

Subexpression encoded extrapolated impulse response FIR filter with perfect residual compensation

Yu, Ya Jun; Shi, Dong; Lim, Yong Ching

2008

Yu, Y. J., Shi, D., & Lim, Y. C. (2008). Subexpression encoded extrapolated impulse response FIR filter with perfect residual compensation. Proceedings of 2008 IEEE International Symposium on Circuits and Systems (pp.2446-2449) Seattle,WA,USA.

<https://hdl.handle.net/10356/90681>

<https://doi.org/10.1109/ISCAS.2008.4541950>

© 2008 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE. This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder. <http://www.ieee.org/portal/site> This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder.

Subexpression Encoded Extrapolated Impulse Response FIR Filter with Perfect Residual Compensation

Ya Jun Yu, Dong Shi *and* Yong Ching Lim

School of Electrical and Electronic Engineering,

Nanyang Technological University, Republic of Singapore 639798

Email: eleyuyj@pmail.ntu.edu.sg, shid0001@ntu.edu.sg, *and* elelimyc@pmail.ntu.edu.sg

Abstract—Earlier research has shown that filter impulse responses are quasi-periodic and filter coefficients can be approximated by extrapolation techniques. In this paper, a discrete coefficient extrapolated impulse response filter is proposed. Residual compensation technique is introduced to perfectly realize the filter coefficients. Both the extrapolated filter coefficients and the residuals are subexpression encoded to minimize the filter complexity. Numerical example shows that the proposed technique achieves minimum number of adders in the synthesis of the filter when compared with any other existing techniques.

I. INTRODUCTION

Multiplierless finite impulse response (FIR) filters are very attractive in VLSI (Very Large Scale Integration) implementation since the multiplication of the input sample data with the filter coefficients can be achieved by a limited number of additions and shifts. Since the shifts for fixed coefficient values can be realized by hardware, for decades, the efforts to reduce the filter complexity have been focused on reducing the number of adders. One of the most efficient approaches to reducing the number of adders in realizing the coefficient multiplication is the subexpression sharing technique.

The subexpression sharing technique can be grouped into three categories. The first category is *common subexpression sharing* [1]–[3] in which the common patterns of the coefficients, usually represented in a canonic signed digital (CSD) form, or binary numbers are found and shared. The second category of algorithms are based on the *adder graph* [4], [5] in which larger coefficients are realized by shifting and adding those already realized smaller coefficients. The third category is the *difference method* [6], [7] of which the cost to realize the differences of the coefficients are minimized to reduce the overall complexity.

Since the subexpression coefficient implementation in transposed form FIR filter is explicitly a multiple constant multiplication (MCM) problem, most of the above algorithms were considered in this platform. As a dual of the transposed form, the direct form FIR filter can always be implemented using the subexpression sharing with exactly the same complexity. However, in transposed form, the adders can be easily classified into the adders for the multiplier block (MB) and the adders in between the delay elements of the delay line, as shown in Fig. 1; these two types of adders are called MB adder and structural adder, respectively. In this paper, the transposed form is considered for easily referring to the different type of adders.

