J. M. A. Tanskanen and V. S. Dimitrov, "Round-off error free fixed-point design of polynomial FIR predictors," in *Proc. The 33rd Asilomar Conference on Signals, Systems, and Computers*, CA, USA, Oct. 1999, in press.

Round-off Error Free Fixed-Point Design of Polynomial FIR Predictors

Jarno M. A. Tanskanen[1] and Vassil S. Dimitrov[2]

[1]Institute of Intelligent Power Electronics
Department of Electrical and Communications Engineering
Helsinki University of Technology, Espoo, Finland
Tel. +358-9-451 2446, Fax: +358-9-460 224, E-mail: jarno.tanskanen@hut.fi

[2]Laboratory of Signal Processing and Computer Technology
Department of Electrical and Communications Engineering
Helsinki University of Technology, Espoo, Finland
Tel. +358-9-451 2455, Fax: +358-9-460 224, E-mail: vdimitro@wooster.hut.fi

Abstract

In this paper, we present a novel method for designing polynomial FIR predictors for fixed-point environments. Our method yields filters that perform exact prediction of polynomial signals even with short coefficient word lengths. Under ordinary coefficient truncation or rounding, prediction capability degrades, or may be totally lost. With the proposed method, the filters are designed so that the predictive properties are exactly preserved in fixed-point implementations. The proposed filter design method is based on integer programming (IP) and can be directly applied to any fixed-point FIR design specifications which can be formulated in a form of linear constraints on the filter coefficients.

1. Introduction

By their nature, digital devices handle numbers using a finite number of bits per digit [1]. In many embedded applications using highly optimized, small and less power consuming application specific integrated circuits (ASICs) it would be desirable to get by with low precision fixed-point arithmetic but having only a very limited number of bits available for presenting filter coefficients results in filter quality degradation and possibly even in a totally unintended kind of filtering operation. In this paper, we present a novel method for designing polynomial FIR predictors [2] whose quantized coefficients exactly fulfill the set constraints and provide for exact prediction even with coefficient precision of 6 bits (4 in some cases).

Our application examples include motion control of an elevator car [3], and mobile phone power control [4]. In these and many other practical applications, measured signals can be accurately modeled as piecewise polynomials buried in noise. Also the closed control loops employed in these applications are inherently delay limited. Hence, polynomial-predictive noise filtering is a naturally lucrative approach. *Our main goal in this paper is to preserve the exact prediction step and dc-gain with quantized coefficient polynomial FIR predictors.* As the method presented in this paper does yield quantized-coefficient filters that exactly preserve the prediction step and dc-gain, and as they are FIR filters, the designed filters are naturally safe for even critical applications in short word length fixed point environments.

2. Polynomial FIR predictors in fixed-point environments

2.1 Polynomial FIR predictors

Polynomial predictive filtering theory has been well established [2,3,4,5] but applicability of polynomial FIR predictors has suffered from the practical constraint of finite coefficient precision. Polynomial FIR predictors, derived in [2], assume a low-degree polynomial input signal contaminated by white noise. Filter output is defined to be a p-step-ahead predicted input,

$$\sum_{k=1}^{N} h(k)x(n-k+1) = x(n+p) \qquad (1)$$

where $h(k)$ are filter coefficients, $x(n)$ are input samples, N is filter length, and p is prediction step. After providing for exact prediction, the rest of the degrees of freedom are used to minimize the white noise gain,

$$NG = \sum_{k=1}^{N} |h(n)|^2 \ . \tag{2}$$

A set of constraints can be derived from the definition of the filter output (1) [2]:

$$g_0 = \sum_{k=1}^{N} h(k) - 1 = 0 \tag{3}$$

$$g_1 = \sum_{k=1}^{N} k h(k) = 0 \tag{4}$$

$$g_2 = \sum_{k=1}^{N} k^2 h(k) = 0 \tag{5}$$

$$\vdots$$

$$g_M = \sum_{k=1}^{N} k^M h(k) = 0 \tag{6}$$

The constraints (3)-(6) yield prediction of the polynomial degrees 0, …, M, and from them can closed form solutions for the FIR coefficients for low-degree polynomial input signals be calculated by the method of Lagrange multipliers [6]. The closed form solutions for FIR coefficients for the first, second, and third degree polynomial input signals are given in [2]. Since prediction of a first degree polynomial signal is, in a way, trivial from an application point of view, in this paper we consider a case with the highest polynomial input signal component degree of two, $M = 2$, as an example. In this case we have to fulfill the constraints (3), (4) and (5), and use the remaining degrees of freedom to minimize the noise gain (2). The exact, i.e., infinite precision, coefficients for the one-step-ahead predictive $p = 1$ second degree polynomial $M = 2$ FIR predictors are given by [2]

$$h(k) = \frac{9N^2 + (9 - 36k)N + 30k^2 - 18N + 6}{N^3 - 3N^2 + 2N}, \quad k = 1, 2, ..., N. \tag{7}$$

