $C(z)$ by injecting uncorrelated random noise into the system. The error signal $e(n)$ is the summation of the control signal at the error sensor, the modelling signal generated by $r(n)$ and the primary noise. All of the FIR filtering and the LMS update are based on the MDF.

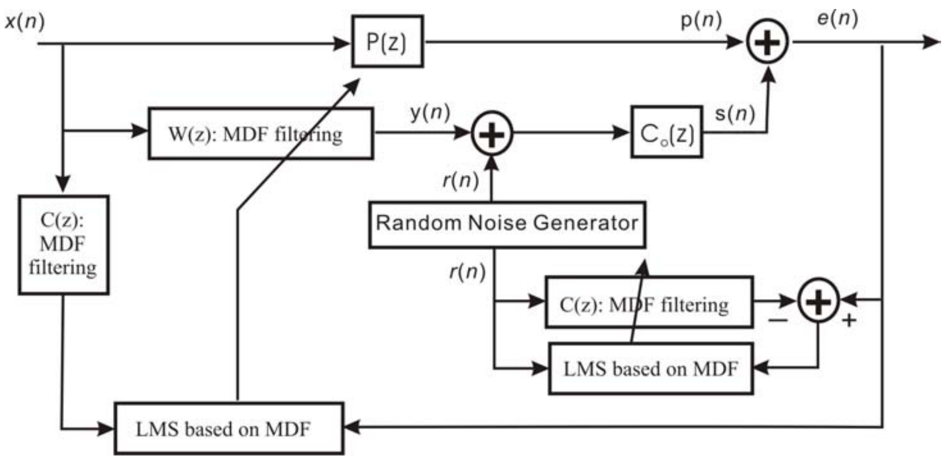

Figure3. Block diagram of the ANC system using the FXLMS algorithm based on the MDF.

To apply the FXLMS algorithm based on the MDF, the control filter length L and the block size N must be determined by considering the causality of the physical system. The length of the FIR filter for the cancellation path model is usually the same as the control filter. Assuming $N = L/K$, the control filter and the cancellation path model can be defined respectively in the frequency domain directly by K complex vectors, $\mathbf{W}_k(m)$ and $\mathbf{C}_k(m)$ for $k = 0, ..., K-1$, where the length of each vector is $2N$. According to Fig. 1, the $2N$ point frequency domain reference signal $\mathbf{X}_{2N}(m)$ can be obtained by applying a $2N$ point FFT on the concatenation of two blocks of the inputs $\mathbf{x}_N(m) = \left[x(n),\ x(n+1),\ ...,\ x(n+N-1)\right]^T$. That is,

$$\mathbf{X}_{2N}(m) = FFT_{2N}\{[\mathbf{x}_N^T(m-1)\ \mathbf{x}_N^T(m)]^T\} \tag{6}$$

The $2N$ point frequency domain modelling signal $\mathbf{R}_{2N}(m)$ can be obtained in the same way as described above and the $2N$ point frequency domain error signal $\mathbf{E}_{2N}(m)$ can be obtained with Eq. (4). By using Eq. (5), the frequency domain vectors for $k = 0, ..., K-1$ of the cancellation path model can be obtained by

$$\mathbf{C}_k(m+1) = \mathbf{C}_k(m) + 2\mu_c FFT_{2N}\{\begin{bmatrix} \mathbf{I}_N & \mathbf{0}_N \\ \mathbf{0}_N & \mathbf{0}_N \end{bmatrix} IFFT_{2N}\{diag[\mathbf{R}_{2N}^*(m-k)]\mathbf{E}_{2N}(m)\}\} \tag{7}$$

where μ_c is the convergence coefficient. Letting μ_w be the convergence coefficient for the control filter update, then the frequency domain vectors for $k = 0, ..., K-1$ of the control filter can be obtained as

