論文 / 著書情報
Article / Book Information

Title	Fast FIR Digital Filter Structures Using Minimal Number of Adders and Its Application to Filter Design
Authors	Mitsuhiko Yagyu, Akinori Nishihara, Nobuo Fujii
Citation	IEICE Trans. Fundamentals, Vol. E79-A, No. 8, pp. 1120-1129
Pub. date	1996, 8
URL	http://search.ieice.org/
Copyright	(c) 1996 Institute of Electronics, Information and Communication Engineers

IEICE TRANS. FUNDAMENTALS, VOL. E79–A, NO. 8 AUGUST 1996

PAPER Special Section on Digital Signal Processing

Fast FIR Digital Filter Structures Using Minimal Number of Adders and Its Application to Filter Design

Mitsuhiko YAGYU†, *Student Member*, Akinori NISHIHARA†, *and* Nobuo FUJII†, *Members*

SUMMARY This paper proposes fast FIR digital filter structures using the minimal number of adders. Filter coefficients are expressed with canonic signed digit (CSD) code and Hartley's technique is used to minimize the number of adders and subtractors. The proposed filters implemented as wired logic are fast because the structure having the shortest critical path is selected. Two algorithms are given to obtain such fast structures. In many examples the critical path length of the filter structures obtained using the proposed method is equal to that of the conventional CSD structures. This paper also presents a new design method of FIR filters using the mixed integer programming (MILP). Utilization of common expressions in Hartley's technique widens the CSD coefficient space. Thus the MILP may lead to better frequency responses. Superior frequency responses are actually obtained in many simulations.

key words: *FIR digital filter, CSD expression, MILP*

1. Introduction

FIR digital filters realized as the hard wired-logic have advantages in respect to processing speed. The hardware cost can be reduced if the multipliers are replaced with some wired shifters, adders and subtractors on the CSDEs (canonic signed digit expressions) of the filter coefficients [1]–[9].

Later several interesting methods [2]–[4] to reduce further the number of adders and subtractors have been proposed. Li implements a multiplier in which the number of adders and subtractors is not more than that of nonzero bits in its CSD [2]. Hartley's technique extracts common CSD expressions (CCSDEs) in all CSD coefficient expressions [3], and the common use of the shift, add and subtract operations corresponding to those CCSDEs reduces the number of adders and subtractors. Hartley's method is more efficient than Li's method, because a whole filter has the minimal number of adders and subtractors. Recently Dempster and Macleod has proposed a method to obtain an optimal filter structure having the minimum number of adders and subtractors [4].

In general the filter structures obtained by using the above method have longer critical paths than the normal CSD structures, which is not analyzed in the above three papers. This paper considers the critical path length as well as the number of adders and subtractors. Firstly an efficient algorithm is presented to find all filter structures whose number of adders and subtractors is minimized by applying Hartley's method which extracts CCSDEs. Hartley's heuristic method is used instead of Dempster and Macleod's optimum method, because it is necessary to obtain all structures in a reasonable time. Then the obtained filters are re-implemented so as to shorten the critical paths. The filter structure which has the shortest critical path is selected as the solution. In many examples the critical paths of such optimized structures are compared with those of normal CSD structures.

The above method reduces the number of adders and subtractors as the frequency response is kept unchanged. If the total number of adders and subtractors is specified, better frequency response could be realized. The mixed integer linear programming (MILP) is often used to design FIR digital filters over discrete coefficient space, because it can obtain the optimal response in the Chebyshev sense [6]. In Lim's design [6], the number of nonzero bits of a CSDE of each coefficient is limited. This limitation determines the coefficient space searched by MILP. The utilization of some specified CCSDEs extends the coefficient space to be searched. Thus a new method using MILP is presented that can design FIR filters over the wider coefficient space. The optimal frequency responses designed in the proposed coefficient space are superior to those designed in the conventional space, which is actually confirmed by many examples.

2. Minimal Number of Adder Structure

2.1 Hartley's Technique for Linear Phase FIR Digital Filters

In this section Hartley's extraction techniques are explained so that it can be applied to obtain the shortest critical path structure. The following 4-tap FIR filter F_1 is given as an example.

$$\begin{aligned} F_1 = {} & 2^{-1} + 2^{-3} + 2^{-5} - 2^{-7} + 2^{-9} \\ & + (-2^{-5} + 2^{-7} - 2^{-9})z^{-1} \\ & + (2^{-7} + 2^{-9})z^{-2} + (2^{-7} - 2^{-9})z^{-3}. \end{aligned} \quad (1)$$

The filter coefficients of F_1 are referred to as $h_0, \cdots, h_3$. Figure 1 shows the CSD structure of F_1. When the number of nonzero bits in all CSDEs of filter coefficients is

Manuscript received December 27, 1995.
Manuscript revised March 7, 1996.

†The authors are with the Faculty of Engineering, Tokyo Institute of Technology, Tokyo, 152 Japan.

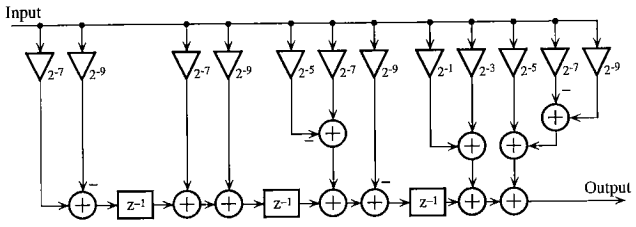


Fig. 1 CSD structure of filter F_1.