In [5], a feedback extension to FIR differentiators is given to provide considerable noise attenuation while maintaining the prediction property set forth by the underlying FIR predictor. In order for the feedback extension to function properly, it is necessary that the underlying FIR basis filters are implemented exactly. Until now, this has been rarely possible in short word length fixed-point environments.

2.2. Coefficient quantization effects

For fixed-point presentation of filter coefficients, two's complement presentation is used, and as the conventional quantization method for the filter coefficients (7), magnitude truncation is applied. In our calculations, 'infinite precision' means the computational precision of Matlab.

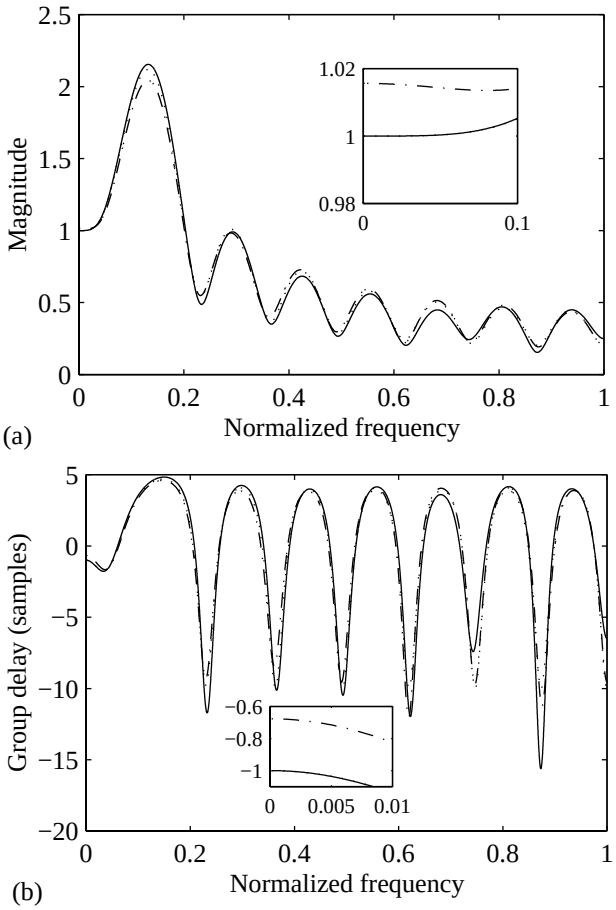

Fig. 1. Frequency response (a) and group delay (b) of the second degree polynomial FIR predictor of length $N = 16$ with coefficients truncated (dash-dot) and ideally quantized (solid) to 6 bits, along with the same filter with infinite precision coefficients (dotted).

The quantization effects can be seen in Fig. 1. In Fig. 1, frequency response and group delay of a second degree polynomial $M = 2$ one-step-ahead $p = 1$ predictor of length $N = 16$ is shown with infinite, and conventionally and ideally quantized 6-bit coefficients. From Fig. 1 it seen that the exact one-step-ahead prediction at zero frequency is lost, and also that the dc-gain is deviated from unity. The one-step-ahead prediction property can be seen as the negative unity group delay at zero frequency, Fig. 1b. Let us note that the coefficients (7) for the $p = 1$, $M = 2$, polynomial FIR predictor of

length $N = 3$ are still exact if quantized to six bits but the frequency response of that filter is not practical for usual applications, and longer filters are needed to provide for better noise attenuation.

3. Polynomial FIR predictor design by linear diophantine equation based solution

The optimization problem that has to be solved, i.e., *solving* (*3*)-(*6*) *exactly for truncated coefficients* $h(k)$ *and* thereafter *minimizing the noise gain* (*2*), can be reformulated as *an integer programming problem*. Suppose that all the coefficients of the filter, $h(k)$, are multiplied by 2^n, where n is the number of bits available, and truncated to yield integer coefficients $h_0^*(k)$ where the asterisk denotes an integer quantity.