$$\mathbf{W}_k(m+1) = \mathbf{W}_k(m) - 2\mu_w FFT_{2N}\{\begin{bmatrix} \mathbf{I}_N & \mathbf{0}_N \\ \mathbf{0}_N & \mathbf{0}_N \end{bmatrix} IFFT_{2N}\{diag[\mathbf{F}_{2N}^*(m-k)]\mathbf{E}_{2N}(m)\}\} \tag{8}$$

where $\mathbf{F}_{2N}(m) = FFT_{2N}\{[\mathbf{f}_N^T(m-1)\ \mathbf{f}_N^T(m)]^T\}$, and the time domain filtered reference signals $\mathbf{f}_N(m) = [f(n),\ f(n+1),\ ...,\ f(n+N-1)]^T$ are given by,

$$\mathbf{f}_N(m) = [\mathbf{0}_N\ \mathbf{I}_N]\sum_{k=0}^{K-1} IFFT\{diag[\mathbf{X}_{2N}(m-k)]\mathbf{C}_k(m)\} \tag{9}$$

The N control filter outputs can be obtained simultaneously by using Eq. (3), and they are then sent to DA converter one by one at the sampling rate.

In a similar way as shown in Fig. 1, the above FXLMS algorithm based on the MDF has an inherent delay of N samples for the control filter filtering. In some practical situations, some part of the noise energy might take a very short time to propagate from the noise source to the error sensor. In this instance, the block size N for the control filter filtering based on the MDF in shown Fig. 3 must be less than the delay to account for this part of the noise. However, small number N would reduce the efficiency of applying the FFT. For example, for active control of noise radiation from a compact source in a large workshop with surrounding secondary sources, the primary noise energy at an error sensor consists of the direct sound and the reverberant sound. The propagation time of the direct sound might be very short from the reference sensor to the error sensor, for example, about 3ms for 1m distance between them. However, the first reflected sound might arrive at the error sensor after about 60ms if the nearest wall is 10m away. For a 10 kHz sampling rate, 3ms represents 30 samples, and 60ms represents 600 samples. To maintain the performance of the FXLMS algorithm based on the MDF, the block size must be less than 30 samples to be able to reduce the direct sound, which significantly reduces the computation complexity advantage of the MDF algorithm.

To overcome the above problem while maintaining the advantages of the FXLMS algorithm based on the MDF, a modified algorithm based on the delayless MDF is presented here using the same time-frequency hybrid approach proposed in [6]. The idea is to calculate the first partition in the time domain. Instead of using Eq. (3), the control output can be obtained as

$$\mathbf{y}_N(m) = \sum_{l=0}^{N-1} x(n-l) w_l(m) + [\mathbf{0}_N \ \mathbf{I}_N] \sum_{k=1}^{K-1} IFFT\{diag[\mathbf{X}_{2N}(m-k)]\mathbf{W}_k(m)\} \tag{10}$$

where

$$[w_0(m),\ w_1(m),\ ...,\ w_{N-1}(m)]^T = [\mathbf{I}_N \ \mathbf{0}_N] IFFT_{2N}[\mathbf{W}_0(m)] \tag{11}$$

Unlike the approach proposed in [6], it is not appropriate for the FXLMS algorithm based on the delayless MDF to update the first partition of the control filter weights in the time domain because the reference signal needs to be filtered by the entire cancellation path transfer function, and with long taps, that would result in a large increase of the computational load.

Calculating the first partition in the time domain has also been suggested in the delayless subband adaptive filters [8], and the delayless subband algorithms have also been applied for ANC to increase the convergence performance and to reduce the computational complexity [1, 8, 9]. However, it has been found that although the MDF can be treated as a special kind of subband adaptive filters, its convergence performance usually is superior to that of subband adaptive filters [10].

