DESIGN OF PRIMITIVE OPERATOR DIGITAL FILTERS USING GENETIC ALGORITHMS

David W. Redmill and David R. Bull
Image Communications Group, Centre for Communications Research,
University of Bristol, Bristol. BS8 1UB UK
Tel: [+44] 117 954 5203, Fax: [+44] 117 954 5206
Email: David.Redmill@Bristol.ac.uk Dave.Bull@Bristol.ac.uk

ABSTRACT

This paper considers the design of low complexity digital filters. Complexity is reduced by constraining the filters to have integer coefficients, which can be efficiently implemented using primitive operator directed graphs (PODG). Genetic Algorithms (GAs) are used in conjunction with a heuristic graph design algorithm, to provide a joint optimization of filter performance and complexity. The proposed technique is used to design 1D filters, 2D filters and perfect reconstruction filter banks.

1 INTRODUCTION

Digital filters are increasingly found in all areas of digital signal processing (DSP). For practical systems, it is often important that the digital filters should have low implementation cost and low power consumption, while operating at high data rates.

To achieve this goal, the filter coefficients are often constrained to be integers, with single extra (possibly floating point) multiplier on the filter output. The filter requires an integer multiplier for each coefficient. These integer multipliers can be efficiently implemented using a combination of additions, subtractions and power of two shifts, collectively referred to as primitive operators.

Various implementation strategies exist. The first is a binary implementation, in which each multiplier is expressed as a simple sum of power-of-two terms. However many values e.g. 15 can be more efficiently implemented using a combination of both additions and subtractions. Thus each multiplier can be implemented as a sum of signed power-of-two (SPT) terms. The third method is to implement the complete multiplier bank using a directed graph structure [1]. This method allows significant savings as a result of the reuse of intermediate partial sums. Figure 2 shows an example of a primitive operator directed graph for a 1D filter.

The resulting design problem thus becomes that of choosing a set of filter coefficients and a corresponding implementation which offers an effective compromise between implementation complexity and performance. There are two main strategies to achieve this.

The first strategy is to first design an appropriate integer coefficient filter, and then to find an efficient implementation. The integer coefficient filter can be designed either by rounding the coefficients from an effective floating point design, or by employing an optimization technique such as linear programming. The optimal SPT representation is easily found using the canonic signed digit (CSD) algorithm [2]. However, the optimal graph design is a more difficult problem, although various heuristic methods exist [1, 3], which give a close to optimal implementation. The problem with this approach, is that it achieves compromise which tends to favor higher performance and higher complexity solutions. The reason for this is due to the choice of filter being made without reference to its complexity.

An alternative method is to use a complexity constrained approach, where a specific implementation is selected with an appropriate complexity constraint. The space of possible parameters is then searched in order to find a filter with optimum performance given the constraints. For example, the filter coefficients can be constrained to comprise a sum of a fixed number (often 2) of SPT terms. The values of each of these SPT terms is then optimized [4, 5, 6, 7, 8, 9]. A relaxation of this constraint was used in [10, 11], where only the total number of SPT terms is constrained and their distribution among the coefficients is varied. Although this approach has been found to work well with an SPT implementation, it is not well suited for use with directed graphs, due to the epistatic nature of the problem. A small change in the graph will typically lead to a large change in filter characteristics. This makes it very difficult to find an optimal solution.

2 PROPOSED METHOD

In this paper, we propose a multi-objective optimization approach. The optimization problem is stated as that of finding a set of integer coefficients which provides an effective compromise between filter order, complexity and performance. Complexity of any candidate solution is estimated by by employing an appropriate design algorithm such as CSD [2] or a directed graph [3]. Although these design algorithms represent an increase of complexity for evaluating candidate solutions, this increase is not significant compared to the complexity of evaluating the frequency response.

Since the parameter space is defined in terms of the integer coefficients, it is easy to move between filters with similar performance, thus aiding the optimization. However, the complexity is still highly epistatic. The optimization problem is easily seen to be both discrete, multi-objective and multi-modal, and thus requires a sophisticated optimization tool. In this paper we have employed a Genetic Algorithm. Genetic algorithms [12] are chosen because they are known to be effective for discrete, multi-objective, multi-modal problems. In the proposed GA, individual solutions are represented as the set of corresponding integer filter coefficients, and the mutation and crossover operators are appropriately modified. An additional mutation operator is also intro-

