THE WORDLENGTH DETERMINATION PROBLEM OF LINEAR TIME INVARIANT SYSTEMS WITH MULTIPLE OUTPUTS - A GEOMETRIC PROGRAMMING APPROACH

S. C. Chan and K. M. Tsui

Department of Electrical and Electronic Engineering,
The University of Hong Kong, Pokfulam Road, Hong Kong.

Abstract—This paper proposes two new methods for optimizing hardware resources in finite wordlength implementation of multiple-output (MO) linear time invariant systems. The hardware complexity is measured by the exact internal wordlength used for each intermediate data. The first method relaxes the wordlength from integer to real-value and formulates the design problem as a geometric programming, from which an optimal solution of the relaxed problem can be determined. The second method is based on a discrete optimization method called the Marginal Analysis method, and it yields the desired wordlengths in integer values. By combining these two methods, a hybrid method is also proposed, which is found to be very effective for large scale MO systems. Design example shows that the proposed algorithms offer better results and a lower design complexity than conventional methods.

I. INTRODUCTION

When implementing digital filters, coefficient round-off and signal round-off errors are two main sources of error [1]. The former happens when the real-valued coefficients of a digital filter are rounded to their fixed-point representations to simplify the hardware implementation. A pioneer work in addressing the coefficient round-off problem and efficient realization of digital filters is due to Lim *et al.* [2], where the filter coefficients are represented as sum-of-power-of-two (SOPOT) or canonical signed digits representations. Since then, various algorithms for determining these SOPOT coefficients were proposed.

On the other hand, signal round-off error occurs when overflow occurs due to insufficient internal wordlength and improper scaling; and when rounding is performed for long intermediate data after multiplications with the filter coefficients. To satisfy a given output accuracy, one usually employs a fixed and long wordlength for all intermediate data, which increases the hardware complexity. In [3], a flexible approach using a random search algorithm was proposed to minimize the hardware complexity of a two-channel filter bank (FB) system while satisfying the given output accuracy. However, its searching time will increase considerably when large number of variables is involved since the searching method is random in nature. A more recent work in [4] recognized the close relationship between the wordlength determination problem and bit allocation problem for data compression [5] – [7] and proposed two algorithms, namely Lagrange multiplier and marginal analysis methods, to solve the problem. These approaches are very effective and they are well-suited to large scale single-output (SO) linear time invariant (LTI) systems. Unfortunately, they cannot be directly applied to multiple-output (MO) LTI systems, because it gives rise to a set of nonlinear equations which has to be solved using numerical method.

In this paper, we reformulate the wordlength determination problem as a geometric programming (GP) problem by relaxing the wordlength from integer to real-value. The main advantage of GP is that the problem is convex and the global optimal solution, if it exists, is guaranteed. Furthermore, additional constraints can be imposed readily to meet different design objectives and constraints. The GP approach is very general and it allows us to solve optimally the relaxed wordlength determination problem of MO LTI systems subject to a prescribed output accuracy or bit budget (total wordlength). The hardware complexity is measured by the exact internal wordlength for each intermediate data and the output accuracy is determined statistically by its output noise power resulting from the rounding operations. Note that our approach can be regarded as a generalization of the approach in [4] where only SO LTI systems were considered. While undesirable negative solution may appear in Lagrange multiplier method in [4] for low target bit accuracy or bit budget, it can be completely avoided by imposing appropriate constraints under the GP framework. Since the solution of the GP approach is generally not integer-valued, it has to be rounded to integers for practical implementation. Motivated by the Marginal Analysis (MA) method in [4], we further extended it to MO LTI systems where the solution is integer-valued. The basic idea is to increase the wordlength of one of the rounding sources successively in order to lower the output round-off noise power as much as possible, until the given bit accuracy or bit budget is met. A simple but effective hybrid approach is also proposed where the GP solution is used as the initial guess to the MA algorithm to further reduce the computational time. Design results show that the MA method gives the closest integer solution to the optimal GP solution and has a much lower design time than the random search algorithm. The hybrid method reduces the design time further, while achieving an almost identical performance.

