공학 연구논문집 제29권 2호 pp. 119~128, 울산대학교.
Journal of Engineeringl Research Vol. 29, No. 2, pp. 119~128, 1998
University of Ulsan.

Effects of Coefficient Quantization in Block Filters

Uipil Chong
School of Computer Engineering and Information Technology

<Abstract>

As VLSI design technology advances, the potential advantages of block filters are shifted from the savings of the number of computations(using FFT) to computational parallelism. Since fixed point finite precision arithmetic is preferred in high throughput digital signal processing systems, the use of proper block filter structures with good numerical properties is desirable.

In this paper, we extend scalar filter structures to block filter structures and investigate their coefficient quantization effects. Since block filters with quantized coefficients are usually periodically time-varying, the errors are described using bi-frequency representations.

Under coefficient quantization, however, deviations from the desired filter functions are larger for block filters than scalar counter parts since poles and zeros are clustered more closely. Therefore, in general, the filter coefficients need higher precision in the block filter implementations. Through computer simulations of the various block filter structures, finite coefficient quantization properties are investigated.

블록 디지털 필터계수의 양자화 효과에 관한 연구

정 의필
컴퓨터정보통신공학부

<요 약>

VLSI의 설계 기술이 발전함에 따라 블록 필터의 설계는 FFT를 이용한 계산의 수를 줄이는

것으로부터 병렬 계산쪽으로 관심을 가지게 되었다. 왜냐하면 적합한 블록 필터의 설계는 계산속도면에서나 수치 정확도면에서 바람직한 결과를 얻을 수 있기 때문이다.

본 연구에서는 스칼라 필터를 블록 필터로 변환하는 과정을 설명하고, 특히 정밀도가 높은 블록 상태공간 필터와 블록 래티스 필터에 대하여 블록 필터 계수의 양자화 효과를 비교하였다.

1. INTRODUCTION

Many block implementation techniques of digital filters have been proposed for high speed processing. As block processing methods for FIR filters, the overlap save and the overlap-add implementations were developed to allow the fast Fourier transform(FFT) for high speed filtering [1] [2]. More recently, block filter structures have been studied for the high block order [3]. In order to increase throughput rate or, equivalently, to use slower clock rate, block(parallel) processing is desired. Especially, as VLSI design technology advances, the potential advantages of block filters are shifted from the savings in the number of computations to the computational parallelism. Figure 1 illustrates block implementation of an arbitrary scalar digital filter H(z). Any scalar transfer function H(z) can be implemented in a parallel structure called block digital filters.

As illustrated in Figure 1, the scalar input x(n) is converted to block(vector) inputs by using a serial-to-parallel converter and the block outputs are converted to a serial scalar output y(n) by using parallel-to-serial converter.

Block filter structures provide parallel digital filtering schemes suitable for VLSI application. In the implementations of fast VLSI digital signal processing system, it is desirable to achieve high throughput. Therefore, the investigation of coefficient quantization effects and roundoff error analysis [4] for various block filter structures are important.

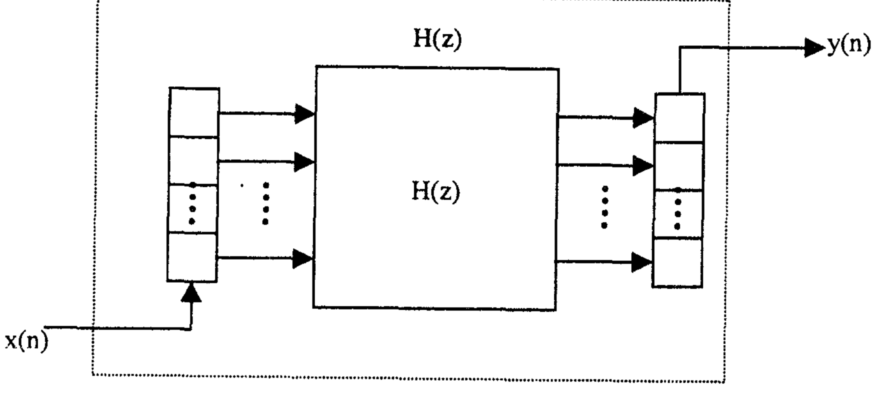

Figure 1. General block digital filter description

State-variable techniques have been used to accomplish of minimizing of filter coefficient quantization effect and roundoff noise. Generally, scalar IIR lattice filter structures have many attractive features; finite precision properties are good, filter structures are modular, and stability is guaranteed. For these reasons, we restrict our study to develop and investigate the Block State Space(BSS) filter and the Block Lattice filter.

