

1, 2) are unbiased and statistically independent. The frequency resolution bandwidths are given by

$$B_i = \frac{2}{\pi} \frac{1}{(M+1)T_s} \frac{1}{M \cdot \text{SNR}_i}, \quad i = 1, 2 \quad (4.1)$$

and the asymptotic variances

$$E(\hat{f}_i - f_i)^2 = \frac{M}{8 \cdot N} B_i^2. \quad (4.2)$$

### B. Superresolution

If  $|f_1 - f_2| < 1/MT_s$ , but the  $\text{SNR}_i$  are high that

$$2\beta(B_1 + B_2) < 1/MT_s, \quad (4.3)$$

or, equivalently,

$$(4\beta/\pi) \cdot \left[ \frac{1}{M \cdot \text{SNR}_1} + \frac{1}{M \cdot \text{SNR}_2} \right] < 1. \quad (4.4)$$

Equation (4.3) means that two sinusoids are resolvable. However, the biases of the AR frequency estimate,  $\alpha B_i/2$ , exist; and the resolution bandwidths  $\beta B_i$  spread by a factor  $\beta$ . And, the statistical variances are given by

$$E(\hat{f}_i - f_i)^2 = \frac{M|\kappa|^2 \Delta B_i^2}{8N(1 - |\epsilon|^2)} = \frac{M\beta B_i^2}{8N(1 - |\epsilon|^2)}. \quad (4.4)$$

### C. Choices of the AR Model $M$ and Sample Size $N$

$(1/M \cdot \text{SNR}_i)$  should be small, at least to meet the resolvability condition (4.4), and  $N$  should be large that  $(1/N \cdot T_s) < \beta B_i/2$ .

#### ACKNOWLEDGMENT

The author wishes to thank Prof. S. Gutmann of Northeastern University for valuable discussions and suggestions. The comments from the Associate Editor and anonymous reviewers are also deeply appreciated.

#### REFERENCES

- [1] R. T. Lacoss, "Data adaptive spectral analysis method," *Geophysics*, vol. 36, pp. 661-675, Aug. 1971.
- [2] E. H. Satorius and J. R. Zeidler, "Maximum entropy spectral analysis of multiple sinusoids in noise," *Geophysics*, vol. 43, pp. 1111-1118, Oct. 1978.
- [3] H. Sakai, "Statistical properties of AR spectral analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, pp. 402-409, Aug. 1979.
- [4] S. W. Lang and J. H. McClellan, "Frequency estimation with maximum entropy spectral estimator," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 716-724, Dec. 1980.
- [5] S. L. Marple, "Resolution of conventional Fourier, autoregressive, and special ARMA methods of spectrum analysis," presented at the IEEE Int. Conf. ASSP, Hartford, CT, 1977.
- [6] S. Haykin, Ed., *Nonlinear Methods of Spectral Analysis*. Berlin Heidelberg, New York: Springer-Verlag, 1979.
- [7] S. M. Kay and S. L. Marple, Jr., "Spectrum analysis—a modern perspective," *Proc. IEEE*, vol. 69, pp. 1380-1419, Nov. 1981.
- [8] D. G. Childers, Ed., *Modern Spectrum Analysis*. New York: IEEE, 1978.
- [9] S. B. Kesler, *Modern Spectrum Analysis, II*. New York: IEEE, 1986.

## A New Array Architecture for Prime-Length Discrete Cosine Transform

Jiun-In Guo, Chi-Min Liu, and Chein-Wei Jen

**Abstract**—A new approach to derive a systolic algorithm for prime-length discrete cosine transform (DCT) is proposed. It makes use of the input/output (I/O) data permutations and the symmetry property of cosine kernels such that the proposed array possesses outstanding performance in hardware cost of the processing elements (PE's), average computation time, and the I/O cost.

### I. INTRODUCTION

The discrete cosine transform (DCT) has been widely used in image coding for its near-optimal performance [1]. Since the DCT is computation intensive, the development of high-speed hardware is necessary in many real-time applications. Systolic arrays are an appropriate architecture to meet the requirements of both high processing speeds and VLSI implementation. However, the computing algorithms encapsulated within systolic arrays need to be developed specifically.

Recently, there were some systolic array architectures [2]–[6] proposed to realize one-dimensional DCT. These architectures can be categorized into linear array architectures [2]–[4] and two-dimensional array architectures [5], [6]. Although the two-dimensional arrays can attain higher speeds than one-dimensional arrays, the hardware complexity of PE's and the control complexity of these two-dimensional arrays are generally higher than those of linear arrays. Furthermore, the two-dimensional arrays need high I/O bandwidth and a large number of I/O channels to attain the higher speeds, unless most operands are preloaded into the arrays instead of being supplied from the input ports. But additional overheads are needed if the operands are preloaded into the arrays like the two-dimensional array in [5]. Considering for example the array in [6], the average computation time for  $N$ -point DCT is  $(\sqrt{N} + 2)$  cycles, while the number of multipliers in the array is  $(4N + 4\sqrt{N})$ , if the clock cycle is assumed to be the consumption time of one multiplier. In addition, undesirable features such as the complex control problems, high I/O bandwidth, and a large number of I/O channels are still accompanied with the array in [6].