The convergence of the FXLMS algorithm based on the MDF can potentially be faster than that of the time domain FXLMS algorithm by using a different convergence coefficient for each frequency bin if the spectrum of the filtered reference signal has a large dynamic range, however it may be slower than that of the traditional frequency domain FXLMS algorithm for strongly correlated signals due to the lack of correlations between the shorter blocks. The time domain FXLMS algorithm has no delay for the control filter filtering, the FXLMS algorithm based on the MDF has N samples delay, and the traditional frequency domain FXLMS algorithm has $L = KN$ samples delay. The FXLMS algorithm based on the delayless MDF also has no delay, but its convergence performance is determined by the division of energy between the time domain and the frequency domain partitions. Their computational complexities will be compared in the next section.

3. COMPUTATIONAL COMPLEXITY

The computational complexity of the FXLMS algorithm based on the MDF is compared here with the ordinary time domain and frequency domain FXLMS algorithms, and the subband algorithms, where the (real) multiplications per input sample is used as a measure. During the calculations, it is assumed that $2N \log_2 2N$ real multiplications are required for a $2N$ point FFT or IFFT, and $8N$ real multiplications are required for $2N$ complex multiplications in the frequency domain FIR filtering or LMS update [2]. For the subband FXLMS algorithm, it is assumed that the length of the prototype filter is K_L, the down sampling rate is D, and the number of subband is K. For each subband signal (reference and error signals) generation,

$2(K_L + K\log_2 K)/D$ real multiplications are needed per input sample. For each subband, the complex filtered reference signal generation and the complex LMS update each needs $4L/D$ multiplications per input sample. Altogether, there are *K* subband cancellation paths and *K* subband complex LMS updates. As the input signals are real, only (*K*/2+1) complex subbands need to be processed per *D* samples. Thus, the total number of real multiplications per fullband input sample for the filtered reference signal generation and control filter update are approximately $4LK/D^2$. For the subband to fullband weight transformation, $K_L + K\log_2 K$ multiplications are needed per *D* samples.

Table 1 shows the average number of real multiplications required per input sample to implement various FXLMS algorithms, where it is assumed that the length of the control filter and the cancellation path model is *L*, the partition number for the MDF algorithms is *K*, and the block size is *N* = *L/K*. In the table, TD FXLMS means the time domain FXLMS algorithm, FD FXLMS means the traditional constrained frequency domain FXLMS algorithm, DFD FXLMS means the traditional delayless constrained frequency domain FXLMS algorithm in [2], Delayless subband means the FXLMS algorithm based on delayless subband filtering, MDF FXLMS means the FXLMS algorithm based on the constrained MDF, and Delayless MDF means the FXLMS algorithm based on the delayless constrained MDF.

For the frequency domain and MDF algorithms, unconstrained implementation can be applied, which uses Eq. (12) instead of Eq. (8) to further reduce the computational complexity by removing one FFT and one IFFT for each block,

$$\mathbf{W}_k(m+1) \approx \mathbf{W}_k(m) - 2\mu_w diag[\mathbf{F}_{2N}^*(m-k)]\mathbf{E}_{2N}(m) \tag{12}$$

However, as the costs of removing two FFT operations are slower convergence and larger misadjustment [4], unconstrained implementations of the frequency domain and MDF algorithms are not considered in this paper. For the frequency domain filtered reference signal generation, the computational complexity can also be reduced by removing one FFT and one IFFT for each block with the following equation instead of Eq. (9)

$$\mathbf{F}_{2N}(m) \approx \sum_{k=0}^{K-1} diag[\mathbf{X}_{2N}(m-k)]\mathbf{C}_k(m) \tag{13}$$

However, as it is not clear whether this saving would seriously bias the adaptation [2], Eq. (13) is not adopted in this paper.

The computational load for the cancellation path modelling is not included in the table for brevity; however, it follows the same trends as for the FXLMS algorithm, and can be estimated by removing the contribution of the filtered reference signal generation part from that of the FXLMS algorithms.