2. BLOCK STATE SPACE FILTERS

Block state space(BSS) filters have been known as a popular block implementation technique for IIR digital filters due to good numerical properties [3][4]. We consider the following 4-th order IIR filter as the system function.

$$H(z)=\frac{b_0+b_1z^{-1}+b_2z^{-2}+b_3z^{-3}+b_4z^{-4}}{1+a_1z^{-1}+a_2z^{-2}+a_3z^{-3}+a_4z^{-4}}\,.$$

The BBS filters require the state space description of a specific prototype scalar digital filter implementation. Any scalar digital filter structure has its own state space description that can be used as a prototype filter. In this example, a direct form II structure is chosen as a prototype, as shown in Figure 2.

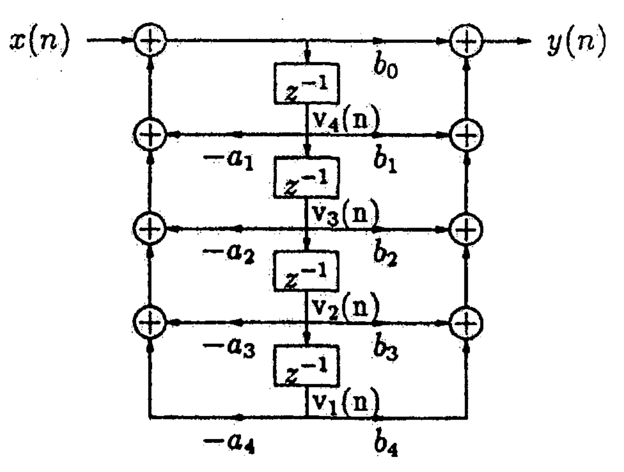

Figure 2. Direct form II structure as a prototype filter

The above direct form II structure has its own state space description .

$$\underline{\nu}(n+1)=A\underline{\nu}(n)+\underline{b}x(n) \tag{1}$$

$$y(n)=\underline{c}^{t}\,\underline{\nu}(n)+dx(n),$$

where the state vector $\underline{\nu}(n)$ is given by

$$\underline{\nu}(n)=[\nu_1(n)\ \ \nu_2(n)\ \ \nu_3(n)\ \ \nu_4(n)]^{t}$$

and

$$A=\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -a_4 & -a_3 & -a_2 & -a_1 \end{bmatrix} \qquad \underline{b}=\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\underline{c}^{t}=[\,b_4-b_0a_4\quad b_3-b_0a_3\quad b_2-b_0a_2\quad b_1-b_0a_1]$$

$$d=b_0$$

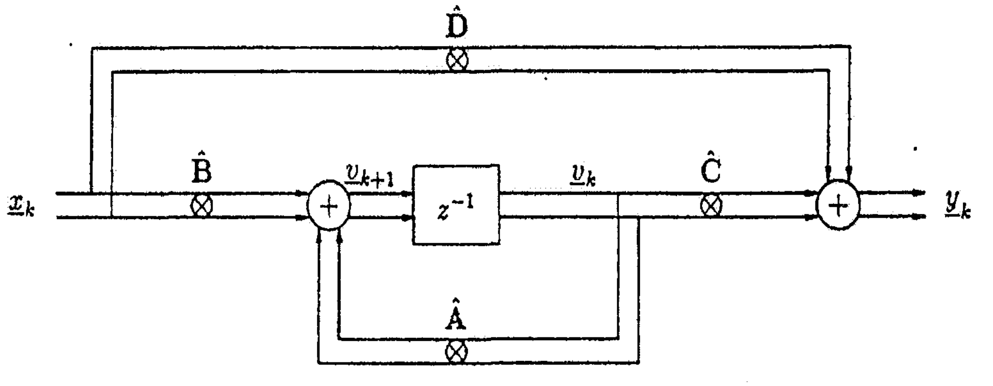

Figure 3. BSS filter with direct form II prototype filter

The equation(1) can be represented in block state space equations as

$$\underline{\nu}_{k+1}=\hat{A}\,\underline{\nu}_k+\hat{B}\,\underline{x}_k \tag{2}$$

$$\underline{y}_k=\hat{C}\,\underline{\nu}_k+\hat{D}\,\underline{x}_k\ , \tag{3}$$

where $\hat{A}=A^4 \qquad \hat{B}=[A^3\underline{b}\quad A^2\underline{b}\quad A\underline{b}\quad \underline{b}]$

$$\hat{C}=\begin{bmatrix} \underline{c}^t \\ \underline{c}^t A \\ \underline{c}^t A^2 \\ \underline{c}^t A^3 \end{bmatrix} \qquad \hat{D}=\begin{bmatrix} d & 0 & 0 & 0 \\ \underline{c}^t \underline{b} & d & 0 & 0 \\ \underline{c}^t A^2 & \underline{c}^t \underline{b} & d & 0 \\ \underline{c}^t A^2 \underline{b} & \underline{c}^t A \underline{b} & \underline{c}^t \underline{b} & d \end{bmatrix}$$

With these matrices $\hat{A}$, $\hat{B}$, $\hat{C}$, and $\hat{D}$, the BSS filter structure of direct form II prototype is represented in Figure 3.

