OPEN ACCESS

University of BRISTOL

Link to published version (if available):
10.1109/ISCAS.1998.694524

Link to publication record in Explore Bristol Research
PDF-document

University of Bristol - Explore Bristol Research

General rights

AUTOMATED DESIGN OF LOW COMPLEXITY FIR FILTERS

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 5195, Fax: [+44] 117 925 5265
Email: David.Redmill@Bristol.ac.uk Dave.Bull@Bristol.ac.uk

ABSTRACT

This paper considers the design of low complexity FIR 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 solution set which represents different compromises between performance, complexity and filter order.

Example results are presented for both one and two dimensional filters, and are shown to provide both superior performance and complexity, compared to previous methods. The main benefits result from the use of a joint optimization, rather than a separable 2-stage approach. The use of a PODG representation is shown to provide significant improvements over a canonic signed digit (CSD) or signed power-of-two (SPT) representation.

1. INTRODUCTION

Finite impulse response (FIR) digital filters have many applications within a wide range of digital signal processing algorithms. Some of these applications (e.g. video processing) require the filter to operate at very high data rates, which gives rise to the need for dedicated high speed application specific integrated circuits (ASICs).

Many filter design techniques (e.g. Parks and McClellan) yield filters with floating point coefficients, which in turn implies the need for high precision multipliers, high hardware complexity and high manufacturing cost. In order to reduce cost, it is desirable to design filters which have similar performance for a significant reduction in complexity. One method for achieving this is to round the floating point coefficients to finite precision integers.

The integer coefficients can then be represented as the sum of a few signed power-of-two (SPT) terms. Thus floating point multipliers are replaced by a few integer adders

---

and power-of-two shifting operations. The minimum number of SPT terms required for any given coefficient can be easily determined using a canonic signed digit (CSD) algorithm [1]. The filter design problem can now be stated as that of finding an optimum set of coefficients and a corresponding SPT representation which optimize both performance and complexity.

Further improvements can be made by implementing the set of integer coefficient multipliers, using a primitive operator directed graph (PODG) representation [2]. The graph consists of, a set of two-input nodes representing adders, and edges with signed power-of-two (SPT) scaling values. For example, the set coefficients 9 and 13 can be represented as $9x = 8x+x, 13x = (9x)+4x$. The task of finding a minimum adder graph for a given set of coefficients is a difficult problem. Various heuristic approaches have been proposed [2, 3], which give optimal graphs for most practical examples.[1]

The above implementation methods lead to the following design strategy. Firstly an integer coefficient filter is designed, either by rounding a floating point design and/or using a optimization technique such as linear programming. Secondly the filter can be implemented using either a CSD or PODG representation. This method yields designs which favor increased performance rather than decreased complexity. For the results shown in figures 1, 4 and 5, this method would yield filters which lie to the right hand end of the various curves.

A second design strategy, is to first constrain the implementation complexity, and then optimize the various parameters to achieve optimal performance. For example various authors [4, 5, 6, 7, 8, 9, 10] have designed filters by constraining them to consists of a finite number of coefficients, each consisting of a finite number of SPT terms. In [11, 12], the conditions were relaxed by constraining only the total number of SPT terms and allowing their distribu-

---

[1]For the results presented in figure 1, at least 80% of the filters have a graph which is known to be optimal in the minimum adder sense. The remaining 20% lie towards the less important right hand side of the figure, and are known to be very close to optimal (within 1 or 2 adders).
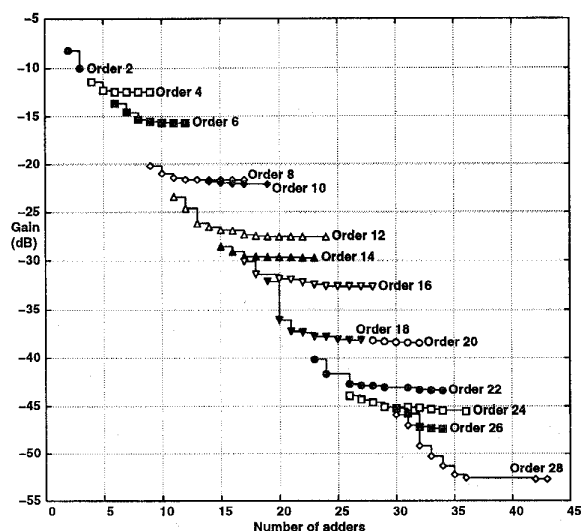
Figure 1: Tradeoffs between complexity, order and attenuation for an example low pass specification.

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

tion to vary. This strategy is more difficult for filters using a PODG representation [13]. The difficulties are due to the discrete, non-uniform, and non-linear nature of the search space.

2. PROPOSED METHOD

In this paper we propose a joint optimization of both complexity and performance. To achieve this we represent each candidate filter in the search space as a set of integer coefficients. Thus for each candidate solution we need to evaluate both the filter performance, and the implementation complexity. Since the graph design method of [3], is relatively quick compared to evaluating the filter performance (in the frequency domain), the approach doesn't require excessive computational effort compared to previous methods (e.g. [11, 12]).