The attractive feature of linear arrays is that the I/O bandwidth and the number of I/O channels can be kept independent of the DCT length if the I/O channels exist only at the two extreme ends of a linear array. As discussed in [8], the high I/O bandwidth required for most systolic arrays would limit computing speeds. Hence, linear arrays should be one feasible architecture for a sys-

Manuscript received January 18, 1991; revised November 4, 1991. Part of this correspondence was presented at the IEEE Workshop on Visual Signal Processing and Communications, June 6–7, 1991. This work was supported by the National Science Council under Grant NSC80-0404-E009-39.

J.-I. Guo is with the Department of Electronics Engineering, Institute of Electronics, National Chiao Tung University, Hsinchu, 30039, Taiwan, Republic of China.

C.-M. Liu is with the Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu, 30039, Taiwan, Republic of China.

C.-W. Jen is with the Department of Electronics Engineering, Institute of Electronics, National Chiao Tung University, Hsinchu, 30039, Taiwan, Republic of China.

IEEE Log Number 9203370.

tem application. However, how to keep I/O channels at the extreme ends of linear arrays and to pursue high computing power at the same time should be a challenging design issue when deriving sys-

$$\begin{bmatrix} Y(0) \\ Y(1) \\ Y(2) \\ Y(3) \\ Y(4) \\ Y(5) \\ Y(6) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \cos(a) & \cos(3a) & \cos(5a) & \cos(7a) & \cos(9a) & \cos(11a) & \cos(13a) & \\ \cos(2a) & \cos(6a) & \cos(10a) & \cos(14a) & \cos(10a) & \cos(6a) & \cos(2a) & \\ \cos(3a) & \cos(9a) & \cos(13a) & \cos(7a) & \cos(a) & \cos(5a) & \cos(11a) & \\ \cos(4a) & \cos(12a) & \cos(8a) & 1 & \cos(8a) & \cos(12a) & \cos(4a) & \\ \cos(5a) & \cos(13a) & \cos(3a) & \cos(7a) & \cos(11a) & \cos(a) & \cos(9a) & \\ \cos(6a) & \cos(10a) & \cos(2a) & \cos(14a) & \cos(2a) & \cos(10a) & \cos(6a) & \end{bmatrix} \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \end{bmatrix} \quad (2)$$

temic algorithms for linear arrays. The approach in [2] is to directly represent the DCT as a matrix-vector multiplication first. Then, the systolic array realization for the matrix-vector multiplication can be directly modified to compute the DCT. Since the designed array in [2] cannot retain the I/O channels at the two extreme ends of itself, a large number of I/O channels and high I/O bandwidth are needed. Another approach [3] modifies the DCT into a form similar to the discrete Fourier transform (DFT) and realizes the DCT by using the array that has been developed for the DFT. Since the twiddle factor  $\exp(j2\pi/N)$  in the DFT is a complex number while the factor  $\cos(2\pi/4N)$  in the DCT is a real number, the designed arrays based on this approach should induce much hardware cost. In addition, the approach in [4] is also to represent the DCT as a matrix-vector multiplication like [2], but it generates the transform kernels recursively in the array instead of prestoring them in memory. The array in [4] uses this method to reduce the I/O cost such as the number of I/O channels and I/O bandwidth, but additional hardware cost is paid for recursive generations of the cosine kernels.

To simultaneously consider the hardware cost, the I/O bandwidth, and the number of I/O channels, a systolic algorithm for prime length DCT is derived in this correspondence. The design approach utilizes the input and output data permutations accompanied with the symmetry property of the cosine kernels such that the proposed array can retain most I/O channels at the two extreme ends and simultaneously attain good performance in average computation time, hardware cost of the PE's, and the number of the PE's. The performance of the proposed array and that of the linear arrays in [2]–[4] are discussed in Section III. From Section III, we can see that the proposed array possesses better performance than the arrays [2], [3] in the hardware cost of the PE's, the average computation time, the number of I/O channels, and the I/O bandwidth. Moreover, it also possesses better performance than the array [4] in the hardware cost of the PE's. The overheads of the proposed array include some additional shift registers, latches, multiplexers, a demultiplexer, and a switching element for solving control problems. Basically, these overheads are minor as compared with the savings in regard to the hardware cost of the PE's in the array. This correspondence is arranged as follows. Section II describes the derivation of the computing algorithm encapsulated in the array. Section III considers the array realization of the proposed systolic algorithm. A brief conclusion is given in Section IV.