The optimization task can now be defined as follows:

Input: Function (2)

$$NG = F(h^*(1), h^*(2), \cdots, h^*(N-1)) = \sum_{k=1}^{N} h^{*2}(k) \qquad (8)$$

with integer variables $h^*(k)$. The constraint conditions (3)-(6) for the coefficients can be formulated as:

$$g_0 = \sum_{k=1}^{N} h^*(k) = 2^n \qquad (9)$$

$$g_1 = \sum_{k=1}^{N} k h^*(k) = 0 \qquad (10)$$

$$g_2 = \sum_{k=1}^{N} k^2 h^*(k) = 0 \qquad (11)$$

$$\vdots$$

$$g_M = \sum_{k=1}^{N} k^M h^*(k) = 0 \qquad (12)$$

Output: An integer vector, $\mathbf{h}^* = (h^*(1), h^*(2), \ldots, h^*(N))$ that minimizes *NG* (8) and satisfies *exactly* the constrains (9)-(12), and thus also (3)-(6).

The solution we offer is based on the following considerations:

1. The task in hand is a quadratic integer programming problem, which is well-known to be an NP-complete problem [7,8,9]; therefore it is unrealistic to find the best solution in a reasonable amount of time, especially for long filters. This state of affairs is in sharp contrast to the quadratic real programming problem [8], which is solvable in polynomial time.
2. Without restricting the variables to be integers, we have a closed form solution of the problem, which is given by (7) for the case $p = 1$, $M = 2$. Although the values computed by this formula are not integers, this expression gives us a very good initial approximation. Though it is at best difficult to prove, it is reasonable to assume that a solution of (9)-(12) would lie in a vicinity of the infinite precision solution (7).
3. To make sure that the conditions (9)-(12) are met exactly, one has to solve the above system in integers. This problem has been a subject of very deep investigations in number theory and the theory of Diophantine equations. By eliminating the variables, one can reduce the problem to a single linear equation of the form:

$$A_1 x_1 + A_2 x_2 + \cdots + A_l x_l = B \qquad (13)$$

where $A_1, A_2, \ldots, A_l$ and B are integers.

The solutions of (13) are usually obtained by multidimensional continued fraction algorithms [10,11], and the reader can find a large variety of methods aimed at solving this class Diophantine equations. Here our approach is based on Clausen-Fortenbacher algorithm [12]. The reasons why we chose this particular technique are: firstly, the algorithm succeeds in finding very fast the solutions of (9)-(12), from which the optimal one, that is, the one that minimizes the noise gain *NG* (8), can be quickly found; secondly, the program provided in [12] can be easily generalized to more than 16 variables (the largest case analyzed by Clausen and Fortenbacher); thirdly, we have a good initial approximation that significantly speeds us the algorithm.

Since the filters shown in this paper are not very long, the algorithm is applied as a exhaustive search for the ideally quantized coefficients that fulfill (9)-(11) over a search band of ±2 from the normally quantized coefficients presented in integer form. This search band is illustrated in Fig. 2 in real number form for the filter length $N = 16$ with coefficient precision of 6 bits along with the conventionally and ideally quantized 6-bit coefficients for the $p = 1$, $M = 2$, $N = 16$ filter. It is worth noting that in our experiments, truncating or rounding the infinite precision coefficients has never but once produced a solution of the system of the Diophantine equations (9)-(12); the filter of the length $N = 3$ is an exception. This demonstrates the necessity of special techniques aimed at solving the integer optimization problem.

In Table 1, the $p = 1$, $M = 2$, $N = 16$, FIR predictor coefficients $h(k)$ (7) are shown in real number form (infinite precision, multiplied by 2^n and truncated to 6 decimals) along with the best 6-bit ideal integer solution obtained within the band shown in Fig. 2. In the example

in Table 1 it can be seen that for 8 out of 16 coefficients one had to approximate the real coefficient with an integer that is not the closest one. Also, 10 out of the 16 ideally quantized coefficients differ from the corresponding magnitude truncated real number form coefficients in Table 1.