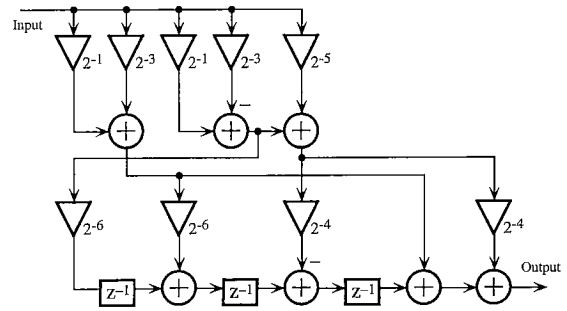
m, the CSD structure has $m-1$ adders and subtractors. In Fig. 1 the number of adders and subtractors is 11. Figure 2 (a) shows the CSDEs of the coefficients of F_1, where $* = -1$. In Fig. 2 (a) the partial expressions $10*, 101$ and $10*01$ exist in more than one place. Note that $10*, 101$ and $10*01$ can be made equivalent in their implementation to $*01, *0*$ and $*010*$, respectively. The shift, add and subtract operations corresponding to those CCSDEs are extracted from all operations in the CSD structure shown in Fig. 1 so that the extracted common operations can be implemented by a single block of adders and subtractors. In this example $10*$ is further extracted from $10*01$. The above processes reduce the number of adders and subtractors. The structure of F_1 realized by this method is shown in Fig. 2 (b), where the number of adders and subtractors is 7. This type of extraction is referred to as the method 1 and the resultant structure is referred to as the structure 1. Note that the CCSDEs to be extracted do not share any nonzero bits. In Fig. 3 the interlaced CCSDEs which do not share any nonzero bits are extracted.

An N-tap linear phase FIR filter has $\lceil \frac{N}{2} \rceil$ independent filter coefficients due to its symmetry. The extraction method is applied to these $\lceil \frac{N}{2} \rceil$ and not all the N coefficients in order not to violate the symmetry. Consider next a 5-tap FIR filter F_2 having CSD coefficients shown in Fig. 4 (a), in which there are 8 adders and subtractors. In this filter F_2 adders and subtractors cannot be reduced by the method 1. In the method 2 therefore the CCSDEs which span several coefficients as shown in Fig. 4 (a) are extracted. Now let h_k, $k = 0, \cdots, N-1$ be all the filter coefficients. Consider the CCSDEs P_n. If some nonzero bits in the CSDEs of the filter coefficients h_i, h_{i+j} and h_{i+p_n} $(0 \leqq \forall j \leqq p_n)$ form a CSDE P_n, additional p_n delay units are required to implement the operations corresponding to P_n. The operations corresponding to P_n $(n = 1, \cdots)$ can be realized using additional $\max_n p_n$ delay units, rather than individually using respective p_n delay units. According to Fig. 4 (a) the structure of F_2 is realized as shown in Fig. 4 (b). The implementation of F_2 has 6 adders and subtractors and an additional delay unit.

Linear phase FIR filters have symmetric coefficients, and they can be efficiently implemented us-

	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}
h_0	1	0	1	0	1	0	*	0	1
h_1	0	0	0	0	*	0	1	0	*
h_2	0	0	0	0	0	0	1	0	1
h_3	0	0	0	0	0	0	1	0	*

(a) The CSDEs of the filter coefficients and the extraction of the CCSDEs.



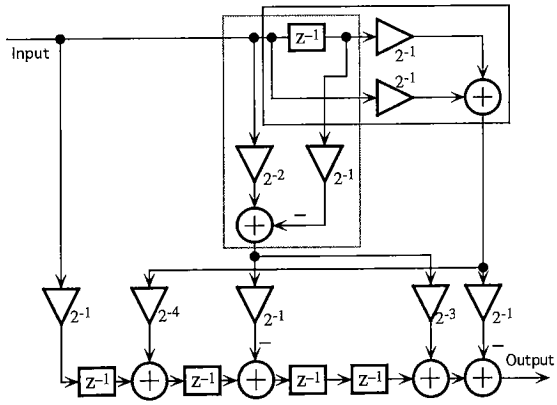
(b) Structure based on the extraction (a).

Fig. 2 Proposed structure of F_1 by method 1.

Fig. 3 The interlaced CCSDEs which do not share nonzero bits one another.

	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}
h_0	0	*	0	0	1
h_1	0	*	0	*	0
h_2	0	0	*	0	0
h_3	0	1	0	0	1
h_4	1	0	0	0	1

(a) The CSDEs of filter coefficients and the extraction of the CCSDEs.



(b) Structure based on the extraction (a).

Fig. 4 Proposed structure of F_2 by method 2.

ing only the independent coefficients. The extraction method 2 destroys the symmetry, and so it is not recommended in general to be applied to linear phase FIR filters. If the number of nonzero bits is limited to 1 in each CSDE, however, a multiplier corresponding to the coefficient is implemented as a wired shifter. Then any adders or subtractors are not reduced even if the symmetry is utilized to the filter structure. If each CSDE has a nonzero bit, there is no CCSDE and the extraction method 1 cannot be applied. In that case the extraction method 2 becomes effective. In this paper the extraction method 2 is applied to linear phase FIR filters whose filter coefficients are expressed as CSDs having only one nonzero bit. Structures thus obtained using the method 2 is referred to as structure 2.

2.2 Proposed Minimization Algorithm

In this section we present algorithms to find all solutions, namely, all the minima in the number of adders and subtractors by extracting CCSDEs from a given set of CSDEs of filter coefficients.

We respectively define $H_0, \cdots, H_{n-1}$ as the CSDEs of the independent filter coefficients $h_0, \cdots, h_{n-1}$. A set H is defined as the set of $H_0, \cdots, H_{n-1}$. If P_i is a CSDE and P_* is the CSDE obtained by multiplying all nonzero bits of P_i by 1 or -1, we denote the relation between P_i and P_* as $P_i = P_*$. The algorithm finds all CCSDEs P_i, $i = 0, \cdots$, which exist in more than one place in $H_0, \cdots, H_{n-1}$. A set Θ is defined as the set of the CCSDEs such that $P_i \in \Theta$ and $P_j \in \Theta$ for $i \neq j$ satisfy $P_i \neq P_j$. Next we describe $P_i > P_j$ if the P_i includes P_j for $i \neq j$. Now Φ_k, $k = 0, \cdots$ is defined as arbitrary sets of CSDEs in Θ. If all CSDEs in a set Φ_i, $i \geqq 0$ are included in P_j and the locations of the CSDEs in P_j do not overlap one another, such Φ_i is written as $\Phi_i(P_j)$, $i = 1, \cdots, c_j$. If all $P_k \in \Phi_i(P_j)$ are included in several $P'_k \in \Phi_{i'}(P_{j'})$ and the locations of $P_k \in \Phi_i(P_j)$ in $P'_k \in \Phi_{i'}(P_{j'})$ do not overlap one another, we describe the relation between $\Phi_i(P_j)$ and $\Phi_{i'}(P_{j'})$ as $\Phi_i(P_j) \leqq \Phi_{i'}(P_{j'})$. If there are one to one relations between k and k' where $P_k = P_{k'}$, $P_k \in \Phi_i(P_j)$ and $P_{k'} \in \Phi_{i'}(P_{j'})$, the relation between $\Phi_i(P_j)$ and $\Phi_{i'}(P_{j'})$ is written as $\Phi_i(P_j) = \Phi_{i'}(P_{j'})$. Next we define the set $\Psi(P_i)$ as the follows:

$$\Psi(P_i) = \{\Phi_j(P_i) \mid \text{There are not any } \Phi_k(P_i) \text{ where } \Phi_j(P_i) < \Phi_k(P_i),\ j \neq k,\ 1 \leqq j, k \leqq c_i\}. \tag{2}$$

Moreover, the following elements $\Phi_j(P_i)$ are removed from the set $\Psi(P_i)$.

$$\Phi_j(P_i) = \Phi_k(P_i),\ j < k. \tag{3}$$

Finally the set $A(P_i)$ is defined as the follows:

$$A(P_i) = \bigcup_j \Phi_j(P_i) \tag{4}$$

In the same way $\Psi(H_i)$ and $A(H_i)$, $i = 0, \cdots, n-1$ are determined also. The algorithm using the extraction method 1 to find all filter realizations which have the minimal number of adders and subtractors is shown as follows:

1. Define X_1^0 as the set which has all H_i as elements. In the following an arbitrary element in X_s^r is described as x_i. Naturally x_i satisfies $x_i = \exists P_j \in \Theta$ or $x_i = \exists H_k \in H$. Let $X_1^1 := X_1^0$, $i := 1$, $i_{\max} := 1$, $m := 1$, $l := 1$.
2. If $X_1^m = \emptyset$, then stop.
3. Remove x_k from X_i^m where $x_k = x_j$, $x_j,\ x_k \in X_i^m$ and $j < k$.
4. If there are not any y and z where $y \in A(x_j)$, $z \in A(x_k)$, $y = z$, $\exists x_j, \exists x_k \in X_i^m$, then go to Step 8.
5. Now determine the set Γ satisfying
$$\Gamma = \{x_j \mid \text{There are not any } x_k \text{ where } x_j,\ x_k \in X_i^m,\ x_j < x_k,\ j \neq k\}. \tag{5}$$
6. Each element in $\Psi(x_j)$ is the set of the CCSDEs which exist in x_j. Let the number of elements in $\Psi(x_j)$ be $\psi(x_j)$. There are $\psi(x_j)$ types of the combinations of the extractions of the CCSDEs from x_j. Let d be the number of the combinations of the extractions of the CCSDEs from all CSDEs in X_i^m. So d can be written as
$$d = \prod_j \psi(x_j),\ x_j \in \Gamma. \tag{6}$$
So Let $X_k^{m+1} := X_i^m$, $k = l, \cdots, l+d$. Replace $\forall x_j \in \Gamma$ by all the elements in $\Phi_a(x_j) \in \Psi(x_j)$ respectively. This replacement means that all elements in the set $\Phi_a(x_j)$ are extracted from x_j. X_l^{m+1} is a set of the CSDE obtained by extracting the CCSDEs from the CSDEs in X_l^m. Let $l := l + d$, $i := i + 1$.
7. if $i > i_{\max}$, then $i := 1$, $i_{\max} := l$, $l := 1$, $m := m+1$ and go to Step 2. Otherwise go to Step 3.
8. The CCSDEs in X_i^m do not have any CCSDEs to be extract. By referring to all sets $X_1^0, X_1^1, \cdots, X_i^m$, which have been used to obtain the set X_i^{m+1}, determine the procedure to extract the CCSDEs, and calculate the number of adders and subtractors. Let $i := i + 1$ and go to Step 7.

An element in $\Psi(P_i)$ is a set of the CCSDEs which can be extracted from P_i all together. All the optimal structures can be obtained by examining all the combinations of only the extractions in Ψ. Due to using such the set Ψ, the computational time can be reduced.

Since the data structure of X_j^i, $i = 0, \cdots$, $j = 1, \cdots$ is a tree structure, the above algorithm can recursively search X_j^i. Then the memory capacity required for the above algorithm can be decreased by the recursive operation.

To implement the shift, add and subtract operations corresponding to the CCSDEs in the extraction method 2, additional delay units are required. In this paper the ratio of the hardware costs of an adder and a delay unit is specified as $1 : w$. Then the hardware cost S of a whole filter is given as

$$S = (\text{The number of adders and subtractors}) + (\text{The number of additional delay units}) \times w. \quad (7)$$

The above cost factor S is minimized. In such a case the computational time for the minimization becomes enormous. So by using the following simple algorithm, which is referred to as the method 2, a local solution is found. Suppose each filter coefficient has only one nonzero bit. H_0 is defined as all CSDEs of filter coefficients. P_i is defined as the CCSDEs. H_0 and P_i have two dimensional CSDEs. D_k, $k = 1, \cdots, d_{max}$ are defined as the sets of P_i corresponding to the shift, add and subtract operations which require k delay units to be realized.

1. Let $k := 1$, and let H_0 be the initial element in the set X.
2. If $k > d_{\max}$, then Stop. Otherwise determine the sets B_j, $j = 2, \cdots, b_{\max}$ of $P_i \in D_k$ which have j nonzero bits. Let $j := b_{\max}$.
3. Find all the locations of each CSDE $P_i \in B_j$ in all CSDEs $x \in X$. Extract $P_i \in B_j$ from all $x \in X$ in descending order of the number of its locations. Then replace the nonzero bits where $P_i \in B_j$ exists, by zero bit successively. Let $P_i \in B_j$ which can be extracted in more than one place in $\forall x \in X$ be one of elements in X.
4. Let $j := j - 1$. If $j < 2$, then let $k := k + 1$ and go to Step 2. Otherwise go to Step 3.