## II. THE ALGORITHM DERIVATION

The DCT is defined as

$$Y(k) = \sum_{i=0}^{N-1} y(i) \cos \left[ \frac{2\pi(2i+1)k}{4N} \right], \quad \text{for } k = 0, 1, \dots, N-1 \quad (1)$$

where  $\{y(i)|i = 0, 1, \dots, N-1\}$  is the input sequence and  $\{Y(k)|k = 0, 1, \dots, N-1\}$  is the output sequence. We represent (1) as a matrix-vector multiplication as follows:

where "a" denotes  $\pi/14$ ; and  $N$  is assumed to be 7. If (2) is directly realized by linear array architectures, as was done in [2], there would be one input port needed in every PE to transmit the cosine kernels for proper operations, and would induce a large number of I/O channels and high I/O bandwidth. It can be shown that the DCT defined in (1) can be formulated as

$$Y(k) = \{2T(k) + x(0)\} \cos \left[ \frac{k\pi}{2N} \right], \quad \text{for } k = 0, 1, \dots, N-1. \quad (3)$$

where

$$T(k) = \sum_{i=1}^{N-1} x(i) \cos \left[ \frac{\pi ik}{N} \right], \quad \text{for } k = 0, 1, \dots, N-1. \quad (4)$$

and  $x(i)$  is another sequence defined as

$$\begin{aligned} x(N-1) &= y(N-1) \\ x(i) &= y(i) - x(i+1) \quad \text{for } i = 0, 1, \dots, N-2. \end{aligned} \quad (5)$$

If  $N$  is a prime number, there exists some number of "g," not necessarily unique, such that there is a one-to-one mapping from integers  $\{i|i = 1, 2, \dots, N-1\}$  to integers  $\{j|j = 1, 2, \dots, N-1\}$ , given by

$$j = |g^i|_N \quad (6)$$

where  $|A|_N$  denotes the result of "A-modulo-N" operation. Then (4) can be reformulated with  $i$  and  $k$  as powers of the primitive element "g." Because  $i$  and  $k$  take on the value zero, and zero is not a power of "g," the zero frequency component must be treated specially, i.e.,

$$Y(0) = \sum_{i=0}^{N-1} y(i) \quad (7)$$

$$Y(k) = \{2T(k) + x(0)\} \cos \left[ \frac{k\pi}{2N} \right], \quad \text{for } k = 1, \dots, N-1. \quad (8a)$$

where

$$T(k) = \sum_{i=1}^{N-1} x(i) \cos \left[ \frac{\pi ik}{N} \right], \quad \text{for } k = 1, \dots, N-1. \quad (8b)$$

Applying (6) to (8b), it follows that

$$\begin{aligned} T'(k) &= T(|g^k|_N) \\ &= \sum_{i=1}^{N-1} x'(i) \times \cos \left[ \frac{\pi}{N} \times |g^i|_N \times |g^k|_N \right], \\ & \quad k = 1, 2, \dots, N-1. \end{aligned} \quad (9a)$$

The term " $|g^i|_N \times |g^k|_N$ " can be expressed as

$$|g^i|_N \times |g^k|_N = |g^{i+k}|_N + m \times N, \quad (9b)$$

$$i, k = 1, 2, \dots, N-1.$$

where " $m$ " is an integer. Then, (9a) can be written as

$$T'(k) = T(|g^k|_N) = \sum_{i=1}^{N-1} x'(i) \times C_k^i, \quad (9c)$$

$$k = 1, 2, \dots, N-1$$

where

$$C_k^i = \begin{cases} \cos\left(\frac{\pi}{N} |g^{i+k}|_N\right), & \text{if } m \text{ is even} \\ -\cos\left(\frac{\pi}{N} |g^{i+k}|_N\right), & \text{if } m \text{ is odd} \end{cases},$$

$$\text{and } x'(i) = x(|g^i|_N).$$

Now (7), (8a), and (9c) constitute the computational equations for the DCT. To see the difference between these computational equations and (1), (9c) is written as

$$\begin{bmatrix} T'(1) \\ T'(2) \\ T'(3) \\ T'(4) \\ T'(5) \\ T'(6) \end{bmatrix} = \begin{bmatrix} T(3) \\ T(2) \\ T(6) \\ T(4) \\ T(5) \\ T(1) \end{bmatrix} = \begin{bmatrix} -\cos(2a) & \cos(6a) & \cos(4a) & -\cos(5a) & \cos(a) & \cos(3a) \\ \cos(6a) & \cos(4a) & -\cos(5a) & -\cos(a) & -\cos(3a) & \cos(2a) \\ \cos(4a) & -\cos(5a) & -\cos(a) & -\cos(3a) & \cos(2a) & \cos(6a) \\ -\cos(5a) & -\cos(a) & -\cos(3a) & \cos(2a) & \cos(6a) & \cos(4a) \\ \cos(a) & -\cos(3a) & \cos(2a) & \cos(6a) & -\cos(4a) & \cos(5a) \\ \cos(3a) & \cos(2a) & \cos(6a) & \cos(4a) & \cos(5a) & \cos(a) \end{bmatrix} \begin{bmatrix} x'(1) \\ x'(2) \\ x'(3) \\ x'(4) \\ x'(5) \\ x'(6) \end{bmatrix} \quad (10)$$

