VLSI ARCHITECTURE OF THE GENERALIZED MULTI DELAY FREQUENCY- DOMAIN ALGORITHM FOR ACOUSTIC ECHO CANCELLATION

Amer EL HELWANI, Patrice LE SCAN

FRANCE TELECOM - CNET
chemin du vieux chene
38243 MEYLAN - FRANCE

ABSTRACT

Within the context of acoustic echo cancellation, the convergence rate of the NLMS adaptive filtering algorithm is not sufficient when the input signal is strongly correlated (speech signal). The GMDFα algorithm allows a faster convergence rate and faster tracking of the variation in the echo path to be identified. In these applications the filter may have several thousand tap length at a sampling rate of 16 KHz. This fact leads to difficulties in real time implementation of the algorithm. This paper describes an optimized architecture of a specific circuit which is designed to perform the computational intensive task of the GMDFα algorithm in real time. It can be carried out for long impulse response filters.

1- Introduction

For some years, there has been considerable interest shown in improving high quality handfree telephone and acoustic conference applications. However, the acoustic coupling between the loudspeaker and the microphone of each terminal is a significant operating difficulty [1-3].

A solution to this problem may be the application of the adaptive filtering method to an Acoustic Echo Canceller (AEC). The adaptive filter results in a FIR filter structure with an impulse response with variable coefficients. An algorithm computes the variation of the filter coefficients and minimizes the matching error energy (Figure 1). However, the adaptive filter length may be several thousand taps at a sampling rate of 16 KHz. This fact leads to difficulties in real-time implementation of the algorithm.

2- The GMDF algorithm

The NLMS algorithm has been used for many years thanks to its simplicity. However its convergence rate and tracking properties are not sufficient when the input signal is strongly correlated (speech signal), and its computational complexity is relatively high. These facts have led specialists to introduce other algorithms which are more efficient, but also more complex. From among these algorithms we have chosen to implement the GMDF algorithm.

The GMDF (Generalized Multi-delay frequecy-domain filtre) algorithm takes its place among the frequency-domain block adaptive algorithms [4-11]. A block formulation involving a filter with fixed coefficients during N samples (N is the block size) allows various techniques (FFT, fast FIR) to be applied in the frequency domain for reducing the arithmetic complexity. In early versions, the size of the block was equal to the number of filter tap-weights. The associated processing delay, equal to the block size, appeared to be prohibitive in applications where long impulse responses have to be processed. One method for reducing the input/output delay is to segment into small blocks the impulse response to be identified. This idea has given rise to several versions of the Multidelay Frequency-Domain Filtering (MDF) algorithms [12].

The GMDFα algorithm [13-15] shares the same basis as most block frequency-domain procedures. The main difference lies in the introduction of a new parameter, α, which controls the amount of overlapping between the successive input and output blocks. This is achieved by using a specific extension of the block OLS (OverLap Save) technique, the WOLA block-convolution method, presented in [1,2]. Obviously, the filter is adapted α times per block, α being the overlapping factor. This oversampling implies an arithmetic complexity incrementation, but allows a fast convergence rate and faster tracking of the variation of the echo path to be identified (Figure 2).
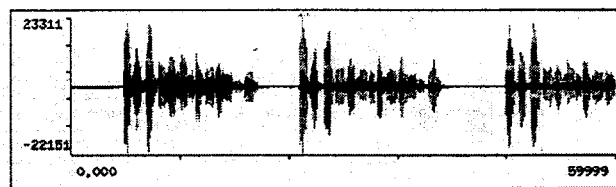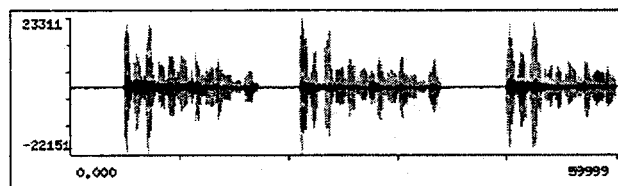

Fig 2.a) LMS algorithm: L=512

Fig 2.b) GMDF algorithm: L=512, K=2, α=4

L being the filter length, K tne block number (K=L/N).