The BSS filters, however, do not have specific structures since they are always implemented in a form shown in Figure 3. In the scalar case, state space equation can describe any specific filter structure. But, in the BSS filter, the state decimation destroys the specific structure and the BSS filter does not have distinctive structure. The structures of the matrices $\hat{A}$, $\hat{B}$, $\hat{C}$ and $\hat{D}$, however, are dependent on the chosen prototype scalar filter structure.

3. BLOCK LATTICE FILTERS

The normalized Gray-Markel lattice filter uses the orthogonal expansion for the denominator polynomial, whereas simple linear combination is used for the numerator polynomial.

For our block IIR lattice filter structure implementation, we choose the normalized Gray-Markel lattice model which is the original scalar lattice filter structure. Figure 4 illustrates 4th order IIR normalized Gray-Markel lattice filter structure. [5]

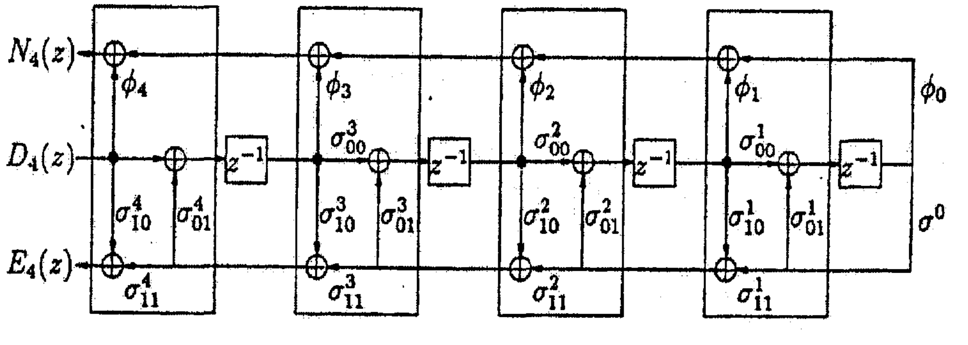

Figure 4. Normalized Gray-Markel Lattice Filter

In Figure 4, the desired transfer function is given by $H(z) = N_4(z)/D_4(z)$, i.e., numerator and denominator polynomials are given by $N_4(z)$ and $D_4(z)$, respectively. In Gray-Markel lattice filter, the denominator polynomial was implemented such that it relates $E_4(z) = z^4 D_4(z^{-1})$, i.e., an allpass realization, and the numerator polynomial was implemented using polynomial expansion.

For the block IIR lattice filter implementation, the block transfer function obtained from the scalar transfer function is required. Block transfer function with right-MFD is given by

$$H(z) = N_M(z) D_M^{-1}(z) \tag{4}$$

where

$$D_M(z) = A_{M,0} z^M + A_{M,1} z^{M-1} + \cdots + A_{M,M}$$

$$N_M(z) = B_{M,0} Z^M + B_{M,1} z^{M-1} + \cdots + B_{M,M}$$

A_i and B_i is the block filter coefficient and M is the block filter order. From the order reduction step utilizing the chain parameter description, the block denominator polynomials D_i, E_i are decomposed as shown in Figure 5.

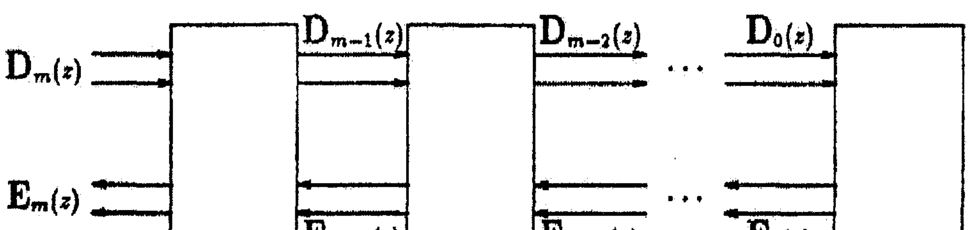

Figure 5. Lattice implementation for denominator

Using the relationships between the scattering and the chain matrix elements, the elements of the scattering block matrix are given [6] as