where " $a$ " denotes  $\pi/7$ ,  $N$  and " $g$ " are assumed to be 7 and 3, respectively. It can be seen that the absolute values of the cosine kernels along same antidiagonal positions in the matrix of (10) are the same while those in the matrix of (2) do not have any specific order like (10). This phenomenon tells that the vector of  $T'(k)$  is the circular convolution of inputs  $x'(i)$  and the cosine kernels. The phenomenon also exists in the DFT, which was firstly found by Rader [10] and has also been used to design the efficient systolic arrays for prime length DFT [9]. Now we apply it to derive the systolic algorithm for DCT. From the viewpoint of array realization, the constant value along the same antidiagonal positions means that this variable can be sent to every PE along a link from one input port at the extreme end of a linear array. The  $(2N-3)$  antidiagonal lines in the matrix of (10) mean that there are only  $(2N-3)$  values instead of  $N^2$  values in the matrix of (2) needed to be sent to the array. This phenomenon can be effectively captured to design the systolic array with a low number of I/O channels and low I/O bandwidth.

From (10), since  $\cos(k\pi/N) = -\cos((N-k)\pi/N)$ , it is observed that the absolute values of the cosine kernels located at the left three columns are the same as those located at the right three columns. This symmetry property benefits further reduction of the computational complexity in the algorithm. As shown in the Appendix, the symmetry property of the cosine kernels can be expressed as the following equation:

$$\begin{aligned} \cos\left[\frac{\pi}{N} |g^i|_N\right] &= \cos\left[\frac{\pi}{N} (N - |g^{i+(N-1)/2}|_N)\right] \\ &= -\cos\left[\frac{\pi}{N} (|g^{i+(N-1)/2}|_N)\right]. \end{aligned} \quad (11)$$

Applying (11) to (9c), (9c) can be written as

$$T'(k) = \sum_{i=1}^{(N-1)/2} x''(i) \times C_k^i, \quad k = 1, 2, \dots, N-1. \quad (12a)$$

where

$$x''(i) = \begin{cases} x'(i) + x'\left(i + \frac{N-1}{2}\right), & \text{if } m1 \text{ and } m2 \text{ are one even number and} \\ & \text{one odd number} \\ x'(i) - x'\left(i + \frac{N-1}{2}\right), & \text{if } m1 \text{ and } m2 \text{ are all even numbers or} \\ & \text{all odd numbers} \end{cases}$$

and

$$C_k^i = \begin{cases} \cos\left(\frac{\pi}{N} |g^{i+k}|_N\right), & \text{if } m1 \text{ is even} \\ -\cos\left(\frac{\pi}{N} |g^{i+k}|_N\right), & \text{if } m1 \text{ is odd} \end{cases};$$

$$i = 1, \dots, \frac{N-1}{2}, \text{ and } k = 1, \dots, N-1.$$

The integers  $m1$  and  $m2$  are determined in the following equations:

$$\begin{cases} |g^i|_N \times |g^k|_N = |g^{i+k}|_N + m1 \times N \\ |g^{i+(N-1)/2}|_N \times |g^k|_N = |g^{i+k+(N-1)/2}|_N + m2 \times N \end{cases};$$

$$k = 1, 2, \dots, N-1, \text{ and } i = 1, \dots, \frac{N-1}{2}. \quad (12b)$$

where

$$|g^{i+k}|_N + |g^{i+k+(N-1)/2}|_N = N.$$

Now (7), (8a), and (12) constitute the computational equations of the DCT in the proposed algorithm. Considering the computational complexity, the number of multiplications has been reduced from  $(N-1)^2$  in (9c) to  $(N-1)^2/2$  in (12a). In addition, the vector of  $T'(k)$  in (12a) is still in a circular convolution form. It will be shown in the next section that such a form is beneficial to the reduction of I/O cost.

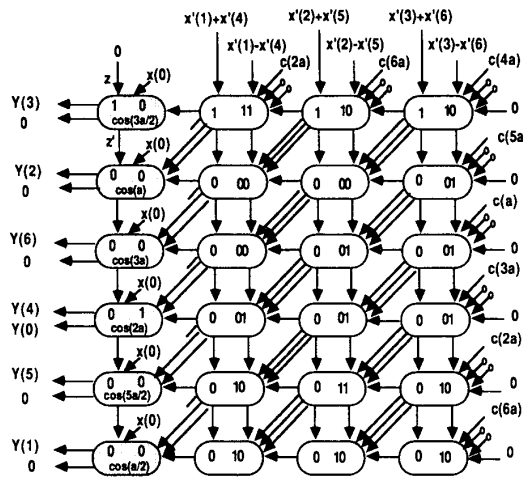


Fig. 1(a). The dependence graph (DG) of the proposed algorithm for 7-point DCT where ‘‘a’’ denotes  $\pi/7$ .