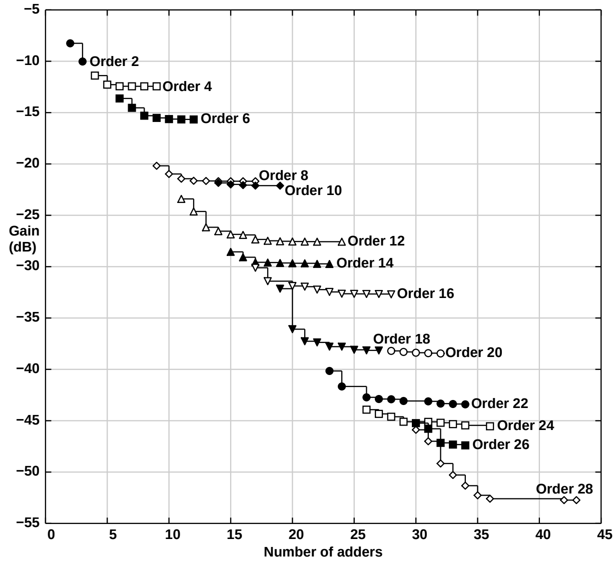

Figure 1: Tradeoffs between complexity, order and attenuation for an example low pass specification.

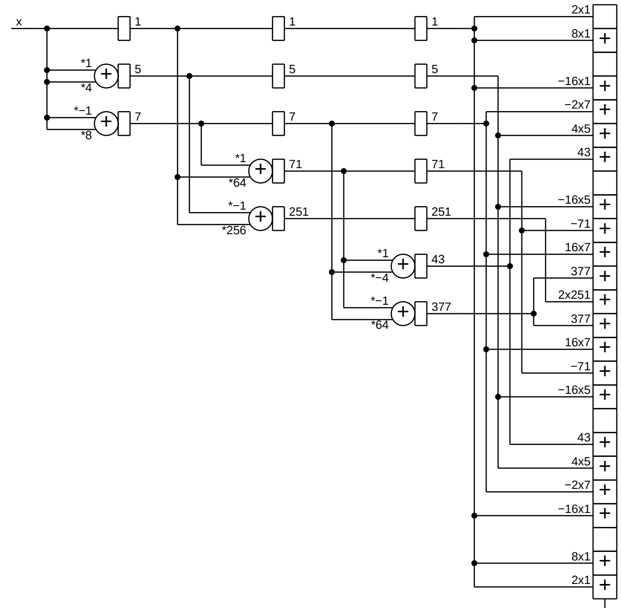

Figure 2: Graph structure for implementation of 24th order filter with 26 adders.

duced, which allows all of the coefficients to be scaled and rounded. This allows the algorithm to effectively explore significantly different regions of the search space, by jumping to filters with similar performance, but significantly different coefficients. The GA produces a pareto-optimal or non dominated set (NDS) of solutions which represent the various performance/complexity/filter-order compromises. Thus the net result of the process is not one solution, but a set of designs representing differing trade-offs between complexity and performance.

3 EXAMPLE DESIGNS

3.1 1D Low-Pass Filters

In order to demonstrate the effectiveness of this approach we shall consider an example minimax design problem. The filter is specified as being a linear-phase, low-pass, 1D-FIR filter with a normalized transition band between 0.15 and 0.25. This example has been chosen to allow comparison with previous work [4, 5, 6].

Figure 1 shows the non dominated set (NDS) obtained for this design. This figure consists of a separate non-increasing curve for each different filter order. Realizable filters lie on and above/right of the curves, while an optimal compromise would be towards the lower left. Ignoring complexity, we see that optimal integer coefficient filters lie at the lower right hand end of these curves. It is now clear that these are unlikely to be as useful as other slightly lower performance filters, which offer a significant reduction in complexity. Alternatively, for a given complexity, significant performance can often be made by using a higher order filter. Finally, we see that there is an approximately linear trade-off between complexity and attenuation (measured in dBs). The gradient of this trade-off varies according to the design specifications.

For comparison, we shall consider the 24th order example of [5], using 35 adders, which has a an attenuation of -43.8 dB. From figure 1, it can be seen that a 24th order filter with slightly improved attenuation (-43.92dB) can be achieved with only 26 adders. Figure 2 shows the graph based structure used to implement this filter, which consists of 6 adders to generate the values of 5, 7, 71, 251, 43 and 377. These values are then scaled by signed powers-of-two and summed in the shift register. Note that since 4 of the coefficients are zero, only 20 instead of 24 adders are used in the shift register. The rectangular boxes represent possible locations for pipelining registers.

From figure 1, we can see that another 24th order filter with 34 adders and an attenuation of -45.45 dB can be designed. Note that this is very close to the optimal (24th order) floating point design (obtained using the Parks and McClellan algorithm) which has an attenuation of -46 dB. If we relax the constraint on the filter order, we can design a 28th order filter with 35 adders and an attenuation of -52.26 dB. Thus, for the same complexity we can achieve an improvement of 8.4 dB over the SPT design of [5].