The above algorithm determines the optimized filter structures which have k, $k = 1, \cdots, d_{\max}$ additional delay units, respectively. Among these structures, the structure having the minimum S is selected as the solution, which is referred to as the structure 2.

2.3 Critical Path Analysis

The algorithms in the above section determine the CCSDEs to be extracted in the CSDEs of the filter coefficients as shown in Figs. 2(a) and 4(a). Those extractions of the CCSDEs enable the filter structure to have the minimal adders and subtractors. The order of additions and subtractions corresponding to the CCSDEs is, however, undetermined yet. In this section we explain a method to order those additions and subtractions so that the critical path may become the shortest. The method gives practical filter structures such as Figs. 2(b) and 4(b) from the extractions of CCSDEs such as Figs. 2(a) and 4(a). Each coefficient of N-tap FIR filter is referred to h_i, $i = 0, \cdots, N-1$, whose CSDE has $\mu(i)$ nonzero bits, and the number of adders and subtractors on the critical path is defined as c. In the CSD structure if the shift, add and subtract operations corresponding to filter coefficients are implemented as the tree structures with the minimum depths and each output of tap delay is added to each root of that tree structure, c is minimum as follows:

$$c = \max_{0 \leqq i \leqq N-2} \{\lceil \log_2 \mu(N-1) \rceil, \lceil \log_2 \mu(i) \rceil + 1\}. \quad (8)$$

Note that the root of the tree structure corresponding to h_{N-1} is not added to any outputs of the tap delays. In the structures 1 and 2 no addition is also given to the output of the tree structure corresponding to h_{N-1}. In the structures 1 and 2 minimum height trees corresponding to filter coefficients can not be implemented as the CSD structures generally, because the critical path must be optimized under the restrictions that the operations corresponding to CCSDEs are extracted.

Firstly the partial tree structures corresponding to the CCSDEs are implemented. Each CSDE corresponding to the filter coefficient is implemented by interlacing those partial tree structures so that the critical path may become the shortest.

Now consider a CSDE P_0 which consists of m extracted CCSDEs $P_1, \cdots, P_m$ and another p nonzero bits, and assume that the partial trees corresponding to P_1, $\cdots$, P_m are implemented as the minimum height trees, respectively. Those heights are defined as $c_1, \cdots, c_m$. A minimum height tree corresponding to p nonzero bits are also implemented. Then that height is defined as c_{m+1}, and c_{m+1} can be written as

$$c_{m+1} = \lceil \log_2 p \rceil. \quad (9)$$

By implementing nonzero bits not extracted in P_0 as an above minimum height tree, P_0 can be regarded as $m+1$ CCSDEs. Accordingly the partial tree T_0 corresponding to P_0 can be implemented by connecting the partial trees corresponding to $m+1$ CCSDEs.

Now we explain how these $m+1$ partial trees are connected. It is assumed that the heights of $m+1$ partial trees are minimum, and those trees corresponding to P_i, $i = 1, \cdots, m+1$ are written as T_i, $i = 1, \cdots, m+1$. The set of the elements T_i, $i = 1, \cdots, m+1$ is defined as Ω. The minimum height tree T_0 can be created by using the following algorithm.

1. Let T_0 be $\emptyset$ and T_0 be an element in Ω.

2. If there is no other element but T_0 in Ω, then stop.
3. Select a tree whose height is minimum in Ω, and let that tree be T^1. Remove T^1 from Ω.
4. Select a tree whose height is minimum in Ω, and let that tree be T^2. Remove T^2 from Ω.
5. By connecting both roots of T^1 and T^2, create a new tree. Then let that created tree be T_0.
6. Go to Step 2.

The tree T_0 created in the above algorithm is the minimum height tree corresponding to P_0. Then the minimum height of T_0 can be written as

$$c = \left\lceil \log_2 \left(\sum_{i=1}^{m+1} 2^{c_i} \right) \right\rceil . \tag{10}$$

In the above algorithm the minimum height tree T_0 can be obtained under the condition that the heights of $m+1$ partial trees are minimum. The extracted CCSDEs form a layer structure in a CSDE of a filter coefficient as shown in Fig. 5. In the order of depth of locations of CCSDEs, minimum height trees corresponding to the CCSDEs are implemented. Then the minimum height trees and their heights corresponding to CSDEs of filter coefficients are determined. After implementing the operations corresponding to the filter coefficients, the operations corresponding to the same CCSDEs are shared. Those common uses of the operations do not cause the critical path to be longer.

Linear phase FIR digital filters have symmetric coefficients. In the structures obtained by using the extraction method 1 outputs of minimum height tree structures corresponding to filter coefficients are added to outputs of tap delays. Figure 6(a) shows a tap structures obtained by using the extraction method 1. In the structure 2 each CSDE corresponding to filter coefficient has only one nonzero bit. The symmetry of filter coefficients is not utilized. In other words, the products with the respective coefficients need not explicitly be computed. Thus as Fig. 6(b) shows, the outputs of tap delays are also regarded as leaves of tree structures and minimum height trees are determined. The heights of such trees become smaller.

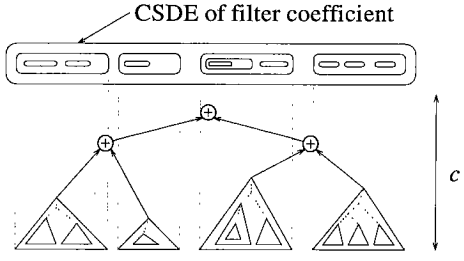
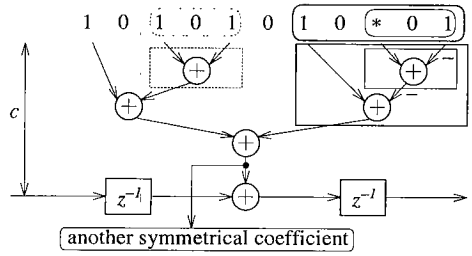
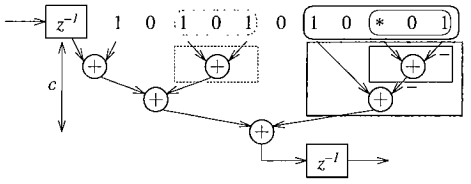


Fig. 5 CCSDEs in a CSDE of a filter coefficients and those tree structures.