III. THE ARRAY REALIZATION

This section considers the array realization of the proposed systolic algorithm. Fig. 1 shows the dependence graph (DG) [12] of the proposed algorithm for a seven-point DCT. The DG clearly shows the data operations, data dependency, and control signals involved in the proposed algorithm. Linear arrays can be constructed from the DG according to the design procedure [12]. And the tag control scheme [13] can be utilized for the I/O control and data control. Based on the two design approaches, Fig. 2 shows the constructed array for seven-point DCT with projection vector [0 1]. For the sake of showing the activity of the array clearly, we rewrite (7), (8a), and (12a) in recursive forms as

$$\begin{aligned}
 z_0^0 &= x(0) \\
 z_0^i &= z_0^{i-1} + 2 \times [x'(i) + x'(i + 3)], \\
 & \quad i = 1, 2, 3, \\
 Y(0) &= z_0^3 \\
 Y'(k) &= \{2T'(k) + x(0)\} \times \cos\left(\frac{\pi}{14} |3^k|_7\right), \\
 & \quad k = 1, \dots, 6, \\
 y_0^0 &= 0 \\
 y_k^i &= y_k^{i-1} + x''(i) \times C_k^i, \\
 & \quad i = 1, 2, 3, \quad k = 1, \dots, 6, \\
 T^*(k) &= y_k^3
 \end{aligned}
 \tag{13a}$$

where

$$C_k^i = \begin{cases} \cos\left(\frac{\pi}{7} |3^{i+k}|_7\right), & \text{if } m1 \text{ is even} \\ -\cos\left(\frac{\pi}{7} |3^{i+k}|_7\right), & \text{if } m1 \text{ is odd} \end{cases}$$

and ‘‘ $y_k^i$ ’’ and ‘‘ $z_0^i$ ’’ are the intermediate results.

From Fig. 2(a), we know that the operations specified in (13a) and (13b) are computed within the left-most PE, while those in (13c) are computed in other PE’s. The multiplication and addition

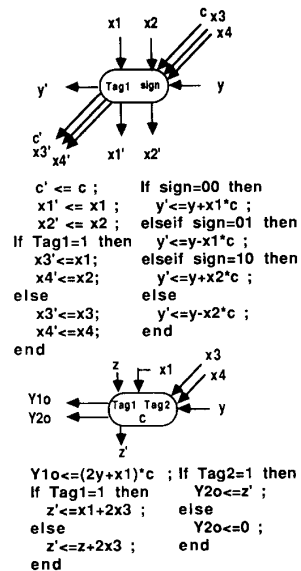
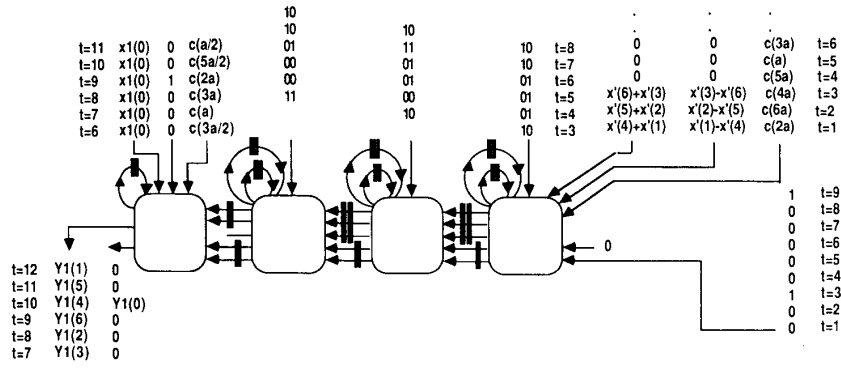
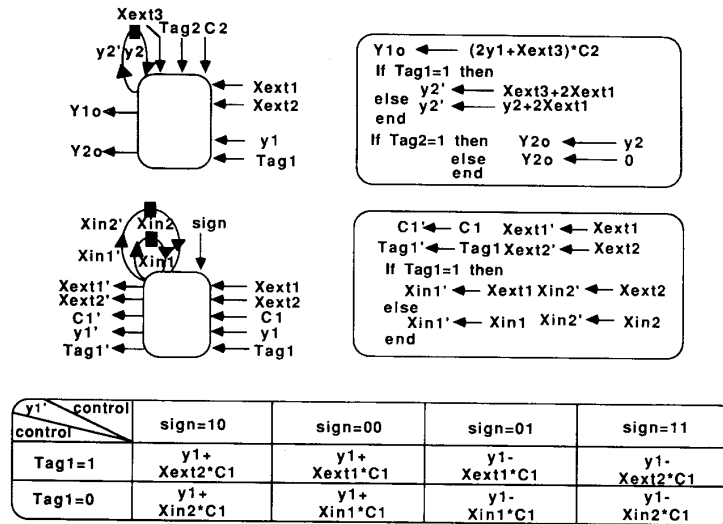


Fig. 1(b). The functions of nodes.