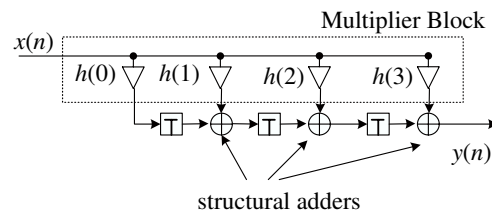


Fig. 1. Transposed form of FIR filter

When considering to implement a set of coefficient values using subexpression sharing, these coefficients are first transformed to positive odd numbers by scaling the coefficient values with a proper signed power-of-two factor. Among these positive odd numbers, only the non-one distinct coefficients are considered, since coefficient with value one and duplicated coefficients can be realized without any cost. Thus, for a filter with L distinct positive non-one odd-valued coefficients, it has been shown in [8] that the lower bound of the number of MB adders of the transposed direct form implementation is equal to the minimum number of adders required to realize the simplest coefficient plus $(L - 1)$. In most cases, the number of adders required to realize the simplest coefficient is one, since there are always very small magnitude coefficients which can be realized by one adder in a practical filter. Therefore, the lower bound basically is determined by the number of distinct coefficients, L . For many benchmark filters, this lower bound has been achieved, for instance, example 1 of reference [9]. The example was a 120th-order filter with 52 distinct coefficients. Many algorithms [5], [10]–[12] have achieved this lower bound. Therefore, to further reduce the number of MB adders, the only possibility is to reduce the lower bound by reducing the number of distinct coefficients; this can be achieved by reducing the dynamic range of the coefficient multiplier.

In this paper, a discrete coefficient extrapolated impulse response FIR filter is proposed. Due to the quasi-periodic property of the impulse response [13], [14], coefficients can be approximated by extrapolation. Residual compensation is introduced to perfectly restore the coefficient values to the original optimum values. By using the proposed technique, the dynamic range of the coefficient multipliers are significantly reduced. The reduction in dynamic range increases the chance of having coefficients with identical values. This results in

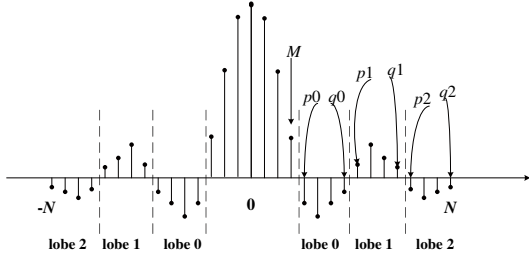


Fig. 2. A typical impulse response sequence of an FIR filter

a reduction in the lower bound of the number of MB adders. Design examples show that the proposed technique reduces the filter complexity in terms of the number of adders significantly compared with any other existing techniques.

II. THE EXTRAPOLATED IMPULSE RESPONSE

This section briefly reviews the concept of extrapolated impulse response technique. A typical impulse response of an FIR filter is quasi-periodic as shown in Fig. 2. Most of the energy of the impulse response is concentrated at the center lobe while the side lobes have decreasing magnitudes. If lobe 0, lobe 1, lobe 2, etc., (see Fig. 2) have the same number of samples, they can be approximated as a scaled version of anyone of the lobes. For instance, lobes 0 and 1 can be approximated as scaled versions of a prototype lobe, say lobe 2. Mathematically, for the i -th lobe, we have:

$$h(n) \approx \alpha_i h(n + k_i), \text{ for } n = p_i, \dots, q_i, \quad (1)$$

where α_i is the scaling factor, k_i is the displacement of the i -th lobe to the prototype lobe; p_i and q_i are the first and the last index of the i -th lobe.

If we assume that the center of the impulse response is at $n = 0$, we have the z -transform transfer function:

$$H(z) = h(0) + \sum_{n=1}^N h(n)(z^n + z^{-n}). \quad (2)$$

Assuming that the lobes begin at $n = p_0$ through $n = q_0$ and $n = p_1$ through $n = q_1, \dots$, $H(z)$ can be rewritten as

$$H(z) = h(0) + \sum_{n=1}^M h(n)(z^n + z^{-n}) + \sum_{n=p_0}^{q_0} h(n)(z^n + z^{-n}) + \sum_{n=p_1}^{q_1} h(n)(z^n + z^{-n}) + \sum_{n=p_2}^{q_2} h(n)(z^n + z^{-n}) + \dots \quad (3)$$