Considering an ANC system with control filter and cancellation path model of length 4096, it can be calculated from the table that the traditional constrained frequency domain FXLMS algorithm can significantly reduce the computational complexity down to about 1.6% of that of the time domain algorithm. However, the associated delay is 4096 samples. The maximum computational complexity reduction for the delayless constrained frequency domain FXLMS algorithm and subband algorithms are about 33% of that of the time domain algorithm. The FXLMS algorithm based on the constrained MDF can reduce the computational complexity down to about 7% with a delay of 128 samples, and the FXLMS algorithm based on the delayless constrained MDF can reduce the computational complexity down to about 8% without bringing any delay to the system. The main reasons for the computational complexity reduction are the use of block processing via FFT and update of the control filter at a lower rate.

.

Table 1. The average number of real multiplications required per input sample to implement various FXLMS algorithms.

	Control filter filtering	Filtered-x signal generation	Control filter update	Transformation	Total
TD FXLMS	L	L	L	0	$3L$
FD FXLMS	8	8	8	$14\log_2 2L$	$24+14\log_2 2L$
DFD FXLMS	L	8	8	$14\log_2 2L$	$16+14\log_2 2L+L$
Delayless subband	L	$2LK/D^2$	$2LK/D^2$	$3(K_L+K\log_2 K)/D$	$3(K_L+K\log_2 K)/D +4LK/D^2+L$
MDF FXLMS	$8K$	$8K$	$8K$	$14\log_2 2N$	$24K+14\log_2 2N$
Delayless MDF	$N+8(K-1)$	$8K$	$8K$	$16\log_2 2N$	$24K+16\log_2 2N+N-8$

4. CONCLUSIONS

FXLMS algorithms based on the multidelay adaptive filter are proposed in this paper, and their convergence properties, delay and computational complexity are discussed and compared with the time-domain, frequency-domain and subband algorithms. It is found that for an ANC system with 4096 tap filter, the FXLMS algorithm based on the delayless MDF can reduce the computational complexity down to 8% of that of the original time domain FXLMS algorithm without bringing any delay to the system.

5. ACKNOWLEDGEMENTS

Projects 10674068 supported by NSFC.

REFERENCES

[1] X. Qiu, N. Li, G. Chen, and C. H. Hansen, "The implementation of delayless subband active noise control algorithms", Proceedings of Active '06, September 18-20, Adelaide, Australia, 2006.

[2] S. J. Elliott, Signal Processing for Active Control (Academic Press, London, 2001).

[3] S. M. Kuo and D. R. Morgan, Active Noise Control Systems - Algorithms and DSP Implementations (John Wiley & Son Inc., 1996).

[4] J. Soo and K. Pang, "Multidelay block frequency domain adaptive filters," IEEE Trans. Acoust., Speech, Signal Processing, **38**, 373–376 (1990).

[5] J. Borrallo and M. Otero, "On the implementation of a partitioned block frequency domain adaptive filter (PBFDAF) for long acoustic echo cancellation," Signal Processing **27**,301-315, (1992).

[6] Y. Bendel, D. Burshtein, O. Shalvi, and E. Weinstein, "Delayless frequency domain acoustics echo cancellation," IEEE Trans. Speech and Audio Processing, **9**, 589-597 (2001).

[7] H. Buchner, J. Benesty, T. Gansler and W. Kellermann, "Robust Extended Multidelay filter and double-talk detector for acoustic echo cancellation," IEEE Trans. Audio, Speech and Language Processing **14**, 1633-1644 (2006).

[8] D. R. Morgan and J. C. Thi, "A delayless subband adaptive filter architecture", IEEE Trans. Signal Processing, **43**, 1818-1830 (1995).

[9] S. J. Park, J. H. Yun and Y. C. Park, "A delayless subband active noise control system for wideband noise control", IEEE Trans. Speech and Audio Processing, **9**, 892-899 (2001).

[10] K. Eneman and M. Moonen, "Hybrid Subband/Frequency-Domain Adaptive Systems. Signal Processing", **81**, 117-136, (2001).