$$\sum_{00}^{m} = [I - K_m^t \ K_m]^{t/2},$$

$$\sum_{01}^{m} = [I - K_m^t \ K_m]^{-1/2} \ K_m^t [I - KK_m^t]^{1/2},$$

$$\Sigma_{10}^{m} = -K_m, \quad \text{and}, \quad \Sigma_{11}^{m} = [I - K_m K_m^t]^{1/2}$$

where $K_m = -B_{m,0} A_{m,0}^{-1}$

Here, the first stage of the denominator synthesis is complete. The procedure can be repeated until the resulting matrix polynomials become the zeroth order, i.e., constant matrices. Then the design procedure is terminated by a constant matrix $R = E_0(z) D_0^{-1}(z)$ connected between the two nodes represented by $D_0(z)$ and $E_0(z)$ as shown in Figure 6.

The numerator matrix polynomial can be expanded using a set of polynomials $D_m(z)$, m= 0, 1,⋯, M, obtained during the denominator BLBR function synthesis, i.e., $N_M(z) = \sum_{k=0}^{M} \Phi_k D_k(z)$. The expansion coefficients, Φ_k, k= 0, 1,⋯, M are obtained by comparing the coefficients of the numerator polynomial with equal powers of z.

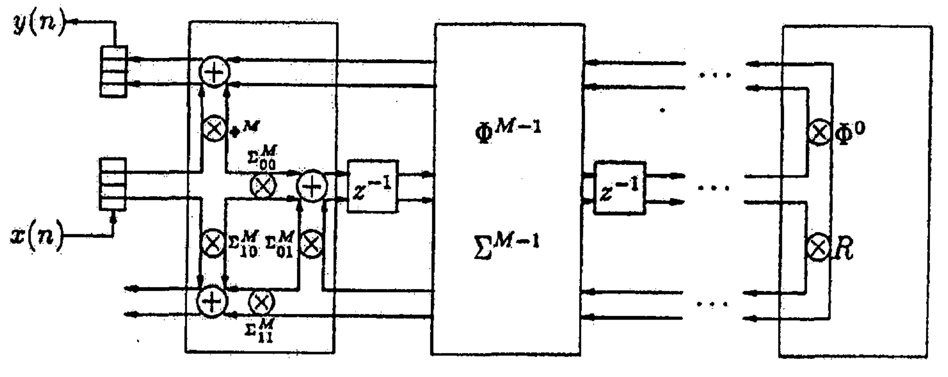

Figure 6. Proposed Block IIR Lattice Filter Structure

4. EXPERIMENT RESULTS

We compare the IIR block lattice with a BSS approach based on a cascaded normal structure given in [7] which is known to have very good finite precision characteristics.

For this purpose, consider the design of a sixth-order lowpass Chebyshev type I.

$$H(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_6 z^{-6}}{a_0 + a_1 z^{-1} + \cdots + a_6 z^{-6}}$$

Where,

$b_0 = 0.00006284$; $b_1 = 0.00037703$; $b_2 = 0.00094258$;

$b_3 = 0.00125677$; $b_4 = 0.00094258$; $b_5 = 0.00037703$;

$b_6 = 0.00006284$;

$a_0 = 1.0$; $a_1 = -4.73500775$; $a_2 = 9.84166914$

$a_3 = -11.42125185$; $a_4 = 7.77931623$; $a_5 = -2.94382481$

$a_6 = 0.48335901$;

For a cascade implementation, three second-order sections of transfer function are formed as

$$H_1(z) = \frac{1 + 2.0543z^{-1} + 1.0553z^{-2}}{1 - 1.54z^{-1} + 0.9129z^{-2}}$$

$$H_2(z) = \frac{1 + 1.991z^{-1} + 1.0001z^{-2}}{1 - 1.5627z^{-1} + 0.7703z^{-2}}$$

$$H_3(z) = \frac{1 + 1.9465z^{-1} + 0.9475z^{-2}}{1 - 1.6324z^{-1} + 0.6874z^{-2}}$$

where $H(z) = H_1(z)\ H_2(z)\ H_3(z)$.