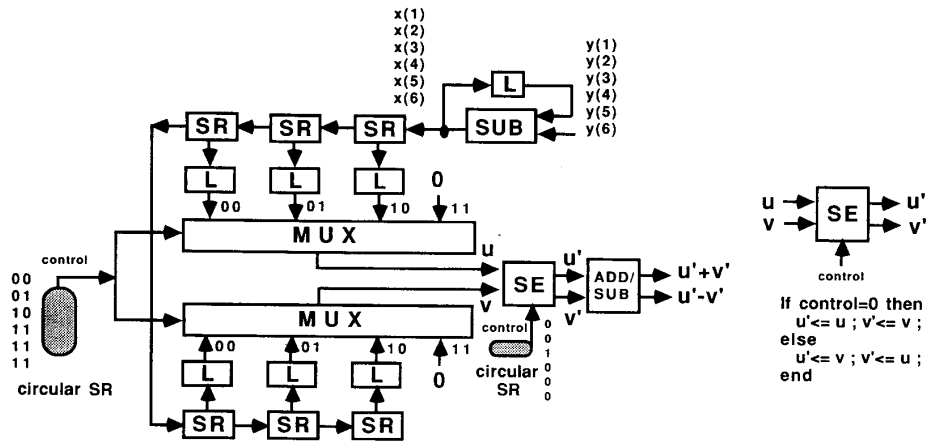
constitute the main functions of the PE’s, which are shown in Fig. 2(b). And three control signals denoted as ‘‘Tag1,’’ ‘‘Tag2,’’ and ‘‘sign’’ are used to select the right operands in the operations. Fig. 2(c) shows the preprocessing stage needed in the array. The intermediate sequence  $x(i)$  can be generated from input sequence  $y(i)$  by a subtractor, and then we use the multiplexers and a switching element to permute the sequence  $x(i)$  where the required control signals can be generated by circular shift registers. Finally, the required data patterns are obtained by adding and subtracting the permuted data. Fig. 2(d) shows the postprocessing stage in the array, which uses a demultiplexer to perform the output data permutation. Similarly, the control signals needed in the demultiplexer can be generated by a circular shift register. The utilization of shift registers and latches in Fig. 2(c) and Fig. 2(d) makes the array able to be pipelined. That is, the intermediate signals  $x(i)$  and output results  $Y(k)$  of current block are shifted into the shift registers seriously. After all of these  $(N - 1)$  values have been



(a)



(b)



(c)

Fig. 2. (a) The array architecture for 7-point DCT where "c()" denotes cos() and "a" denotes  $\pi/7$ . (b) The functions of the PE's in the array. (c) The preprocessing stage in the array where SR denotes shift register, SE denotes switching element, and L denotes latch. (Continued on next page.)

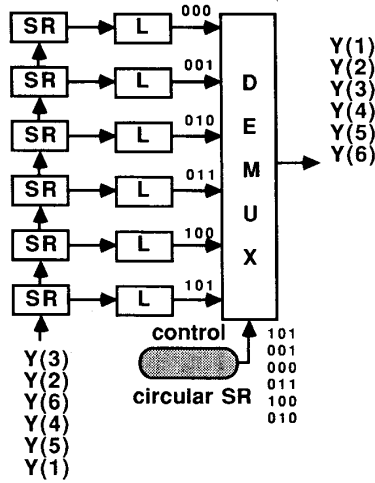


Fig. 2. (Continued) (d) The postprocessing stage in the array where SR denotes shift register and L denotes latch.

shifted into the registers, they are shifted parallelly into the latches for the I/O data permutations such that the data of next block can be continuously shifted into the registers without any time delay. Therefore, the proposed array including the preprocessing and postprocessing stages can be fully pipelined, and a high throughput rate of the design can be attained.

In order to see the features of the proposed array more clearly, (12a) is expressed as

$$\begin{aligned}
 \begin{bmatrix} T'(1) \\ T'(2) \\ T'(3) \\ T'(4) \\ T'(5) \\ T'(6) \end{bmatrix} &= \begin{bmatrix} T(3) \\ T(2) \\ T(6) \\ T(4) \\ T(5) \\ T(1) \end{bmatrix} \\
 &= \begin{bmatrix} -\cos(2a) & \cos(6a) & \cos(4a) \\ \cos(6a) & \cos(4a) & -\cos(5a) \\ \cos(4a) & -\cos(5a) & -\cos(a) \\ -\cos(5a) & -\cos(a) & -\cos(3a) \\ \cos(a) & -\cos(3a) & \cos(2a) \\ \cos(3a) & \cos(2a) & \cos(6a) \end{bmatrix} \\
 &\quad \cdot \begin{bmatrix} x'(1) \pm x'(4) \\ x'(2) \pm x'(5) \\ x'(3) \pm x'(6) \end{bmatrix} \quad (14)
 \end{aligned}$$

where "a" denotes  $\pi/7$ ,  $N$  and "g" are assumed to be 7 and 3, respectively. If "k" is equal to 1, 5, and 6, the minus signs in the input vector are valid. Otherwise, plus signs are valid. As shown in (14), there are  $(3N - 5)/2$  values needed to be sent to the array for computing the  $N$ -point DCT. And  $C = \{\cos(2a), \cos(6a), \cos(4a), \cos(5a), \cos(a), \cos(3a), \cos(2a), \cos(6a)\}$  is the sequence of these eight values for the seven-point DCT. It is observed that the last two cosine kernels are identical to the first two cosine kernels in  $C$ . And these common cosine kernels can be shared for computing two neighboring blocks successively. As many image blocks are processed continuously, it is only necessary to send six

values instead of eight to the array for computing each seven-point DCT. It can be seen from the array in Fig. 2(a) that only  $(N - 1)$  cosine kernels are needed to compute an  $N$ -point DCT. And, the average computation time for computing the  $N$ -point DCT is  $(N - 1)$  cycles. This phenomenon is induced from the cyclic property of the modulo operation in (6), i.e.,  $|g^i|_N = |g^{N-1-i}|_N$ .