Note that the proposed method does not restrict our consideration to designs using directed graphs. By replacing the graph design algorithm with a CSD algorithm, and repeating the experimentation, we can gain a similar NDS for designs using CSD. Figure 3 shows a comparison of the two methods, from which it can be seen that for more complex filters, the directed graph method yields significantly better results.

3.2 1D Band-Pass Filters

To demonstrate the flexibility of this approach, a linear-phase band-pass filter, with transition bands $0.15 - 0.25$ and $0.35 - 0.45$, has been designed. Figure 4 shows the resulting NDS. A 28th order filter with an attenuation of -50.96 dB can be designed with 31 adders. To design a comparable (complexity) filter with 2 SPT terms per coefficient (as in [6]) we must consider a maximum of 20th order[1], which gives an attenuation of about -36 dB (from [6]). Note that even an optimal floating point 20th order filter only achieves -39dB.

3.3 2D Low-pass Filters

The same method can be used to design 2D filters. Figure 5 shows results for a 2-D circularly symmetric, low-pass filter, with an annular transition band between 0.15 and

[1] A 20th order linear phase filter requires up to 20 adders in the shift register and 11 coefficients with up to one adder each giving a maximum of 31 adders

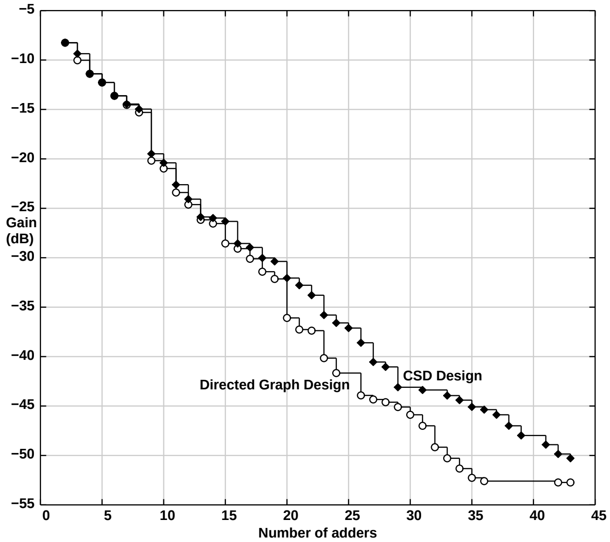

Figure 3: Comparison between CSD and Directed Graph Design Methods.

0.35. These results can be compared to previous work of [7, 8, 9, 10, 11], and are seen to exhibit improved complexity and/or performance. For example, in [10], a 53 adder, 7x7 filter with a maximum ripple of -28.8dB was designed. Figure 5 shows that similar performance can be achieved with only 49 adders. The improvement obtained by using larger filters becomes less significant in two dimensions, since an increase in size/filter order implies a much larger increase in complexity, due to the larger 2D shift register structure.

3.4 Perfect Reconstruction Filter Banks

As a final example, we shall show how these techniques can be used to design linear phase perfect reconstruction filter banks. Figure 6 shows the typical structure for a 2-band filter bank which is commonly used to build sub-band and wavelet transforms. Although the proposed method could be used to design the individual filters directly, it is in practice very difficult to obtain perfect reconstruction (see [13]). This motivates the use of a lattice architecture. Various lattice architectures exist, including those of [14, 15, 16, 17]. These offer a convenient representation which is both efficient and ensures perfect reconstruction for arbitrary parameters.

In this paper we will use a design technique similar to that proposed in [18], which is based on the use of a variable transformation in conjunction with the lattice of [17]. Figure 8 shows a two stage lattice structure. Perfect reconstruction is ensured for arbitrary functions $p(z)$ and $q(z)$. Although the proposed method could be used to design $p(z)$ and $q(z)$ directly, it is convenient to write these in terms of a single transformation $A(z) = -zp(z^2) = \frac{1}{2}z^{-1}q(z^2)$ under the constraint $A(z) = -A(-z)$ (only odd terms). The corresponding filters can be written as:

$$H_0(z) = 1 + \frac{1}{2}A(z) - \frac{1}{2}A^2(z) \qquad G_0(z) = 1 + A(z)$$

Linear phase is achieved by imposing a symmetry constraint $A(z) = A(z^{-1})$. Zeros at $z = -1$ (which encourage good stop-band attenuation and regularity) can be achieved by restricting $A(1) = 1$. Since the implementation of these filters is achieved using the lattice shown in figure 8, it can be seen that the analysis filter bank requires 2 transformations $p(z), q(z)$ and a further 2 adders. However, it should be noted that these are at the down-sampled rate.