2-1 The algorithm complexity

The filter taps are adapted α times per block of N samples, so the computational task (compared to a classical Frequency Adaptive Filter) is multiplied by α:

$$CC_{GMDF} = \alpha\left((10\frac{L}{N}+6)MA + \frac{3+2K}{N}FFT\right)/sample$$

where L is the filter length, N the block size, K the number of the blocks (K=L/N), FFT means a 2N point FFT, MA = real multiply-add.

Using the radix-2 algorithm, a 2N-point FFT requires NlogN complex multiply-adds (CMA). If we consider that 1 CMA= 4 MA, then a 2N-point FFT requires (4NlogN) real multiply-adds:

$$CC_{GMDF} = \alpha(10K+6+(12+8K)\log(N))MA/sample$$

The Unconstrained algorithm (UGMDF) [3] is a version which requires 2K FFT less than the constrained version, its performances being slightly inferior.

Likewise, the number of memorization elements increases (we should memorize the FFT of the αK last input blocks) :

$$ME_{GMDF}=2[\alpha(K-1)+1](N+1)]+(2K+6)(N+1) + 6N$$
Memorization Elements

Compared to the NLMS ($CC_{NLMS}=2L$, $ME_{NLMS}=2L$) we can notice the significant reduction in arithmetic complexity, especially when α =1 (table 1). On the other hand the GMDF algorithm requires a larger number of memorization elements (table 2).

	α=1	α=2	α=4
L=512, N=128, K=4	354 (0.34)	708 (0.70)	1416 (1.4)
L=512, N=64, K=8	542 (0.53)	1048 (1.02)	2096 (2.05)
L=1024, N=128, K=8	618 (0.3)	1236 (0.6)	2472 (1.2)
L=2048, N=256, K=8	694 (0.17)	1388 (0.34)	2776 (0.68)

table 1- GMDF theoretical computational task (comparison with the NLMS algorithm)

	α=1	α=2	α=4
L=512, N=128, K=4	3606 (3.52)	4380 (4.28)	5928 (5.79)
L=512, N=64, K=8	2854 (2.79)	3764 (3.68)	5584 (5.45)
L=1024, N=128, K=8	5670 (2.76)	7476 (3.65)	11088 (5.41)
L=2048, N=256, K=8	11302 (2.76)	14900 (3.64)	22096 (5.39)

table 2- GMDF theoretical memory size (comparison with the NLMS algorithm)

3- VLSI Implementation of the algorithm on ASIC

The implementation of this algorithm on a General Purpose Digital Signal Processing (GPDSP) has been studied[2]. Nevertheless, the application of the algorithm in real time came up against some difficulties. This led us to look for a specific architecture to implement it. This solution requires a long design time, but is more efficient if we consider the computation rapidity, the circuit area and the power consumption criteria. It allows an appropriate choice of the architectural elements (operating unit, memory types,...) in order to achieve the computational task with minimum loss of time. This solution appears to be the only realistic possibility for implementing filters having long impulse responses.

The current VLSI technology CMOS 0.5μ allows less than 20ns computation time for a 16X16 multiplication-accumulation, which in turn allows at least 3125 MA (Multiply-Add) to be performed in a 62500ns sampling period (Fe=16KHz). Therfore, the arithmetic complexity problem will not overcome even for L=2048 taps.

In contrast, the filter length is limited by the large dimension of the required memory: 11000 words for L=1024 and 22000 for L=2048 with an oversampling factor of α=4.

In VLSI integration, the solution to this problem is given by the use of delay lines with optimized area as memorization elements. This is possible in our case, because the algorithm allows a sequential access to the input samples and the filter taps. This delay line is a dynamic memory with sequential access, high density: 1 memory-point = 3 transistors (Fig.2), and very high speed. This delay line has been already designed at CNET.