Exerting the specific order of the cosine kernels in the matrix of (14), these kernels in the array are imported from the right-most PE instead of being imported from every PE as the approach in [2]. Therefore, the proposed array requires a low number of I/O channels and low I/O bandwidth. Considering the I/O cost, the I/O cost of the designs [2]-[4] are proportional to  $(N + 2)L$  [2],  $(N + 3)L$  [3], and  $8L$  [4] where  $L$  is the wordlength. And, the I/O cost of the proposed array is only proportional to  $7L + N + 2$ . Also, the proposed array needs much lower hardware cost than the designs [2]-[4]. The required numbers of multipliers are  $N$  [2],  $4N + 4$  [3], and  $2N - 2$  [4], which are much larger than the  $(N + 1)/2$  of the proposed array. Moreover, regarding to the average computation time, the proposed array needs  $(N - 1)$  cycles for computing  $N$ -point DCT, which is better than the  $N$  cycles in [2], and also better than the  $(N + 1)$  cycles in [3]. The hardware overheads of the proposed array include some shift registers, latches, multiplexers, a demultiplexer, and a switching element for solving the control problems and the I/O data permutations. And the cycle time of the array includes the multiplication and addition time as well as the time for multiplexing. However, these overheads are minor as compared with the savings of hardware cost in the proposed array. As a whole, the proposed array excels the arrays [2], [3] in average computation time, hardware cost of PE's, the number of I/O channels, and the I/O bandwidth. It also excels the array [4] in hardware cost of the PE's.

#### IV. CONCLUSIONS

In this correspondence, a new approach to derive the systolic algorithm for prime length DCT is presented. This approach induces the array to have good performance in hardware cost of PE's, average computation time, the number of I/O channels, and the I/O bandwidth. Also, this design approach can be similarly applied to derive the systolic algorithms for discrete sine transform (DST) and discrete Fourier transform (DFT) [9]. Although the proposed systolic algorithm and array are derived under the restriction that  $N$  is a prime number, they can be applied to the nonprime length DCT by appending the input data from nonprime length to prime length at the expense of some overheads in hardware cost and average computation time. With these overheads, the hardware cost of the proposed array is still lower than that in the arrays [2]-[4]. However, it is not always a drawback that  $N$  is a prime number. It is known that the blocking effect will occur in the DCT as applied to image coding with low bit rate. And the overlapping method is one of the remedies for this problem [11]. Applying the proposed algorithm to the nonprime length DCT by using the overlapping method can also reduce the undesirable blocking effect.

#### APPENDIX

In the Appendix, the proof of (11) is given. At first, (11) is rewritten here as

$$\begin{aligned}
 \cos \left[ \frac{\pi}{N} |g^i|_N \right] &= \cos \left[ \frac{\pi}{N} (N - |g^{i+(N-1)/2}|_N) \right] \\
 &= -\cos \left[ \frac{\pi}{N} (|g^{i+(N-1)/2}|_N) \right], \\
 1 \leq i \leq N - 1. \quad (A1)
 \end{aligned}$$

The necessary and sufficient condition that (A1) holds is

$$|g^i|_N = N - |g^{i+(N-1)/2}|_N.$$

That is

$$|g^i|_N + |g^{i+(N-1)/2}|_N = N \quad (\text{A2})$$

where "g" is a primitive element. According to the number theory [7], we have

$$|g^{(N-1)/2}|_N = N - 1; \quad |A \times B|_N = |A|_N \times |B|_N$$

then

$$\begin{aligned} |g^{i+(N-1)/2}|_N &= |g^i \times g^{(N-1)/2}|_N = ||g^i|_N \times |g^{(N-1)/2}|_N|_N \\ &= ||g^i|_N \times (N-1)|_N = |-|g^i|_N|_N \\ &= |N - |g^i|_N|_N \end{aligned}$$

as  $0 < |g^i|_N \leq N-1$ ,  $1 \leq i \leq N-1$ , we have

$$|N - |g^i|_N|_N = N - |g^i|_N.$$

It means that

$$|g^{i+(N-1)/2}|_N = |N - |g^i|_N|_N = N - |g^i|_N.$$

So

$$|g^i|_N + |g^{i+(N-1)/2}|_N = N.$$

Therefore, (11) is proved.

#### ACKNOWLEDGMENT

The authors are very grateful to the reviewers for their constructive comments.