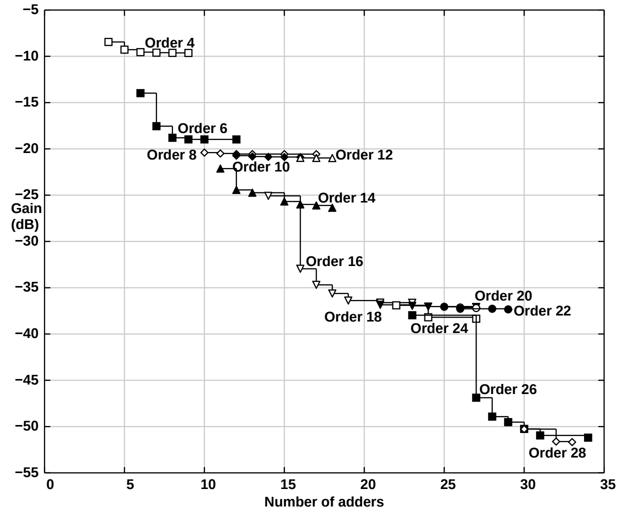

Figure 4: Tradeoffs between complexity, order and attenuation for an example band pass filter specification.

Figure 7 shows results achieved for the design of a pair of filters with a transition band between 0.2 and 0.3. These can be compared with example 2 from [13] in which an approximately perfect reconstruction pair using 88 adders was designed with a maximum ripple of about -30dB for transition band 0.l8 to 0.32. Figure 7 shows that perfect reconstruction filters can be designed with half the complexity (44 adders per pair of samples), and a maximum ripple of -40dB for a tighter transition band specification. The results also show improved performance compared to the previously presented design of [18], in which a filter pair using 48 adders (per pair of samples) was designed with a maximum ripple of -35dB.

4 CONCLUSIONS

This paper has examined the design of low complexity FIR filters and filter banks using a primitive operator directed graph (PODG) representation. Genetic algorithms have been used to perform a joint optimization of both filter performance and complexity.

The result is a flexible design tool which, given a filter specification, provides the user with a variety of solutions, which offer various compromises between performance, complexity and filter order. In this way the designer can easily choose a filter best suited for the given application. The approach has been demonstrated for a variety of problems including 1D filters, 2D filters and perfect reconstruction filter banks. Significant improvements are obtained compared to previous methods. These are due to a number of factors including: t5he use of a directed graph implementation, a truly multi-objective optimization strategy and the use of heuristic implementation algorithms which help to maintain a simpler search space. The proposed method is also well suited to other problems such as IIR filters, cascaded filters etc. The method is also not restricted to minimax designs, but can be used with any other objective performance measure or measures.

References

[1] D. R. Bull and D. H. Horrocks. Primitive operator digital filters. *IEE-Proceedings-G*, 138:401–412, 1991.

[2] A. Avizienis. Signed-digit number representation for fast parallel arithmetic. *IRE Trans. Electon. Comput.*, 10:389–400, 1961.

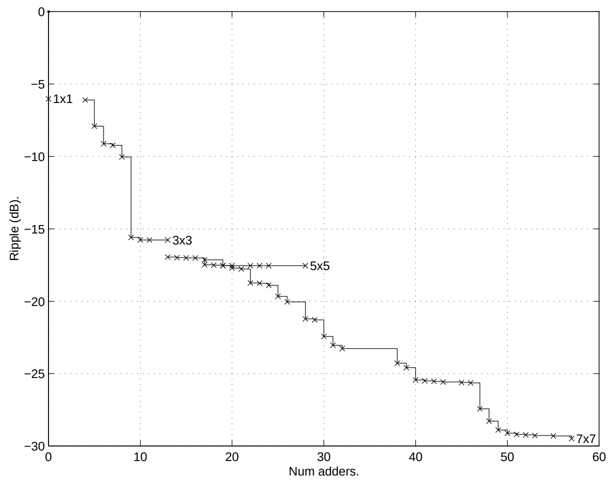

Figure 5: Example design for 2-D circularly symmetric low-pass filter.

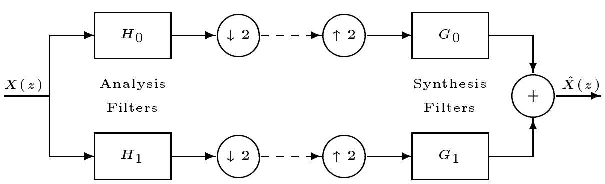

Figure 6: 2-band filter bank.