The durations of these lobes are $q_0 - p_0 + 1, q_1 - p_1 + 1, q_2 - p_2 + 1, \dots$ and they might not be all equal. However, they can be separated into several groups, and each group consists of lobes with the same duration. Thus, in each group, any lobe can always be approximated as the scaled version of a prototype lobe in that group. For expository convenience, we assume all the lobes have the same duration d since other cases are just simple extensions. As a result, (3) can be written as:

$$H(z) = h(0) + \sum_{n=1}^M h(n)(z^n + z^{-n}) +$$

$$\sum_{r=0}^{R-1} \sum_{m=1}^d h(M + m + rd)(z^{M+m+rd} + z^{-(M+m+rd)}), \quad (4)$$

where R is the number of lobes, d is the duration of each lobe. If the lobe with the smallest magnitude (usually, the $(R-1)$ -th lobe)¹ is chosen as the prototype lobe, $H(z)$ can be approximated by

$$H(z) \approx \hat{H}(z) = h(0) + \sum_{n=1}^M (z^n + z^{-n}) +$$

$$\sum_{m=1}^d h(M + m + (R-1)d) \sum_{r=0}^{R-1} \alpha_r (z^{M+m+rd} + z^{-(M+m+rd)}), \quad (5)$$

where α_r is the r -th scaling factor and $\alpha_{R-1} = 1$.

A realization of a 12th-order symmetrical filter using the above extrapolation technique is given in Fig. 3(a), where the coefficients $h(3)$ and $h(4)$ are implemented as scaled version of $h(5)$ and $h(6)$ with a scaling factor α_0 .

Optimization techniques have been proposed in [13], [14] to optimize the filter coefficients of the prototype lobe as well as the scaling factors. Although, the filter complexity has been reduced, the frequency response of the extrapolated filter is generally degraded compared with that of the minimax optimum. To meet a given specification, the order of the extrapolated filter is, in general, longer than that of the minimax optimum.

III. DISCRETE COEFFICIENT EXTRAPOLATED FILTER WITH PERFECT RESIDUAL COMPENSATION

In the traditional extrapolated filters, the frequency response is degraded due to the fact that the complexity reduction is achieved by reducing the degree of freedom of filter coefficients. Coefficients are only approximations of their optimum values. In this section, a perfect residual compensation technique is proposed to restore the optimum coefficient values when multiplierless filters are implemented.

Assume that the discrete space optimum impulse response of a $2N$ -th order linear phase FIR filter for a given specification is $h(n)$. The discrete space, \mathcal{D} , may be the finite-word length space or the signed power-of-two space. Therefore, $h(n) \in \mathcal{D}$, and $h(-n) = h(n)$, for $-N \leq n \leq N$.

Since the coefficients are symmetric, only the coefficients with non-negative index are considered. Assume further that the coefficients are quasi-periodic with duration d from $n = M + 1$ to $n = M + Rd$ for R periods. Thus, $0 \leq M < N$, $R \leq \frac{N-M}{d}$ and both are integers. Following the procedure of the traditional extrapolated filters, the filter coefficients of $h(n)$ for $0 \leq n \leq M$ and $M + Rd + 1 \leq n \leq N$ are implemented accurately. For $h(n)$ within the range $M + 1 \leq n \leq M + Rd$, if the period with the smallest magnitude is chosen as the prototype lobe, and all the other periods are approximated as the scaled versions of the prototype lobe, the

¹lobe 0 is considered as the zero-th lobe

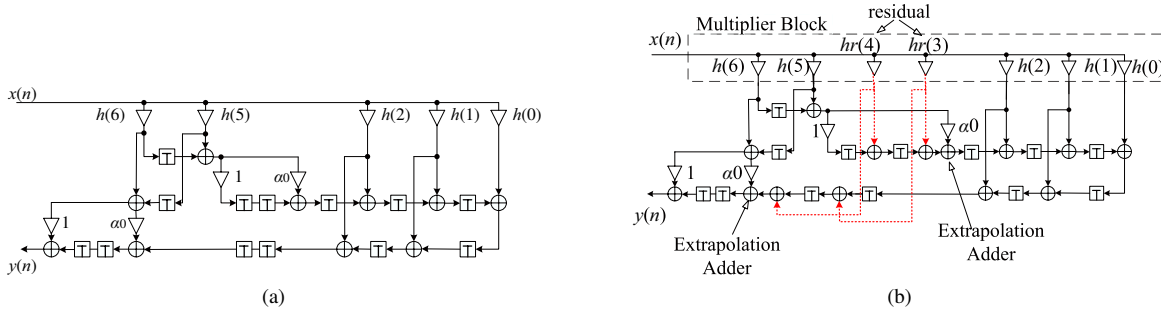


Fig. 3. (a) Traditional Extrapolation Structure (b) Extrapolation Structure with Residual Compensation