#### REFERENCES

- [1] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, pp. 90-93, Jan. 1974.
- [2] U. Totzek and F. Matthiesen, "Two-dimensional discrete cosine transform with linear systolic arrays," in *Proc. Int. Conf. Systolic Arrays*, Ireland, 1989, pp. 388-397.
- [3] N. I. Cho and S. U. Lee, "DCT algorithms for VLSI parallel implementations," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 1, pp. 121-127, Jan. 1990.
- [4] L. W. Chang and M. C. Wu, "A unified systolic array for discrete cosine and sine transforms," *IEEE Trans. Signal Processing*, vol. 39, no. 1, pp. 192-194, Jan. 1991.
- [5] C. Chakrabarti and J. Ja'Ja', "Systolic architectures for the computation of the discrete Hartley and the discrete cosine transforms based on prime factor decomposition," *IEEE Trans. Comput.*, vol. 39, no. 11, pp. 1359-1368, Nov. 1990.
- [6] M. H. Lee, "On computing 2-D systolic algorithm for discrete cosine transform," *IEEE Trans. Circuits Syst.*, vol. 37, no. 10, pp. 1321-1323, Oct. 1990.
- [7] Shu Lin and Daniel J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1983, Chapter 2, Section 2.2, pp. 19-24.
- [8] A. L. Fisher and H. T. Kung, "Special-purpose VLSI architectures: general discussions and a case study," in *VLSI and Modern Signal Processing*, S. Y. Kung et al., Eds. Englewood Cliffs, NJ: Prentice-Hall, 1985, Chapter 8, pp. 154-169.
- [9] C. M. Liu and C. W. Jen, "A new systolic array algorithm for discrete Fourier transform," *IEEE Trans. Comput.*, 1990, submitted for publication, also in *Proc. Int. Symp. on Circuits and Systems*, Singapore, 1991.
- [10] C. M. Rader, "Discrete Fourier transforms when the number of data samples is prime," *Proc. IEEE*, vol. 56, 1968, pp. 1107-1108.
- [11] H. C. Reeve, III, and J. R. Lim, "Reduction of blocking effect in image coding," in *Proc. ICASSP 83*, Boston, MA, 1983, pp. 1212-1215.
- [12] S. Y. Kung, *VLSI Array Processors*. Englewood Cliffs, NJ: Prentice-Hall, 1988, Chapters 3 and 4, pp. 110-282.
- [13] C. W. Jen and H. Y. Hsu, "The design of a systolic array with tags input," in *Proc. ISCAS*, Finland, 1988, pp. 2263-2266.

## Utilization of Bandpass Filtering for the Matrix Pencil Method

Fengduo Hu, T. K. Sarkar, and Yingbo Hua

**Abstract**—This correspondence describes an algorithm named the bandpass matrix pencil (BPMP) method for estimating the parameters of an exponential data sequence. The matrix pencil (MP) method, along with a filtering technique, is used to estimate the complex exponentials of the signal. However, due to special requirements to the filtered data by the MP method, the prefiltering process is not trivial. The approach presented here utilizes the backward process for the IIR filtering and the circular convolution for the FIR filtering, respectively. Monte Carlo simulations are presented to illustrate the performance of the proposed filtering schemes.

#### I. INTRODUCTION

The mathematical model of an observed signal can generally be formulated as

$$y(k) = x(k) + n(k) = \sum_{i=1}^M R_i Z_i^k + n(k),$$

$$k = 0, 1, \dots, N-1 \quad (1)$$

where

$$z_i = \exp(-\alpha_i + j\omega_i) \quad (2)$$

and  $z_i$ 's and  $R_i$ 's are the poles and residues of the signal, respectively.  $M$  is the number of poles of the signal, and  $n(k)$  is the background noise.  $\alpha_i$  and  $\omega_i$  are the damping factor and angular frequency of the  $i$ th sinusoid, respectively. Once the number of poles and their values have been determined, the residues at the poles can be found by the least squares method. Hence, only the problem of estimation of the poles is considered in this correspondence.

The most popular method for pole retrieval is Prony's method. However, Prony's method is notorious for its extreme sensitivity to noise. There are many modified versions of the Prony method. The most well known one is the principal eigenvector (PE) method [1]. Recently, Hua and Sarkar [2], [3] developed a new technique, named the matrix pencil (MP) method, for pole estimation. The advantage of using matrix pencil is that the signal poles can be found directly from the eigenvalues of the matrix contrast to the PE method, which generally requires two-step processes. In the first step one solves a matrix equation, and finds the roots of a polynomial equation in the second step.

Manuscript received August 10, 1989; revised October 24, 1991.

F. Hu is with Entropic Speech Inc., Cupertino, CA 95014.

T. K. Sarkar is with the Department of Electrical and Computer Engineering, Syracuse University, Syracuse, NY 13244-1240.

Y. Hua is with the Department of Electrical Engineering, University of Melbourne, Parkville, Victoria, Australia 3052.

IEEE Log Number 9203378.