(a) Method 1.

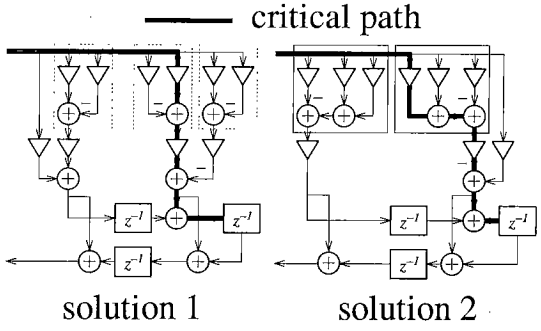


(b) Method 2.

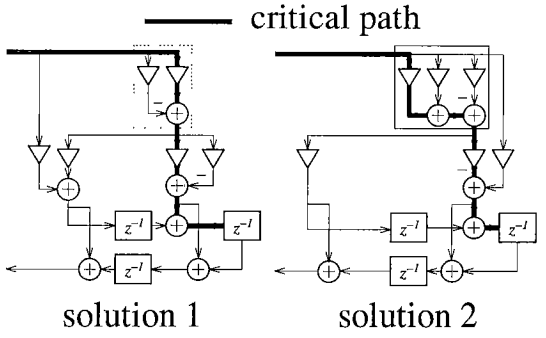
Fig. 6 Tap structures by the method 1 and method 2.

	solution 1		solution 2
h_0	0 0 1 0 1 0 *	h_0	0 0 1 0 1 0 *
h_1	1 0 * 0 * 0 1	h_1	1 0 * 0 * 0 1

(a) Two ways of the extractions giving the minimal number of adders and subtractors.



(b) The structures obtained by implementing the operations corresponding to the filter coefficients as the minimum height trees.



(c) After sharing the operations corresponding to the CCSDEs.

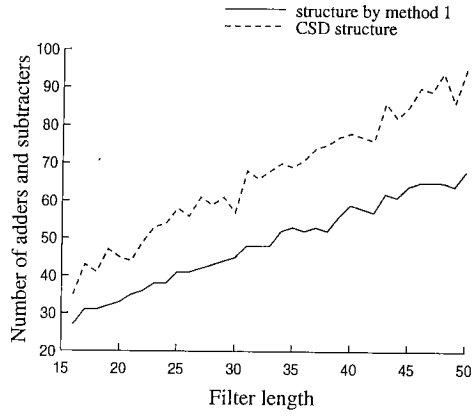
Fig. 7 The implementation of the structure 1 having the best critical path.

Figures 7(a), (b) and (c) show examples of the optimization of critical paths. Consider a 4-tap linear phase FIR filter whose coefficients are h_i $(i = 0, \cdots, 3)$ where $h_0 = h_3$ and $h_1 = h_2$. Each CSDE of the independent coefficient is shown in Fig. 7 (a). Then three types of the extractions can be obtained by using the algorithm of the extraction method 1. Only two types of the extractions are shown in Fig. 7 (a). From each solution the operations corresponding to all the CCSDEs and CSDEs of the coefficients are implemented as the minimum height trees according to the above algorithm as shown in Fig. 7 (b). Then sharing the operations corresponding to the same CCSDEs leads to the structures shown in Fig. 7 (c). Figures 7 (b) and 7 (c) also show that such a sharing does not affect the critical path. From Fig. 7 (c), the critical path obtained by solution 1 is shorter than that obtained by solution 2. Then the extractions illustrated as solution 1 are selected. In these examples the critical path in the CSD structure are equal to that from the solution 1. But in general the critical path in the structure 1 or 2 is longer than that in the CSD structure.

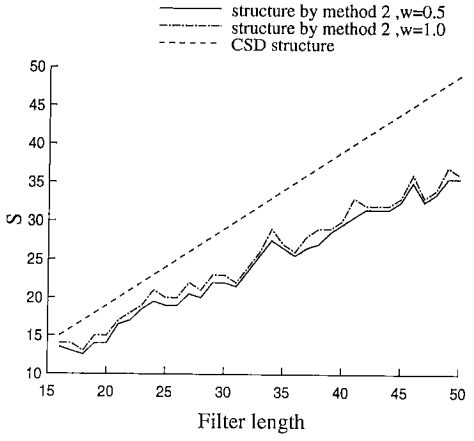
2.4 Comparison between the Proposed Structures and the CSD Structures

In this section, the numbers of adders and subtractors and the critical paths in the proposed structures are compared with those in the conventional CSD structures. In the case of the structure 2, the cost factor S defined in (7) is compared with that in the CSD structure. The ratio w is assumed to be 0.5 or 1.0. The specifications of filters are: filter length $16 - 50$, passband $0.0 - 0.2$, stopband $0.25 - 0.5$, passband and stopband weights $1 : 1$. The equiripple linear phase FIR filters satisfying these specifications are designed using the Remez exchange algorithm. Then all the filter coefficients are rounded off to CSDs of wordlength 12 or 16 for the methods 1 and 2, respectively. In the latter case the number of nonzero bits is limited to one in each coefficient. By using the method proposed in [10], it is possible to round off the coefficients to the CSDs having only one nonzero bit. Figures 8 (a) and (b) show the results of the structure 1 and the structure 2 together with the conventional CSD structures, respectively. From these figures, the structures 1 and 2 have far less adders and subtractors than the CSD structures. Especially the 19-tap FIR filter structure by the method 1 has only 68% adders and subtractors of CSD structure.

The numbers of adders and subtractors on the critical paths in the structure 1 happen to be equal to those in the CSD structures, and become 4 in all examples. Accordingly the operational speeds of the structure 1 and CSD structures are equal in all the given specifications. So the operational speeds of both structures are guessed to be equal in all practical specification. Figure 9 shows the numbers of adders and subtractors on the critical paths in the structure 2 and in the CSD structures. Because all the CSDEs of coefficients have only one nonzero bit, only one adder or subtractor is on the critical path in the CSD structures. Hence more adders



(a) Method 1.



(b) Method 2.

Fig. 8 The number of adders and subtractors in structures by the method 1, method 2 and CSD structures. S=(the number of adders and subtractors)+(the number of another delay units)$\times w$.