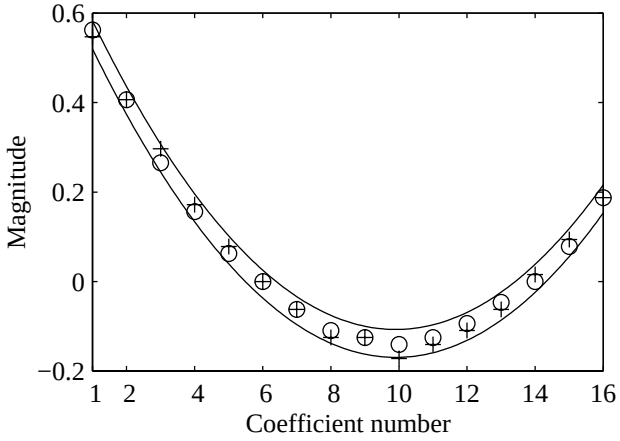

Fig. 2. Ideal quantization search band (between solid lines) for the filter length $N = 8$ with the coefficient precisions of 6 bits. Circles 'o' denote the truncated, and plusses '+' the ideally quantized coefficients.

Table 1. The infinite precision presentation (real number form, truncated to 6 decimals) of the digital filter coefficients computed by (7) for the filter length $N = 16$ and their best integer solutions within the search band that guarantees the exact solution of (9)-(11) while also minimizing the noise gain (8) with the coefficient precision of 6 bits.

Coeff.	Real number form	Best int. solut.	Coeff.	Real number form	Best int. solut.
64 $h(1)$	36.000000	35*	64 $h(9)$	-8.800000	-8*
64 $h(2)$	26.400000	26	64 $h(10)$	-9.257143	-11**†
64 $h(3)$	17.942857	19**†	64 $h(11)$	-8.571429	-9†
64 $h(4)$	10.628571	11†	64 $h(12)$	-6.742857	-7†
64 $h(5)$	4.457143	5*†	64 $h(13)$	-3.771429	-4†
64 $h(6)$	-0.571429	0*	64 $h(14)$	0.342857	1*†
64 $h(7)$	-4.457143	-4	64 $h(15)$	5.600000	6†
64 $h(8)$	-7.200000	-8*†	64 $h(16)$	12.000000	12

* The best integer solution is not the integer closest to the real (infinite precision) coefficient value.

** The best integer solution is not the nearest integer on either side of the real (infinite precision) coefficient value.

† The best integer solution is not the magnitude truncated real number form coefficient.

Table 2 lists the numbers of solutions that exactly satisfy the constraints (9)-(11) for coefficient precisions 6, 8, 10, 12, 14, and 16 bits for the filter lengths $N = 8$ and $N = 16$. To find the optimum solution, it is necessary of search all the solutions and to select the one which minimizes noise gain (8). Within the search band of ±2, for filter of lengths of $N = 8$ and $N = 16$, there are $4^8 = 65536$, and $4^{16} = 4.294967296 \cdot 10^9$ possible coefficients vectors to be tested against the constraints (9)-(12), respectively. For the filter length $N = 8$, the search and noise gain minimization takes less than one second on a 166 MHz Pentium processor using exhaustive search programmed with C language while for $N = 16$ the time is around 47 minutes. This clearly demonstrates for the necessity of applying efficient algorithms to solve the Diophantine equation (13) when designing longer filters with $N = 50, \ldots, 100$. For many applications, also the first-found solution could most probably be adequate, which should be checked by comparing the noise gain against the noise gain of the corresponding infinite precision filter, and the application at hand. By starting the search with coefficients initially approximated towards zero, it is possible to design a search algorithm whose first found solution is at least not one of the highest noise gain solutions. Table 2 demonstrates that there truly are several exact solutions of (9)-(11) within a close vicinity of the infinite precision coefficients, and that there are still some degrees of freedom left for minimizing the noise gain (8) after fulfilling the constraints (9)-(11).

Table 2. Numbers N_{IQS} of coefficient vectors $\mathbf{h}^*$ that exactly satisfy the constraints (9)-(11) within the band of ±2 for the filter lengths $N = 8$ and $N = 16$ as functions of coefficient precision, 6, 8, 10, 12, 14, and 16 bits.

Coefficient precision (bits)	6	8	10	12	14	16
N_{IQS}, $N = 8$	15	14	15	15	14	15
N_{IQS}, $N = 16$	55086	54760	49164	54394	54760	49164

4. Characteristics of the truncated and ideally quantized coefficient filters

In Fig. 1 it can be seen that the predictor with conventionally quantized coefficients does not provide for exact prediction at zero frequency, and also the dc-gain of exact unity is lost, whereas the predictor with the ideally quantized coefficients possesses both quantities *exactly*, as it should, since it satisfies the constraints (3)-(5) *exactly*. It is to be noted that generally the deviation from the exact prediction step and dc-gain values, due to conventional coefficient quantization, gets larger as the filter length increases, but for practical applications longer filters may be needed to provide for lower noise gains.