approximated coefficients, denoted as $h_a(n)$, from $n = M + 1$ to $n = M + (R - 1)d$ become

$$\begin{aligned} h_a(n - (R - r - 1)d) &= \alpha_r h(n) \\ \text{for } M + (R - 1)d + 1 \leq n \leq M + Rd \text{ and} \\ 0 \leq r \leq R - 2, \end{aligned} \quad (6)$$

where the scaling factors, α_r , are integers or power-of-two numbers for multiplierless implementation. Therefore, the coefficient residuals, denoted as $h_r(n)$, due to the approximation are given by

$$h_r(n) = h(n) - h_a(n), \text{ for } M + 1 \leq n \leq M + (R - 1)d. \quad (7)$$

In the implementation of the extrapolated filter, if the coefficient residuals are compensated by adding the products of the signal samples and the residuals into the tap delay line, as shown in Fig. 3(b), the filter impulse response is restored perfectly.

Comparing the residual compensated extrapolated filter implementation shown in Fig. 3(b) with the non-extrapolated transposed direct form implementation shown in Fig. 1, it can be seen that the numbers of structural adders are the same for both structures, since generally each extrapolated coefficient is compensated by a corresponding residual. The MB adders of the proposed technique can be further classified into the prototype coefficient adders and the residual adders, which are used to realize the prototype coefficients and residuals, respectively. Both the prototype coefficients and residuals can share the same MB in subexpression sharing, since they are multiplied with the same signal, as shown in Fig. 3(b). Since the residual is the difference of the optimum coefficient value and the extrapolated approximation, its magnitude is much smaller than the original optimum value. The magnitudes of the prototype coefficients are small, since the lobe with the smallest magnitude of coefficient values is selected as the prototype lobe. Thus, the dynamic range of the coefficient multiplier is significantly reduced. As a result, the number of distinct odd positive integers is reduced.

It should be noted from Fig. 3 that, besides the structural adders and MB adders, another two types of adders are employed in the residual compensated extrapolated filters. The first type of adder is the extrapolation adders used to add the extrapolated terms into the tap delay line; and the second type

is the scaling factor adders used to generate proper scaling factors if they are not power-of-two numbers. To minimize the number of overall adders, the scaling factors are chosen in such a way that as few adders as possible are used to reduce the number of distinct residuals to few as possible. If power-of-two numbers are used for the scaling factors, no additional scaling factor adders are required. However, in some cases, using a few scaling factor adders may achieve more saving in the residual adders.

In spite of the overhead which might be caused by the extrapolation adders and scaling factor adders, the proposed technique still significantly reduce the overall number of adders when compared with any other existing techniques.

IV. DESIGN EXAMPLE

We shall illustrate the proposed technique using an example taken from literature [9]. The discrete filter coefficient values presented in [9] are also listed as $h(n)$ in Table I for easy reference. As we have indicated in the introduction of this paper, the number of non-one distinct coefficient values (after they have been transformed to positive odd numbers) of the coefficient set of this filter is 52. Therefore, the lower bound of the MB adders is 52, which have been achieved by many algorithms [5], [10]–[12].