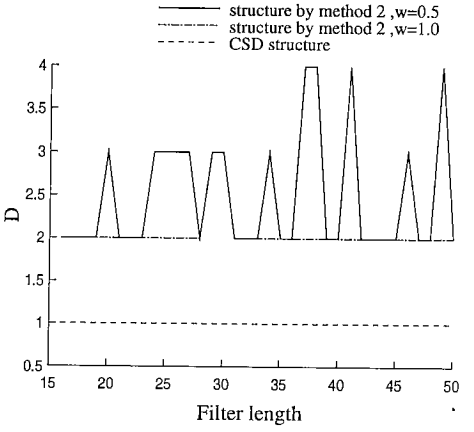


Fig. 9 The number of adders and subtractors on the critical path in structures by method 2 and CSD structures. D=the number of adders and subtractors on the critical path.

and subtractors are obviously on the critical path in the structure 2.

3. MILP-Based Filter Design for the Proposed Structure

3.1 CSD Coefficient Space

In this section the CSDE is modified as [9], because the representable range of the normal CSDs is narrower than that of the binary representations. Then the CSDs proposed in [9], whose representable range is widened, are used in designing filters.

Many algorithms to design linear phase FIR filters are proposed in [5]–[9]. The MILP method [6] is often used to design FIR filters whose coefficients are CSDs, because it can obtain the filter having the optimum frequency response under the condition that the maximum number of nonzero bits in the CSDE of each filter coefficient is limited. Here let that maximum number be 2 and the coefficient wordlength be 9. Each filter coefficient may have at most two adders or subtractors. The number of all CSDs satisfying those conditions is shown in Table 1. In the conventional MILP method the coefficient space is spanned by those CSD numbers satisfying the condition. Next consider several CCSDEs are extracted from CSDEs of filter coefficients. Let the CCSDEs be two CSDEs, 101 and $10*$ for example. A 9 bit CSDE $P_1 = 10*00*0*0$ has 4 nonzero bits. Since P_1 does not satisfy the above condition, it is excluded in the conventional coefficient space. P_1 includes the CCSDEs 101 and $10*$, which can be extracted as follows:

$$P_1 : (10*)00(*0*)0. \tag{11}$$

The operations corresponding to $10*$ and 101 can be implemented using one subtractor and one adder, respectively, and additional adder and subtractor are needed to connect both the outputs of those operations with the input of a tap delay. The operations corresponding to $10*$ and 101 can be commonly used to implement other filter coefficients. So the operations corresponding to $10*$ and 101 are excluded from P_1. Then the operations corresponding to P_1 have only one adder and one subtractor as shown in Fig. 10, and satisfies the condition that the maximum number of adders and subtractors is 2. Those adder and subtractor are the remaining operations after extraction. In this paper M^*, the maximum number of adders and subtractors in the operations remaining after extracting the CCSDEs from the CSDEs of filter coefficients is specified.

Now consider a 9 bit CSDE $P_2 = 11000*0*0$, which is included in the widened range as proposed in [9]. The CSDE P_2 is not a normal CSD but one of the CSDs proposed in [9]. Since the numbers $11_{(2)}$ and $10*_{(2)}$ are equal, the CCSDEs 11 and $*0*$ can be extracted from P_2 as the follows:

Table 1 The number of expressions in various discrete spaces. The wordlength is 9. The CSDEs in the conventional space have at most 2 nonzero bits. The operations corresponding to the CSDEs in the proposed space have at most 2 adders and subtractors in the operations remained by extracting the CCSDEs 101 and $10*$ from the CSDEs.

binary	1023
conventional space	133
proposed space	493

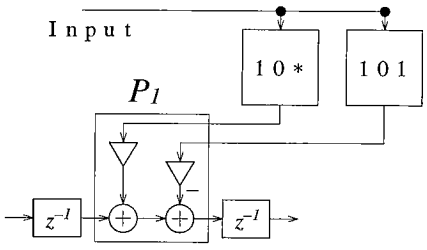


Fig. 10 The adder and subtractor in the operations corresponding to P_1.

$$P_2 : (11)000(*0*)0. \tag{12}$$

Although the CSDE P_2 has 4 nonzero bits, P_2 is now included in the proposed CSD coefficient space. Note that the length of the expression 11 is different from that of $10*$. If the CSDE 11 is replaced by the CSDE $10*$ in P_2, the length of P_2 become longer and the longer wordlength may not satisfy the specification. Then all CSDEs $10*$ and $(*01)$ in the CSDEs of coefficients are replaced by the expressions 11 and $(**)$, respectively. The number of CSDs included in the proposed CSD coefficient space is shown in Table 1. The table shows that the coefficient space to be searched by MILP is extended. In spite of limiting the number of nonzero bits by $M^* = 2$, the proposed space can cover about 50% of all the representable expressions. Accordingly this wider coefficient space helps the MILP to find a better solution. If the coefficient wordlength is l, the number of all CSDEs is 2^l. The given CCSDEs are extracted from the 2^l CSDEs, and the CSDEs satisfying the specification are chosen as the CSDEs in the coefficient space.

The branch and bound algorithm to solve MILP problems is useful for its flexibility. It can deal with any discrete space. Here the branch and bound algorithm is used to solve the MILP problems over the discrete space which spans the candidates of filter coefficients determined by extracting given CCSDEs. Lim's papers [6] mentions applications of the branch and bound method to filter design in detail.

In the Sect. 2 the two algorithms are shown. By using those algorithms the number of adders and subtractors used in every filter can be minimized. So by searching all the filters which have CSD coefficients with those algorithms, the filter whose frequency response is the best would be obtained instead of limiting the CCSDEs to be extracted. But the number of such filters is 2^{lN}

where l is the wordlength and N is the number of the independent coefficients. Then it is practically impossible to search all the filters due to enormous computing time, and the proposed design method is effective on that point.

3.2 Comparison between the Proposed Method and the Conventional Method to Design Filter

The MILP in the proposed method searches over the wider coefficient space than in the conventional method. Now many specifications are designed and the Chebyshev errors of the resultant frequency responses are compared. The specifications in frequency responses are: passband $0.0-0.15$, stopband $0.25-0.5$, wordlength 9, filter length $25-37$. Passband and stopband weight is $1:1$.