This solution results in a 30% area gain (compared to a RAM), and allows the execution of a read-write cycle in one clock cycle, which enables the operating unit to be used at maximum rate .

For example, for a 16 bit data path in CMOS 0.5μ technology, we have the following results (MEA: MEmorization Area):
- for L=512, α=1 ==>MEA=3.8 mm^2, α=2 ==>MEA=5.5mm^2, α=4 ==>MEA=9.45mm^2 ;
- for L=1024, α=1 ==>MEA=7 mm^2, α=2 ==>MEA=10.5 mm^2, α=4 ==>MEA=17.5 mm^2 ;
- for L=2048, α=1 ==>MEA=13.5 mm^2, α=2 ==>MEA=20 mm^2, α=4 ==>MEA=33.5 mm^2

which means that if we use dynamic delay lines, the circuit area remains reasonable large for a long filter length.

3-1 Architecture

The specific VLSI architecture for the GMDF algorithm implementation (Figure 5) involves [16]:

FFT operator: a specific FFT operator ensures a high computation rapidity and avoids the disadvantages of communication with the outside world. This operator allows a M-point FFT to be achieved in $M\log_2 M$ clock cycles.

Memorization elements:
-lar_x : delay line (2N) to store the current input block;

-lar_y: delay line (N) to store the N last reference output samples;
- lar_μ : delay line (N+1) to store the adaptation steps;
- lar_er: delay line (R=N/α) to store the current error output vector;
-lar_X: α delay lines (2[α(K-1)+1](N+1)]) to store the FFT of the αK last input blocks;
-lar_H: delay line (2K(N+1) to store the complex filter taps);
-lar_a: delay line to store intermediate computations.

Operating unit (OU): is composed of 2 multipliers with multiplexed inputs and one accumulator. It allows the execution of a complex multiplication in only two clock pulses (Figure 3).

control unit: generates, in response to the clock pulses, the commands for the memories, multiplexors, and registers. To ensure a correct synchronisation, the operating unit registers are sensible to the rising edge clock pulse while the memorization elements are sensible to the falling edge. This control unit is a state machine, it is automatically synthesized using logic synthesis tools.

3-2 Functional description

The latest R input samples makes up, together with the old (2N-R) samples, the current block (lar_x) (Figure 4). The FFT of this block is memorized in lar_X pushing out the oldest block Then the frequency-domain convolution product is computed (lar_X.lar_H). A reverse FFT operator computes the time-domain convolution product (lar_a) which is used with the reference output vector (lar_y) to synthesize the current error vector (lar_e : R samples) using the WOLA technique. Finally, the frequency-domain filter taps are sequentially adapted, constrained and memorized in lar_H.

3-3 VLSI design

The algorithm has been simulated in VHDL using speech files to determine the datapath width and the memory word size. The algorithm will be implemented in fixed point arithmetic instead of floating point for having less complexity, higher speed and less power consumption. We are designing now the ASIC using logic synthesis for two applications 7 kHZ bandwith telephone and car cellular telephony. Using the CMOS $0,5\mu$ technology, we have the following results:

The circuit speed

Using the proposed architecture the execution of the algorithm requires:

$\alpha[5K + 6logN + 4KlogN + 5K/N]$ cycles / input sample
(constrained case)

$\alpha[5K + 6logN + 5K/N]$ cycles / input sample
(unconstrained case)

The frequency of the clock signal is 50 MHz (1 cycle=20 ns) which allows to perform the algorithm computational task in real time for filters having several thousand tap length (L=8000, K=8, α=4)

The circuit area:

for L=1024 taps, K=8, α=4
- FFT operator: 1,8 mm2
- Memorization elements: 17,5 mm2
- operating unit: 0,5mm2
- control unit: 0,3mm2

Therefore, the whole circuit area will not exceed 30 mm^2.