The noise gains of the best ideally quantized coefficient filters within the search band are listed in Table 3. From Table 3 it can be seen that as the coefficient precision is increased, the noise gain of the

ideally quantized coefficient filter approaches the noise gain of the corresponding infinite precision coefficient filter, and that the noise gain loss can be regarded marginal with any of the ideally quantized coefficient precisions, 6, 8, …, 16 bits.

Table 3. Noise gains of the ideally quantized coefficient, $p = 1$, $M = 2$, polynomial FIR predictors of lengths $N = 8$ and $N = 16$ as functions of coefficient precision in bits. Also the noise gains of the same filters with infinite precision coefficients are mentioned.

Coefficient precision (bits)	*NG*, *N* = 8	*NG*, *N* = 16
6	1.9531250000	0.7324218750
8	1.9472656250	0.7304687500
10	1.9464721679	0.7303638458
12	1.9464287757	0.7303590774
14	1.9464285820	0.7303571626
16	1.9464285718	0.7303571444
Inf. precision	1.9464285714	0.7303571428

5. Conclusions

A new technique for perfect polynomial-predictive FIR digital filter coefficient quantization has been proposed. As it is demonstrated in this paper, the filter design constraints giving the filters their polynomial signal prediction properties can be exactly satisfied with low fixed-point coefficient precisions, and thus, the influence of the round-off errors is eliminated. For the second degree polynomial FIR predictors, used in this paper as an example, the conditions can be exactly satisfied with even as low as 6-bit coefficient precision, with still some degrees of freedom left to minimize the noise gain of the designed fixed-point coefficients filter. The proposed integer programming method for fixed-point filter design is well suited for all filter design tasks in which the design criteria can be formulated in a form of linear constraints on the filter coefficients.

Acknowledgment

The work of V. S. Dimitrov has been funded by the Technology Development Centre of Finland, Nokia Corporation, Sonera Ltd., Finland, and Omnitele Ltd., Finland. The work of J. M. A. Tanskanen has been partially funded by the parties mentioned above, and his work has also been supported by Jenny ja Antti Wihuri Foundation, Finland, Walter Ahlström Foundation, Finland, The Finnish Society of Electronics Engineers, and by Foundation of Technology, Finland.

References

[1] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications.* New York, NY, USA: Macmillan Publishing Company, 1992.

[2] P. Heinonen and Y. Neuvo, "FIR-median hybrid filters with predictive FIR substructures," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 36, pp. 892–899, June 1988.

[3] S. Väliviita and S. J. Ovaska, "Delayless recursive differentiator with efficient noise attenuation for control instrumentation," *Signal Processing*, vol. 69, pp. 267–280, Sept. 1998.

[4] P. T. Harju, T. I. Laakso, and S. J. Ovaska, "Applying IIR predictors on Rayleigh fading signal," *Signal Processing*, vol. 48, pp. 91–96, Jan. 1996.

[5] S. J. Ovaska, O. Vainio, and T. I. Laakso, "Design of predictive IIR filters via feedback extension of FIR forward predictors," *Proc. 38th Midwest Symposium on Circuits and Systems*, Rio de Janeiro, Brazil, 1995, pp. 370-375.

[6] D. Bertsekas, *Constrained Optimization and Lagrange Multipliers Methods.* New York, NY, USA: Academic Press, 1982.

[7] E. L. Johnson, *Integer programming, facets, subadditivity and duality for group and semigroup problems.* Philadelphia, PA, USA: SIAM, 1980.

[8] A. Schrijver, *Theory of Linear and Integer Programming.* New York, NY, USA: John Wiley, 1986.

[9] C. R. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity.* Englewood Cliffs, NJ, USA: Prentice Hall, 1982.

[10] G. Szekeres, "Multidimensional continued fraction algorithms", *Ann. Univ. Sci. Budapest Eotvos*, Sect. Math., vol. 13, pp. 113–140, 1970.

[11] H. R. P. Ferguson and R. W. Forcade, "Generalization of the Euclidean algorithm for real numbers to all dimensions higher than two," *Bull. AMS*, pp. 912–914, Nov. 1979.

[12] M. Clausen and A. Fortenbacher, "Efficient solution of linear Diophantine equations," *Journal of Symbolic Computation*, vol. 8, pp. 201–216, July/Aug. 1989.