Since the shortest CSDEs are $10*$ and 101 in all possible CSDEs, they are expected to appear in CSD filter coefficients with the highest probability. So $10*$ and 101 are extracted as the CCSDEs here. M^* is given as 2.

Firstly filters are designed to meet those specifications using the proposed method. The number of adders and subtractors in the operations of the independent coefficients is calculated as M. So the total number of nonzero bits in the CSDEs of the filter coefficients is given as $M+1$. For the reference the conventional CSD structures which have $M+1$ nonzero bits are designed for the same specifications so that the total numbers of adders and subtractors are equal for the proposed and the conventional structures. The design method in [9] is used in this case because that method surpasses others including MILP which search over the conventional CSD coefficient space.

Figure 11 shows the results of the proposed method together with those obtained by the conventional method [9]. In the design example for filter length 34, the optimized frequency responses using the proposed and conventional method are shown in Figs. 12 (a) and (b). From these figures the frequency responses designed by the proposed method are far better than those designed by the conventional method [9]. Figures 13 (a) and (b) show the structures of the 34 tap filters designed by using the proposed and conventional method. In these figures, the numbers of adders and subtractors in the operations of the independent coefficients are 25 in both the structures.

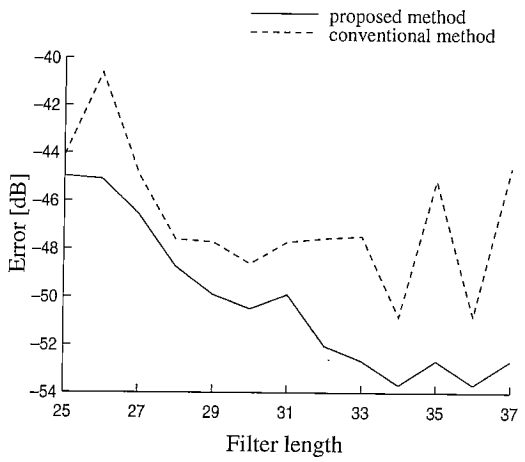
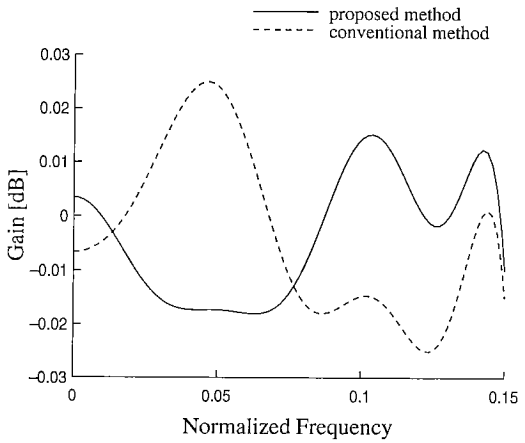
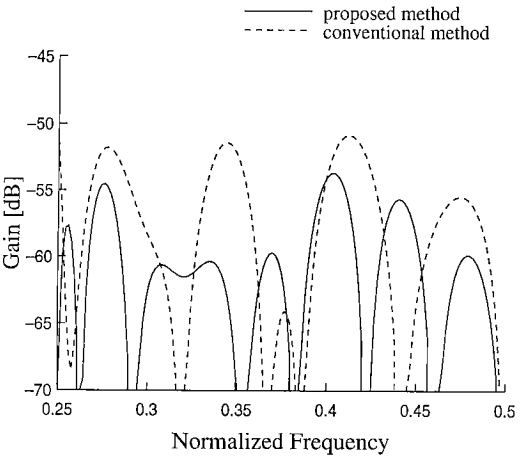


Fig. 11 The errors of frequency responses.



(a) Frequency responses in the passband.

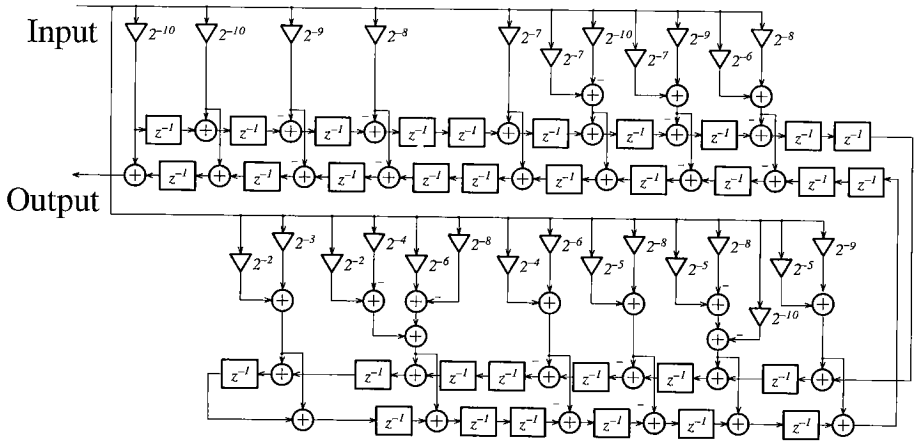


(b) Frequency responses in the stopband.

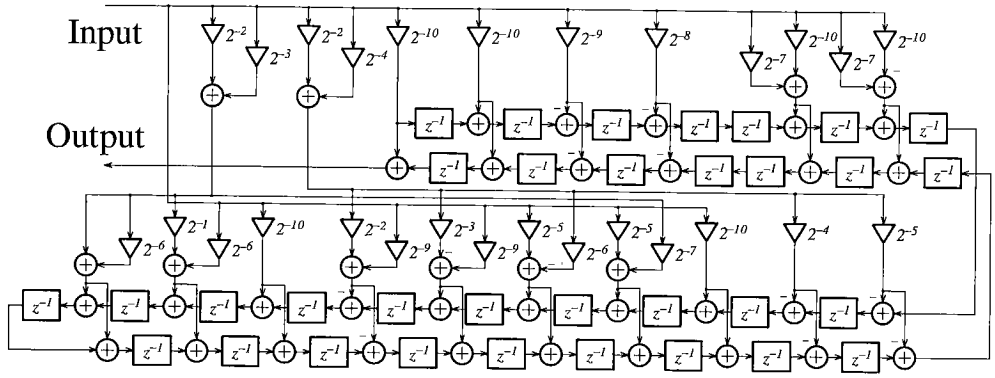
Fig. 12 The designed frequency responses for 34-tap FIR filters.