By inspecting the coefficient values $h(n)$ in Table I, it is noted that the filter impulse response shows a quasi-periodicity for 5 periods, from $h(4)$ to $h(48)$, with period duration of 9. The period with the minimum coefficient magnitude, i.e., from $h(40)$ to $h(48)$ is chosen as the prototype lobe to approximate the other 4 lobes. The scaling factors for lobes $h(4)$ to $h(12)$, $h(13)$ to $h(21)$, $h(22)$ to $h(30)$ and $h(31)$ to $h(39)$ are chosen to be 20, -8 , 4, and -2 , respectively. Therefore, the approximated coefficient values are given by:

$$\begin{aligned} h_a(4 + k) &= 20h(40 + k), \quad h_a(13 + k) = -8h(40 + k), \\ h_a(22 + k) &= 4h(40 + k), \quad h_a(31 + k) = -2h(40 + k), \\ \text{for } k &= 0, 1, \dots, 8. \end{aligned} \quad (8)$$

Thus, the residuals are:

$$\begin{aligned} h_r(4 + k) &= h(4 + k) - 20h(40 + k), \\ h_r(13 + k) &= h(13 + k) + 8h(40 + k), \\ h_r(22 + k) &= h(22 + k) - 4h(40 + k), \end{aligned}$$

$$h_r(31+k) = h(31+k) + 2h(40+k),$$

for $k = 0, 1, \dots, 8$. (9)

Both $h_a(n)$ and $h_r(n)$ for $4 \leq n \leq 39$ are listed in Table I.

It can be seen from Table I that the residual coefficient values from $h_r(4)$ to $h_r(39)$ are much smaller than their original values. To implement the filter in residual compensated extrapolated form, the coefficient values to be synthesized are $h(0)$ to $h(3)$, $h_r(4)$ to $h_r(39)$ and $h(40)$ to $h(60)$. Among these coefficient values, the number of distinct non-one coefficient values (after they have been transformed to positive odd numbers) have been reduced to 31. Using RAG-n algorithm [5] to generate the subexpression coefficients, the resulting MB requires 33 adders, which is close to the new lower bound of 31. The reason that 2 more adders are used than the lower bound is that some coefficient values of the center lobe are realized directly, and their magnitudes are large; more than an adder may be required to generate such coefficient values.

Symmetric coefficients of linear phase FIR filter can share the MB. However, the extrapolation adders and scaling factor adders for symmetric lobes cannot be shared with each other. Therefore, besides the MB adders, the extrapolated realization requires additional 8 adders to add the extrapolated lobes into the tap delay line, and additional 2 adders to generate the scaling factor 20. Thus, in total 43 adders are required to synthesize the filter, as shown in Table II. The best results obtained in published literature requires 52 adders; the result is also listed in Table II for comparison.

V. CONCLUSION

In this paper, an extrapolated structure with residual compensation for synthesizing linear phase FIR filters is proposed. Residual compensation provides perfect restoration of the coefficient values and therefore preserves the filter performance. The proposed technique successfully reduces the dynamic range of coefficient values, and consequently reduces the required number of adders to synthesize the filter. The choosing of scaling factors is also discussed in the light of minimizing the total number of adders.

REFERENCES

- [1] R. I. Hartley, "Subexpression sharing in filters using canonic signed digit multipliers," *IEEE Trans. Circuits Syst. II*, vol. 43, pp. 677–688, Oct. 1996.
- [2] F. Xu, C.-H. Chang, and C.-C. Jong, "Contention resolution algorithm for common subexpression elimination in digital filter design," *IEEE Trans. Circuits Syst. II*, vol. 52, pp. 695–700, Oct. 2005.
- [3] I.-C. Park and H.-J. Kang, "Digital filter synthesis based on an algorithm to generate all minimal signed digit representation," *IEEE Trans. Computer-Aided Design*, vol. 21, pp. 1525–1529, Dec. 2002.
- [4] D. R. Bull and D. H. Horrocks, "Primitive operator digital filters," *IEE Proc. G*, vol. 138, pp. 401–412, June 1991.
- [5] A. G. Dempster and M. D. Macleod, "Use of minimum-adder multiplier blocks in FIR digital filters," *IEEE Trans. Circuits Syst. II*, vol. 42, pp. 569–577, Sept. 1995.
- [6] N. Sankarayya and K. Roy, "Algorithms for lower power and high speed FIR filter realization using differential coefficients," *IEEE Trans. Circuits Syst. II*, vol. 43, pp. 488–497, 1997.