The paper is organized as follows: Section II gives an overview of GP. The signal round-off error model and overflow handling are briefly reviewed in Section III. Section IV describes the proposed algorithms for determining the internal wordlength using the GP and MA methods. This is then followed by a design example and comparison to illustrate the effectiveness of the proposed approaches in Section V. Finally, conclusion is drawn in Section VI.

II. OVERVIEW OF GEOMETRIC PROGRAMMING

Geometric programming (GP) has recently received considerable attention as a flexible and powerful framework for solving a wide variety of engineering problems. Interested readers are referred to [9] and references therein for more information of GP. Before proceeding to standard formulation of a GP problem, two types of typical functions used in GP, namely monomial and posynomial functions, are first introduced. A monomial function, $p(\boldsymbol{x})$, of N real positive variables, $\boldsymbol{x} = (x_1, \ldots, x_N)$, has the form:

$$p(\boldsymbol{x}) = c x_1^{a_1} x_2^{a_2} \cdots x_N^{a_N},$$

where $c > 0$ and a_n is a real number for $n = 1, \ldots, N$. A posynomial function, $q(\boldsymbol{x})$, is a sum of monomial functions with the form:

$$q(\boldsymbol{x}) = \sum_{k=1}^{K} p_k(\boldsymbol{x}) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_N^{a_{Nk}}.$$

In the standard form of GP, a posynomial objective function of design variables $\boldsymbol{x}=(x_1,\ldots,x_N)$, $q_0(\boldsymbol{x})$, is minimized subject to a series of posynomial inequality constraints and monomial equality constraints having the form $q_i(\boldsymbol{x})\le 1$ and $p_i(\boldsymbol{x})=1$, respectively. That is:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & q_0(\boldsymbol{x}) \\ \text{subject to} \quad & q_i(\boldsymbol{x})\le 1, \quad i=1,\ldots,N_q, \\ & p_i(\boldsymbol{x})=1, \quad i=1,\ldots,N_p, \\ & x_n>0, \qquad n=1,\ldots,N. \end{aligned}$$

Since the GP problem is convex, the optimality of the solution, if there is any, can be guaranteed. Moreover, GP can be solved efficiently even with large number of variables and constraints [9].

III. Signal Round-off and Overflow Analysis

A. — Signal Round-off Analysis

Signal round-off errors occur due to rounding of the intermediate signal after multiplications. Since the exact round-off errors are difficult to analyze, they are usually treated as uncorrelated white noises. In fixed-point arithmetic, each intermediate signal can be represented in the form of $<I/F>$, where I is the number of integer bits including the sign bit and F is the number of fractional bits. If F bits are rounded to B bits, where $B<F$, then the quantization noise is assumed to have a zero mean and a variance P_e equal to:

$$P_e=\frac{\Delta^2}{12}=\frac{2^{-2(B-1)}}{12}=\frac{2^{-2B}}{3}, \tag{1}$$

where $\Delta=2^{-(B-1)}$. Without loss of generality, consider the round-off noise model of a J-output MO LTI system in Fig. 1, where the signals to be quantized are $s_i[n]$ for $i=1,\ldots,M$; M is the total number of rounding sources. From (1), if $s_i[n]$ is rounded to b_i bits, then the variance of the quantization error, $e_i[n]$, is given by $2^{-2b_i}/3$. Note that this error can affect more than one output depending on the internal structure of the system, such as the radix-p Fast Fourier Transform and polyphase structured M-channel FB systems. Let the transfer function from $s_i[n]$ to the j-th output $y_j[n]$ be $H_{j,i}(\omega)$, $i=1,\ldots,M$ and $j=1,\ldots,J$. Furthermore, we assume that the noise sources are uncorrelated. Hence the variance of the output noise at $y_j[n]$ can be expressed as follows:

$$\sigma_{e_j}^2=\sum_{i=1}^{M}c_{j,i}\sigma_i^2=\frac{1}{3}\sum_{i=1}^{M}c_{j,i}2^{-2b_i}, \quad j=1,\ldots,J, \tag{2}$$

where $c_{j,i}=\frac{1}{2\pi}\int_{-\pi}^{\pi}|H_{j,i}(\omega)|^2\,d\omega=\sum_k|h_{j,i}[k]|^2$; $H_{j,i}(\omega)$ is the transfer function from $s_i(n)$ to $y_j(n)$; and $h_{j,i}(k)$ is the impulse response corresponding to $H_{j,i}(\omega)$. The accuracy at the j-th output, in terms of the number of fractional bits, is then approximately given by:

$$A_j\approx\left|10\cdot[\log_{10}(\sigma_{e_j}^2)+\log_{10}(3)]\right|/6 \text{ bits}, \quad j=1,\ldots,J. \tag{3}$$

It should be noted that the larger the number of noise sources, the lower will be the accuracy. The noise power can however be reduced by increasing the internal wordlengths for the fractional bits, at the expense of increased hardware complexity.

B. — Overflow Handling

Signal overflows occur when the allocated wordlength of the integer bits is insufficient to accommodate the growth in integer wordlength of the signal after additions. In FIR filters, it is possible to determine whether signal overflow will occur at a particular adder using the L1 scaling measure. More precisely, the input signal $x[n]$ is assumed to take on its maximum value denoted by $x_{\max}$. Then, the maximum value after implementing the *k-th* impulse response coefficient of the target system is bounded by:

$$y_{\max,k}=x_{\max}\sum_{n}^{k}|h[n]|. \tag{4}$$

Using (4), it is possible to determine the worst-case integer wordlength of each adder and hence the size of its output register to avoid signal overflow. There is, however, an option to retain or decrease the number of bits in the fractional part, depending on the required output accuracy. To determine this option, we can imagine that a noise is generated by the rounding option and the minimum acceptable wordlength is then determined as if it was a rounding source due to multiplication. If the minimum wordlength obtained is larger than the existing wordlength, then the wordlength has to be increased. Otherwise, rounding can be performed if the additional noise generated does not violate the prescribed accuracy. In IIR filters, scaling is usually performed at certain stages of the system to avoid overflow. Since scaling is a multiplication operation, it can be treated similarly by our model.

IV. Wordlength Determination

A. —Geometric Programming Problem Formulation

The problem of determining the wordlengths for a given output noise power $\sigma_{e_j}^2=P_j$ at the j-th output can be formulated as the following constrained optimization problem:

$$\begin{aligned} &\min_{\boldsymbol{b}} \; f_0(\boldsymbol{b})=\sum_{i=1}^{M}w_ib_i=\boldsymbol{w}^T\boldsymbol{b} \\ &\text{subject to } f_j(\boldsymbol{b})=\frac{1}{3}\sum_{i=1}^{M}c_{j,i}2^{-2b_i}-P_j\le 0, \; j=1,\ldots,J \end{aligned} \tag{5}$$

where $\boldsymbol{w}$ is a constant weight vector, and $\boldsymbol{b}$ is the variable vector representing the fractional part of the internal wordlengths to be determined. In most cases, w_i are chosen as one for all i. If we allow $\boldsymbol{b}$ to take on real values instead of integer values and consider only one output (i.e. $J=1$), then the minimization problem in (5) can be solved analytically using the method of Lagrange multiplier [4]. However, for extremely large target variance or low bit accuracy, negative solution may appear, which should be avoided.