4. Conclusion

This paper proposes fast FIR digital filter structures using minimal number of adders. Hartley's technique, which reduces the number of adders and subtractors in CSD filter structures, is modified to obtain all filters whose operations have minimal number of adders and subtractors. Among them a filter with the shortest critical path is selected. The critical paths obtained in this way happened to be equal to those of conventional CSD structures in all examples we tried. This paper also presents the method to design FIR filters to



(a) The structure obtained by the conventional method.



(b) The structure obtained by the proposed method.

Fig. 13 The obtained filter structures for 34-tap FIR filter.

which Hartley's technique are applied. The application of the technique extends the CSD coefficient space to be searched by the MILP. The CSD coefficient space is now determined by the maximum number of adders and subtractors used in the filter structures and not the number of nonzero digits. For the same numbers of adders and subtractors, the proposed method, which searches the wider coefficient space, can yield fairly superior frequency responses than the conventional method.

References

[1] R. Jain, P.T. Yang, and T. Yoshino, "FIRGEN: A computer-aided design system for high performance FIR filter integrated circuits," IEEE Trans. Signal Process., vol.39, no.7, pp.1655–1668, July 1991.

[2] D. Li, "Minimum number of adders for implementing a multiplier and its application to the design of multiplier-less digital filters," IEEE Trans. Circuits & Syst., vol.42, no.7, pp.453–460, July 1995.

[3] R. Hartley, "Optimization of canonic signed digit multipliers for filter design," Proc. IEEE Int. Symp. Circuits & Syst., pp.1992–1995, June 1991.

[4] A.G. Dempster and M.D. Macleod, "Use of minimum-adder multiplier blocks in FIR digital filters," IEEE Trans. Circuits & Syst., vol.42, no.9, pp.569–577, Sept. 1995.

[5] H. Samueli, "An improved search algorithm for the design of multiplierless FIR filters with powers-of-two coefficients," IEEE Trans. Circuits & Syst., vol.36, no.7, pp.1044–1047, July 1989.

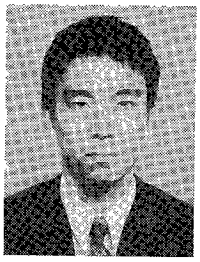
[6] Y.C. Lim, "Design of discrete-coefficient-value linear phase FIR filters with optimum normalized peak ripple magnitude," IEEE Trans. Circuits & Syst., vol.37, no.12, pp.1480–1486, Dec. 1990.

[7] N. Benvenuto, M. Marchesi, and A. Uncini, "Applications of simulated annealing for the design of special digital filters," IEEE Trans. Signal Process., vol.40, no.2, pp.323–332, Feb. 1992.

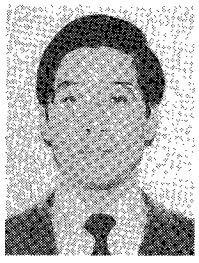
[8] M. Yagyu, T. Yoshida, A. Nishihara, and N. Fujii, "Design of FIR digital filters with minimum weight representation," Proc. IEEE Int. Symp. Circuits & Syst., pp.227–230, May 1995.

[9] M. Yagyu, A. Nishihara, and N. Fujii, "Design of FIR digital filters using estimates of error function over CSD coefficient space," IEICE Trans. Fundamentals, vol.E79-A, no.3, pp.283–290, March 1996.

[10] Y.C. Lim, J.B. Evans, and B. Liu, "Decomposition of binary integers into signed power-of-two terms," IEEE Trans. Circuits & Syst., vol.38, no.6, pp.667–672, June 1991.



Mitsuhiko Yagyu was born in Osaka, Japan, on December 3, 1969. He received the B.S. and M.E. degrees from Tokyo Institute of Technology, Tokyo, Japan, in 1993 and 1995 respectively. He is currently working toward the Ph.D. degree in the Graduate School of Science and Engineering, Tokyo Institute of Technology. His main research interest is in digital filter design.



Akinori Nishihara was born in Fukuoka, Japan, on February 26, 1951. He received the B.E., M.E. and Dr.Eng. degrees in electronics from Tokyo Institute of Technology in 1973, 1975 and 1978, respectively. Since 1978 he has been with Tokyo Institute of Technology, where he is now Associate Professor of Department of Physical Electronics, Faculty of Engineering. His main research interests are in filter design, 1D and multi-D signal processing, and modern applications of classical circuit theory. From 1990 to 1994 he served as an Associate Editor of the IEICE Trans. Fundamentals, and is now serving as an Associate Editor of the IEEE Trans. Circuits & Systems II. He is now Student Activities Committee Chair, IEEE Region 10 (Asia Pacific Region). Dr. Nishihara is a member of IEEE, EURASIP, ECS and JET.

Nobuo Fujii received B.E. degree from Keio University, Yokohama, Japan, and M.E. and Doctor of Engineering degrees from Tokyo Institute of Technology, Tokyo, Japan, in 1966, 1968, and 1971, respectively. Since 1971, he has been with the Faculty of Engineering, Tokyo Institute of Technology where he is now a professor in the Department of Physical Electronics. From 1984 to 1985, he was a visiting scholar at the University of California, Santa Barbara. From 1990 to 1992, he served as an editor of the Transaction of the Institute of Electronics, Information, and Communication Engineers and is now one of the chief editors of the International Journal of Analog Integrated Circuits and Signal Processing, Kluwer Academic Publishers. He is the chairman of the technical group of electronic circuits of IEE Japan and the chairman of the Circuits and Systems Society of IEEE Tokyo Chapter. His main interest lies in the fields of active networks, analog integrated circuits, and analog signal processing. He is the recipient of the Best Paper Award of the Institute of Electrical and Communication Engineers of Japan. He is the author of more than 10 books. Dr. Fujii is a member of the Institute of Electrical and Electronics Engineers, the Institute of Electrical Engineers of Japan.