Conclusion

Working in the frequency domain requires less computation but more memorization elements, which means a larger area of VLSI circuit implementing the algorithm. The circuit has been designed to achieve the arithmetic complexity of the GMDF algorithm at maximum rate allowing the estimation of acoustic channel with a long impulse response. Dynamic delay lines memory have been used to reduce the circuit area. We have used a reliable and fast method based on a VHDL description of the whole circuit allowing the front end design using automatic synthesis tools. The proposed architecture can be considered as a reference for frequency-domain algorithm implementations.

References

[1] A. Gilloire, J.F. Zurcher "Achieving the control of the acoustic echo in audio terminals", signal processing theory and practice, Elsevier 1988 pp.491.

[2] J.P. Julien, G. Le Tourneur, A. Gilloire, "Acoustic echo controler for wide-band hands free telephony", Signal processing theory and practice, Elsevier 1990 pp.1983.

[3] E. Hänsler, "The hands-free telephone problem - An annotated bibliography update-", annals of telecommunications, tome 49, N°7-8, Jul-Aug.1994.

[4] M.J. Dentino, J. McCool and B. Widrow, "Adaptive filtering in the frequency domain", Proc. IEEE, vol. 66, pp. 1658-1659, Dec. 1978.

[5] N.J. Bershad and F.L. Feintuch, "Analysis of the frequency domain adaptive filters", Proc. IEEE, vol. 67, pp. 1658-1659, Dec. 1979.

[6] E.R. Ferrara, "Fast implementation of LMS Adaptive Filters", IEEE Trans. ACoust., Speech, Sig. Proc., vol ASSP-28, NO 4, Aug. 1980.

[7] S.S. Narayan and A.M. Peterson, "Frequency domain least mean-square algorithm", Proc. IEEE, vol 69, p. 124-126, Jan. 1981.

[8] G.A. Clark, S.K. Mitra and S.R. Parker, "Block implementation of adaptive digital filters" IEEE Trans. CAS, vol.. 28, NO 6, June 1981.

[9] D. Mansour and A.H. Gray, JR. "Performance characteristics of the unconstrained frequency-domain adaptive filter", Proc. 1982 IEEE Int. Symp. Circuits Syst., May 1982, pp. 695-698.

[10] D. Mansour and A.H. Gray, JR. "Unconstrained frequency-domain adaptive filter", IEEE Trans. on ASSP, vol. 30, NO 8, Oct.1982.

[11] J.C. LEE and C.K. Un "Performance Analysis of Frequency-Domain Block LMS Adaptive Digital Filters", IEEE trans on CAS, vol 36, NO 2, Feb. 1989.

[12] J.S. Soo and KK. Pang "Multidelay Block Frequency Domain Adaptive Filter" IEEE Trans. on ASSP, Feb. 1990, pp. 373-376.

[13] E.Moulines, O.Ait Amrane, Y.Grenier, "The Generalized Multidelay Adaptive Filter : Structure and convergence analysis", Télécom Paris 93 D004, 1993.

[14] O.Ait Amrane, "Identification de systèmes à réponse impulsionnelle longue par filtre adaptatif en fréquence:

application à l'annulation d'échos acoustiques", Rapport de thèse, Sept.92.

[15] J. Prado, E. Moulines, "Frequency-domain adaptive filtering with applications to acoustic echo cancellation", annals of telecommunications, n07-8, pp. 414-428, Jul.-Aug. 94.

[16] A. El Helwani, P. Le Scan, "Dispositif de supression d'écho entre deux voies de transmission couplées notamment en téléphonie mains-libres", French patent, n°94 09634, Aug.94.

Fig 1- Acoustic Echo Cancellation method

Fig. 2- electrical diagram of a memory point

Fig 3- Operating Unit

Fig 4- Memorization of the input block

Fig. 5 - Specific Architecture for the GMDFα Algorithm