In this paper, we shall reformulate the above problem as a GP problem. Define a new variable $x_i=2^{2b_i}$, then the objective function in (5) can be written as:

$$f_0(\boldsymbol{x})=\sum_{i=1}^{M}w_ib_i=\sum_{i=1}^{M}\ln(x_i)^{w_i/(2\ln 2)}=\ln\left[\prod_{i=1}^{M}(x_i)^{w_i/(2\ln 2)}\right]. \tag{6}$$

Since the logarithm function is monotonic, minimizing $f(\boldsymbol{x})$ is equivalent to minimizing the following monomial:

$$g_0(\boldsymbol{x})=\prod_{i=1}^{M}(x_i)^{w_i/(2\ln 2)}. \tag{7}$$

Similarly the constraints, $f_j(\boldsymbol{b})$, can be written as:

$$g_j(\boldsymbol{x})=\frac{1}{3P_j}\sum_{i=1}^{M}c_{j,i}x_i^{-1}\le 1, \; j=1,\ldots,J, \tag{8}$$

where $g_j(\boldsymbol{x})$ are posynomial functions. This is a standard geometric programming problem and is convex. Moreover, undesirable negative result of b_i can be completely avoided by imposing the constraints $x_i\ge 1$, $i=1,\ldots,M$. Hence, the equivalent GP problem is given by:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & g_0(\boldsymbol{x}) \\ \text{subject to} \quad & g_j(\boldsymbol{x}) \le 1, \quad j = 1,\ldots,J, \\ & x_i \ge 1, \qquad i = 1,\ldots,M. \end{aligned} \tag{9}$$

Alternatively, we can minimize $\sigma_{e_j}^2$, $j = 1,\ldots,J$, subject to a given bit budget: $\sum_{i=1}^{M} w_i b_i = B_{spec}$. Since each noise source would affect more than one output, the design problem is equivalent to:

$$\begin{aligned} \min_{\boldsymbol{b}} \quad & \max\left\{\frac{1}{3}\sum_{i=1}^{M} c_{j,i} 2^{-2b_i},\ j = 1,...,J\right\} \\ \text{subject to} \quad & \sum_{i=1}^{M} w_i b_i = B_{spec}. \end{aligned} \tag{10}$$

Using again the change of variable, the constrained minimization problem above can be written as follows:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \max\left\{\frac{1}{3}\sum_{i=1}^{M} c_{j,i} x_i^{-1},\ j = 1,...,J\right\} \\ \text{subject to} \quad & e^{-B_{spec}} \prod_{i=1}^{M} (x_i)^{w_i/(2\ln 2)} = 1, \\ & x_i \ge 1, \qquad i = 1,\ldots,M. \end{aligned} \tag{11}$$

It is shown in [9] that the objective function in (11) can also be handled using GP by introducing new variables and bounding constraints. Here, we introduce a new variable t and rewrite (11) as follows:

$$\begin{aligned} \min \quad & t \\ \text{subject to} \quad & \frac{1}{3}\sum_{i=1}^{M} c_{j,i} x_i^{-1} \le t, \quad j = 1,\ldots,J, \\ & e^{-B_{spec}} \prod_{i=1}^{M} (x_i)^{w_i/(2\ln 2)} = 1, \\ & x_i \ge 1, \qquad i = 1,\ldots,M, \end{aligned} \tag{12}$$

which is also a standard GP. It should be noted that the solutions of GP formulation above and the method of Lagrange multiplier in [4] b_i are real-valued. To obtain an integer solution, a simple way is to round the solutions to the next largest integers. On the other hand, as pointed out in [4], the wordlength determination problem for SO LTI system is similar to the classical problem in signal compression and hence another method based on the MA s method [6], [7] can be used to obtain the integer solution. Next, we shall extend this approach to MO LTI systems.