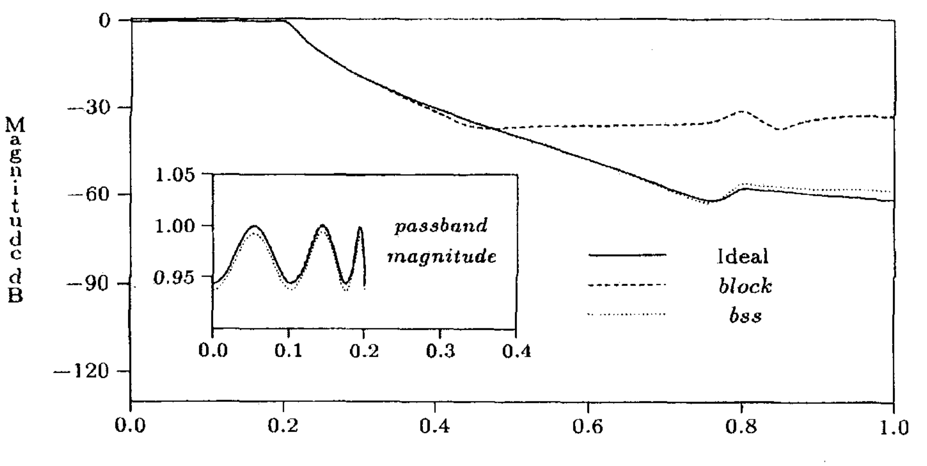

Figure 7. Magnitude responses for 11-bit quantization, Block lattice and BSS (Normal structure prototype)

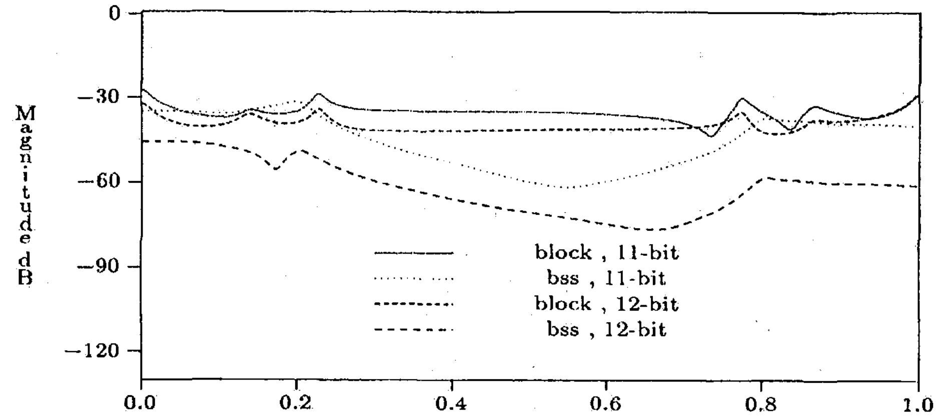

Figure 8. Magnitude responses for Block lattice and BSS

Each second order stage is first realized in a normal form and then its BSS realization is found. The overall block implementation is obtained by cascading each subsection BSS implementations. L=2 is chosen for reduced complexity of presentation. This implementation is compared with the block lattice implementation with L=2. From Figure 7, it can be seen that the $G_0(e^{jw})$ of block lattice filters are better than that of cascaded BSS filters in the passband but the BSS implementation has better attenuation characteristic. From Figure 8, it can also be seen that the aliasing-like error, $G_1(e^{jw})$ in the block lattice is generally larger than the cascaded BSS implementation.

5. CONCLUSIONS

Finite coefficient precision effects for IIR block lattice and BSS are investigated by computer simulations. It has been shown that block lattice structures provide a viable alternative block filter implementations with relatively small coefficient quantization effects and high throughput. Disadvantages of the proposed block filters is that aliasing-like error tends to deteriorate attenuation characteristics and requires more computations. As in the scalar case, factorized cascade BSS implementation is preferable for reduced coefficient quantization effects. Cascade BSS implementation tends to reduce aliasing-like errors.

6. REFERENCES

[1] T. G. Stockham, "High speed convolution and correlation," 1966 Spring Joint Comput. Conf., AFIPS Conf. Proc., vol. 28. Washington, D. C., 1966.

[2] B. Gold and C. M. Rader, Digital Processing of Signals, McGraw-Hill, New York, 1969.

[3] Y. B. Jang, "Block Digital filter Implementation Techniques and Transform Domain Processing." Dissertation at Polytechnic Univ., 1994.

[4] Uipil Chong and S. P. Kim, "Roundoff error analysis of block digital filters," Proc. KSEA, pp.129-133, Mar. 1996.

[5] J. D. Markel and A. H. Gray, "Fixed-point implementation algorithms for a class of orthogonal polynomial filter structures," IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP-23, NO.5, pp.486-494, Oct. 1975.

[6] S. P. Kim and Y. B. Jang, "Block lattice digital filters," Proc. ECCTD 1993, pp.1395-1400, Switzerland.

[7] Jan Zeman and Allen G. Lindgren, "Fast digital filters with low round-off noise," IEEE Trans. Circuits and Systems, Vol. CAS-28, NO.7 pp.716-23, Jul. 1981.