TABLE I
IMPULSE RESPONSE OF A 120-TH ORDER FILTER, $h(n)$ IS THE ORIGINAL DISCRETE COEFFICIENT VALUE OBTAINED IN [9], $h_a(n)$ IS THE EXTRAPOLATED APPROXIMATION OF THE ORIGINAL COEFFICIENT VALUE, AND $h_r(n)$ IS THE RESULTING RESIDUAL. COEFFICIENTS FROM $h(40)$ TO $h(48)$ ARE USED AS THE PROTOTYPE LOBE

n	$h(n)$	$h_a(n)$	$h_r(n)$	n	$h(n)$	$h_a(n)$	$h_r(n)$
0	14686			31	-78	-72	-6
1	-13494			32	-148	-140	-8
2	10267			33	277	260	17
3	-5927			34	-267	-250	-17
4	1653	720	933	35	143	136	7
5	1515	1400	115	36	23	16	7
6	-3000	-2600	-400	37	-152	-136	-16
7	2823	2500	323	38	193	178	15
8	-1522	-1360	-162	39	-142	-138	-4
9	-109	-160	51	40	36		
10	1336	1360	-24	41	70		
11	-1739	-1780	41	42	-130		
12	1312	1380	-68	43	125		
13	-390	-288	-102	44	-68		
14	-546	-560	14	45	-8		
15	1090	1040	50	46	68		
16	-1071	-1000	-71	47	-89		
17	585	544	41	48	69		
18	84	64	20	49	-24		
19	-616	-544	-72	50	-24		
20	798	712	86	51	55		
21	-600	-522	-48	52	-61		
22	165	144	21	53	45		
23	282	280	2	54	-18		
24	-541	-520	-21	55	-8		
25	524	500	24	56	24		
26	-277	-272	-5	57	-28		
27	-59	-32	-27	58	23		
28	323	272	51	59	-14		
29	-408	-356	-52	60	6		
30	302	276	26				

TABLE II
NUMBER OF ADDERS USED TO SYNTHESIZE THE FILTER COEFFICIENTS

	Direct Struct.	Extrap. Struct.
Adders in Multiplier Block	52 [10]	33
Extrapolation Adders	NA	8
Scaling Factor Adders	NA	2
Total Adders	52	43

- [7] O. Gustafsson and L. Wanhammar, "Improved multiple constant multiplication using minimum spanning trees," in *Proc. Asilomar Conf. Sigals, Syst., Comp.*, 2004, pp. 63–66.
- [8] O. Gustafsson, "Lower bounds for constant multiplication problems," to appear in *IEEE Trans. Circuits Syst. II*.
- [9] Y. C. Lim and S. R. Parker, "Discrete coefficient FIR digital filter design based upon an LMS criteria," *IEEE Trans. Circuits Syst.*, vol. 30, pp. 723–739, Oct. 1983.
- [10] C.-Y. Yao, H.-H. Chen, T.-F. Lin, *et al.*, "A novel common-subexpression-elimination method for synthesizing fixed-point FIR filters," *IEEE Trans. Circuits Syst. I*, vol. 51, pp. 2215–2221, Nov. 2004.
- [11] O. Gustafsson, "A difference based adder graph heuristic for multiple constant multiplication problems," in *Proc. IEEE ISCAS'07*, New Orleans, LA, 2007, pp. 1097–1100.
- [12] Y. Voronenko and M. Püschel, "Multiplierless multiple constant multiplication," *ACM Trans. Algorithms*, vol. 3, pp. 1–38, May 2007.
- [13] Y. C. Lim and B. Liu, "Extrapolated impulse response FIR filters," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 1548–1551, Dec. 1990.
- [14] Y. J. Yu, K. L. Teo, Y. C. Lim, *et al.*, "Extrapolated impulse response filter and its application in the synthesis of digital filters using the frequency-response masking technique," *Signal Processing (Elsevier)*, vol. 85/3, pp. 581–590, Mar. 2005.