B. — Bit Allocation Algorithm

To solve the problem in (5) using the MA method, the variable b_i is first initialized to zero. Then the algorithm allocates one bit to one of b_i's until the target noise powers at all outputs are met. In each step, the one with the largest reduction in total noise power at one of the outputs is selected and its wordlength will be increased by one bit. More precisely, the pseudo code of the algorithm can be summarized as follows:

$b_i = 0$; (or $b_i = floor(b_i^{opt})$; // from (9))
while ($\frac{1}{3}\sum_{i=1}^{M} c_{j,i} 2^{-2b_i} > P_j$, $j = 1,\ldots,J$) {
  compute $k = \arg\max_i \xi_i$
  $\xi_i = \max\left\{\left|\frac{c_{j,i}(2^{-2(b_i+1)} - 2^{-2b_i})}{w_i}\right|,\ j = 1,\ldots,J\right\}$;
  $b_k \leftarrow b_k + 1$;}

Table 1: Minimization of $\boldsymbol{w}^T\boldsymbol{b}$ subject to prescribed noise power P_j.

Note that b_i's are both non-negative and integer-valued. A similar algorithm for minimizing $\sigma_{e_j}^2$ subject to a given bit budget B_{spec} can be derived as follows:

$b_i = 0$; (or $b_i = floor(b_i^{opt})$; // from (12))
while ($\sum_{i=1}^{M} w_i b_i < B_{spec}$) {
  compute $k = \arg\max_i \xi_i$
  $\xi_i = \max\left\{\left|\frac{c_{j,i}(2^{-2(b_i+1)} - 2^{-2b_i})}{w_i}\right|,\ j = 1,\ldots,J\right\}$;
  $b_k \leftarrow b_k + 1$;}

Table 2: Minimization of $\sigma_{e_j}^2$ subject to prescribed bit budget B_{spec}.

Again, b_i are both non-negative and integer-valued. For multiplier-less realization using SOPOT coefficients, one can easily compute the wordlength required to achieve a given output error variance. Once it is determined, the exact rounding operation at each node can be determined and hence the complexity of the adders and registers can be determined exactly. The overflow prevention can also be determined according to section III-B if the maximum input format is known. Finally, the algorithms described in sections IV-A and IV-B can be combined to yield a hybrid algorithm, which shortens significantly the search time, as we shall illustrate in next section.

V. Design Example

In this example, all wordlength determination algorithms, including the random search algorithm proposed in [3], are implemented using Matlab Ver. 6.0 in a Pentium 4 personal computer. The GP optimization was carried out using the MOSEK Matlab Toolbox [10] and it took a few seconds to obtain a solution. For comparison purposes, the hardware implementation issue of a two-channel structural perfect reconstruction FB [11] having the same specification of example 1 in [3] was considered. Fig. 2 shows the structure of the analysis bank under consideration, which are parameterized by two subfilters $\beta(z)$ and $\alpha(z)$. It can be seen that the internal wordlengths of the former would affect the accuracy of both outputs. Since the coefficients of both subfilters are fixed, they can be implemented without any multiplications using SOPOT representation. Table 3 summarizes the SOPOT coefficients of $\beta(z)$ and $\alpha(z)$. Furthermore, by implementing the filters using multiplier-block (MB) [12], significant savings in hardware resources can be achieved. Once the SOPOT coefficients are determined, the transfer functions of these filters are known. The internal wordlength are then minimized subject to the prescribed 16-bit accuracy using the random search algorithm. The results so obtained are summarized as follows: $\boldsymbol{w}^T\boldsymbol{b} = 544$; $\sigma_{e_1}^2 = 0.308\times10^{-10}$ ($A_1 = 16.723$); $\sigma_{e_2}^2 = 0.648\times10^{-10}$ ($A_2 = 16.186$) and the computational time is about 5 minutes. Interested reader are referred to [3] for more details regarding the design aspects of the two-channel FB, such as the determination of SOPOT coefficients and the optimization of the internal wordlengths subject to the prescribed output accuracy using the random search algorithm. Next we shall employ the proposed wordlength determination algorithms to solve the same problem.