[3] A. G. Dempster and M. D. Macleod. Use of minimum-adder multiplier blocks in FIR digital filters. *IEEE Trans CS-II: Analog and Digital Signal Processing*, 42:569–577, 1995.

[4] N. Benvenuto, M. Marchesi, and A. Uncini. Applications of simulated annealing for the design of special digital filters. *IEEE Trans SP*, 40:323–332, 1992.

[5] H. Samueli. An improved search algorithm for the design of multiplierless FIR filters with powers-of-two coefficients. *IEEE Trans CS*, 36:1044–1047, 1989.

[6] Q. Zhao and Y. Tadokoro. A simple design of FIR filters with powers-of-two coefficients. *IEEE Trans CS*, 35:566–570, 1988.

[7] L. Banzato. A design method for 2-D multiplierless FIR digital filters for video applications. *IEEE Trans CSVT*, 2:273–284, 1992.

[8] P. Siohan and A. Benlismane. Finite precision design of optimal linear phase 2-D FIR digital filters. *IEEE Trans CS*, 36:11–21, 1989.

[9] S. Sriranganathan, D. R. Bull, and D. W. Redmill. The optimisation of multiplierless two dimensional digital filters. In *SPIE VCIP*, 1996.

[10] S. Sriranganathan, D. R. Bull, and D. W. Redmill. Low complexity two-dimensional digital filters using unconstrained SPT term allocation. In *Proc IEEE ISCAS 96, Atlanta, USA.*, pages 762–765, 1996.

[11] D. R. Bull and D. W. Redmill. Optimisation of image coding algorithms and architectures using genetic algorithms. *IEEE Trans Industrial Electronics*, 43:549–557, 1996.

[12] David E. Goldberg. *Genetic Algorithms in Search and Optimization*. Addison-Wesley, 1988.

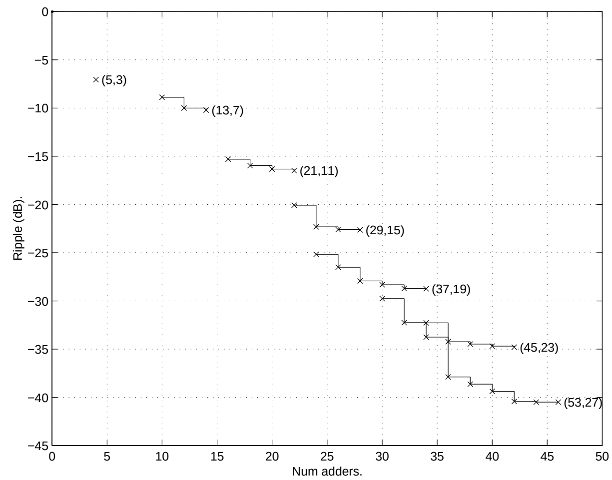

Figure 7: Example design for a perfect reconstruction filter bank.

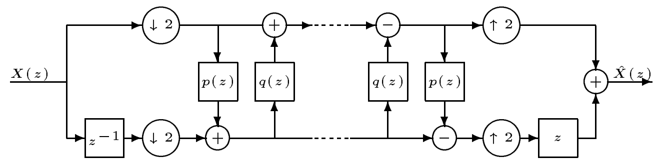

Figure 8: Two stage system.

[13] B. R. Horng, H. Samueli, and A. N. Willson. The design of low-complexity linear-phase FIR filter banks using powers-of-two coefficients with an application to subband image coding. *IEEE Trans CSVT*, 1:318–323, 1991.

[14] P. P. Vaidyanathan and P.Q. Hoang. Lattice structures for optimal design and robust implementation of two-channel perfect-reconstruction QMF filter banks. *IEEE Trans ASSP*, 36:81–94, 1988.

[15] T. Nguyen and P. P. Vaidyanathan. Two-channel perfect-reconstruction FIR QMF structures which yield linear-phase analysis and synthesis filters. *IEEE Trans ASSP*, 37:676–690, 1989.

[16] M. Vetterli and D. LeGall. Perfect reconstruction FIR filter banks: Some properties and factorizations. *IEEE Trans ASSP*, 37:1057–1071, 1989.

[17] F. A. M. L. Bruekers and A. W. M. van den Enden. New networks for perfect inversion and perfect reconstruction. *IEEE JSAC*, 10:130–137, 1993.

[18] D. W. Redmill D. R. Bull and R. R. Martin. Design of multiplier free linear phase perfect reconstruction filter banks using transformations and genetic algorithms. In *IEE IPA-97*, pages 766–770, 1997.