With the same specification in [3], there are totally $M = 29$ rounding sources in the FB in Fig. 2. Using the GP formulation in (9), the optimal wordlength format for each intermediate signal is obtained and the optimal value of $\boldsymbol{w}^T\boldsymbol{b}$ is found to be

518.9. Note that the GP algorithm also works well for a large-scale system, thanks to the efficiency and reliability of GP. However, as mentioned earlier, the entries of the vector $\boldsymbol{b}$ are not integer-valued. Therefore, for practical implementation, they are rounded to the closest integer just larger than them such that the 16-bit accuracy is still met. The corresponding value of $\boldsymbol{w}^T\boldsymbol{b}$ becomes 491 and the output noise powers $\sigma_{e_1}^2$ and $\sigma_{e_2}^2$ (output accuracies A_1 and A_2) are respectively decreased (increased) to 0.170×10^{-10} and 0.338×10^{-10} (17.155 and 16.657). For the MA method, we obtain the following results in Table 4: $\boldsymbol{w}^T\boldsymbol{b}=525$; $\sigma_{e_1}^2=0.170\times10^{-10}$ ($A_1=17.155$); $\sigma_{e_2}^2=0.810\times10^{-10}$ ($A_2=16.023$) and the computation time is about 0.7 second. This algorithm gives the closest integer solution to the optimal GP solution among the three algorithms studied and also has a much lower computational time than the random search algorithm. In order to avoid overflow, the worst-case integer bit format of each intermediate signal can then be calculated as described in section IV-B, assuming that the input signal $x[n]$ has a format of <1/13>, i.e. 14 bits with $x_{\max}=0.99988$. The outputs $y_1[n]$ and $y_2[n]$ are found to have wordlength formats of <3/19> and <4/19>, respectively. The wordlength formats of each output are also shown in Fig. 2. In the hybrid approach, the solution obtained from (9) is used as an initial guess to the MA method. This further reduces the computational time of the MA method. Moreover, it is expected that the computational saving would be substantial when the hybrid approach is applied to large scale MO LTI systems.

VI. Conclusion

Two novel methods for the wordlength determination of MO LTI systems subject to a prescribed output accuracy or bit budget are presented. The first one is able to determine an optimal solution of this problem using GP, assuming that the wordlength is a real-valued quantity. The second one is based on the MA method and gives integer-valued solution. A hybrid method, which combines these methods, is also proposed for large scale MO systems. Using the two-channel FB as an example, design results show that proposed approaches offer better results and a lower design complexity than conventional methods.

References

[1] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing*, Englewood Cliffs, NJ: Prentice-Hall, 1989.

[2] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete power-of-two coefficient space," *IEEE Trans. ASSP-31*, pp. 583-591, April 1983.

[3] Carson K. S. Pun, S. C. Chan, K. L. Ho, "Efficient design of a class of multiplier-less perfect reconstruction two-channel filter banks and wavelets with prescribed output accuracy," in *Proc.* of *the 11th IEEE Workshop on Statistical Signal Processing*, pp. 599-602, 2001.

[4] S. C. Chan and K. M. Tsui, "Wordlength Determination Algorithms for Hardware Implementation of Linear Time Invariant Systems with Prescribed Output Accuracy," in *Proc. IEEE ISCAS'2005*, pp. 2607-2610, May 23-26, 2005.

[5] A. Segal, "Bit allocation and encoding of vector resource," *IEEE Tran. Info. Theory*, vol. 22, pp. 162-169, Mar. 1976.

[6] B. Fox, "Discrete optimization via marginal analysis," *Management Science*, vol. 13, pp. 210-216, Nov. 1966.

[7] S. W. Wu and A. Gersho, "Rate-constrained picture-adaptive quantization for JPEG baseline coders," in *Proc. IEEE ICASSP'1993*, vol. 5, pp. 390-392, 1993.

[8] S. Boyd and L. Vandenberghe, "Convex optimization," *Cambridge University Press*, 2004.

[9] S. Boyd, S.-J. Kim, L. Vandenberghe, and A. Hassibi "A tutorial on geometric programming," 2004. Available from www.stanford.edu/~boyd/gp_tutorial.html.

[10] MOSEK ApS. The MOSEK Optimization Tools Version 2.5. User's Manual and Reference, 2002. Available from www.mosek.com.

[11] S. M. Phoong, C. W. Kim, P.P. Vaidyanathan and R. Ansari, "A new class of two-channel biothogonal filter banks and wavelet bases," *IEEE Trans. Signal Processing*, vol. 43, pp. 649-664, Mar. 1995.

[12] A. G. Dempster and M. D. Macleod, "Use of minimum-adder multiplier blocks in FIR digital filters," *IEEE Trans. CAS. II*, vol. 42, pp. 569-577, Sept. 1995.

n	$\beta(z)$	$\alpha(z)$
0	$2^{-5}+2^{-7}$	-2^{-7}
1	$-2^{-3}+2^{-6}-2^{-8}$	$2^{-5}-2^{-7}$
2	$2^{-1}+2^{-5}+2^{-8}$	$-2^{-4}+2^{-6}-2^{-8}$
3	$2^{-0}-2^{-2}-2^{-6}$	$2^{-4}+2^{-6}+2^{-8}$
4	$-2^{-2}-2^{-4}+2^{-7}$	$-2^{-2}+2^{-4}+2^{-7}$
5	$2^{-2}-2^{-5}-2^{-7}$	$2^{-1}+2^{-3}-2^{-6}$
6	$-2^{-3}-2^{-5}$	$2^{-1}+2^{-3}+2^{-5}$
7	$2^{-3}-2^{-7}$	$-2^{-2}+2^{-5}$
8	$-2^{-3}+2^{-5}+2^{-8}$	$2^{-3}-2^{-9}$
9	2^{-4}	$-2^{-4}-2^{-6}$
10	$-2^{-5}-2^{-6}+2^{-8}$	$2^{-4}-2^{-6}+2^{-9}$
11	$2^{-5}-2^{-7}$	$-2^{-5}+2^{-8}$
12	-2^{-6}	2^{-6}
13		-2^{-7}
14		2^{-8}

Table 3. SOPOT coefficients of $\beta(z)$ and $\alpha(z)$.

	Time	$\boldsymbol{w}^T\boldsymbol{b}$	Output bit accuracy	
			A_1	A_2
Random search algorithm [3]	~5 min	544	16.723	16.186
GP solution	~0.4 sec	518.9	16.464	16.000
Rounded GP solution	~0.4 sec	538	17.155	16.657
Marginal Analysis algorithm	~0.7 sec	525	17.155	16.023
Hybrid approach	~0.5 sec	525	17.155	16.023

Table 4. Summary of design results.

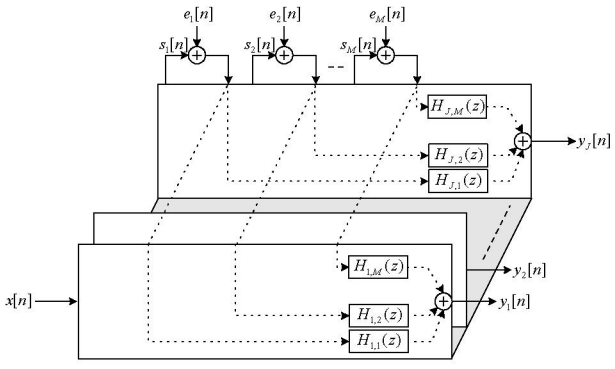

Fig. 1: Round-off noise model of multiple-output linear time invariant system.

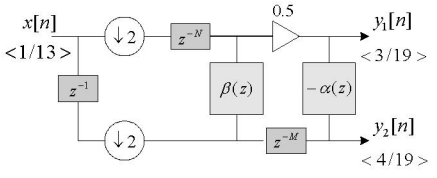

Fig. 2: Structure of structural perfect reconstruction filter bank (analysis filter) [11].