In this paper we use a Genetic Algorithm (GA) to perform the optimization, since GAs have been proved successful, on a wide variety of discrete, non-uniform, multi-objective optimization problems. GAs use a population of candidate solutions to evolve towards an effective solution. Each candidate solution (individual) is described by a chromosome. The fitness of each individual is measured and used to control which individuals are selected to form the next generation. Between generations, various genetic operators (including crossover and mutation) are applied in order to search new regions of the solution space. GAs have been successfully used for low complexity filter design problems including, cascaded FIR filters [14], IIR filters [15] and SPT designs [11, 12].
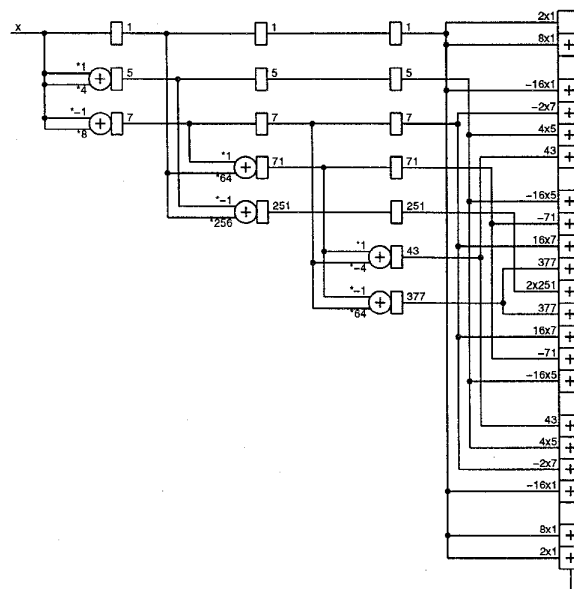
In this paper, the GA is used to provide a non dominated set (NDS) of solutions which represent various compromises between complexity and performance and filter order. Due to the nature of the search space, it is natural to use a GA with integer valued genes (one per coefficient) rather than the more usual binary coding. This approach maintains a close relationship between the chromosome structure, and the problem space. This in turn allows us to custom design the genetic operators to suit the particular problem. The operators used consist of:

- Uniform crossover.

- Mutation of coefficients by randomly adding a signed power-of-two (SPT) value.

- Mutation of an entire chromosome by scaling and rounding all the coefficients.

The last form of mutation is particularly useful, since it allows the GA to easily search filters with similar performance, but significantly different coefficients.

3. RESULTS

To demonstrate the effectiveness of the proposed techniques, we have used them to design odd length linear phase filters, using a minimax criterion.

The first example is a 1D low-pass filter with normalized pass and stop bands of $0 - 0.15$ and $0.25 - 0.5$ respectively. This example was chosen to allow comparison with previous work [4, 5, 6].
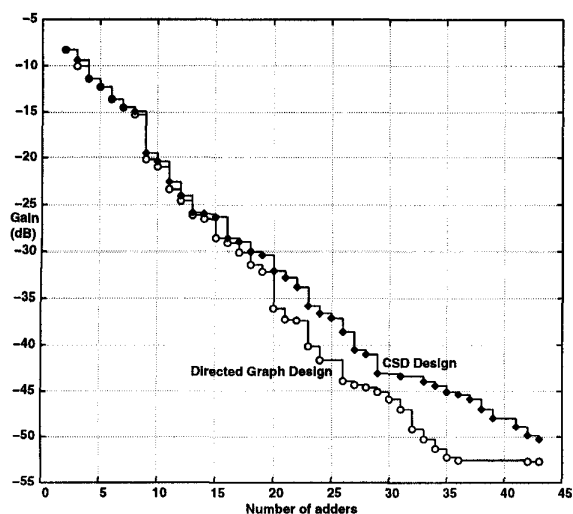
V-430

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

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

Figure 1 shows the non dominated set (NDS) obtained for this design. This figure consists of a separate 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 of the plot. It can be seen, that the optimal integer filters (no constraint on complexity) lie at the bottom right hand end of these curves, and require much higher complexity than other slightly lower performance filters. Thus, for a given filter order, the joint optimization allows filters with significantly lower complexity and only slightly lower performance, than an optimal integer designs. Secondly, we can see that significant performance improvements (at a given complexity) can often be made by using a higher order filter. Thirdly we note that some filter orders (e.g. 10 and 20 for this example) are of little use. The reason for this, is that the corresponding coefficients are usually very small in the optimum minimax designs, and thus have little effect. These coefficients can be usefully approximated to zero. Finally, we note 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, and 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. The graph 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
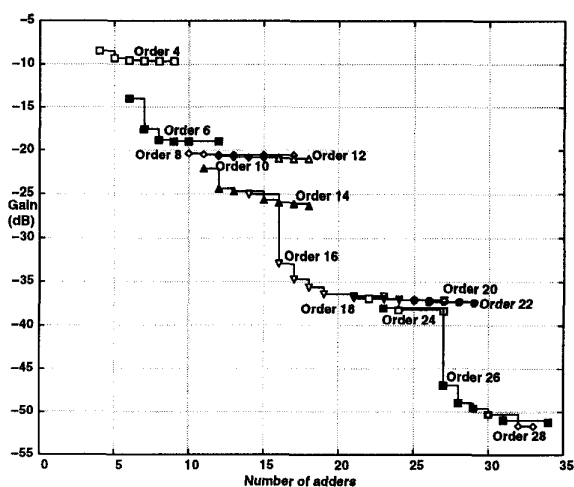
the shift register. The rectangular boxes represent possible locations for pipelining registers.

Figure 1 also shows 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 CSD design of [5].

Note that the use of the GA 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.

To demonstrate the flexibility of this approach, a bandpass 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[2], which gives an attenuation of about -36 dB (from [6]). Note that even an optimal floating point 20th order filter only achieves -39dB.

The method can also be used to design two-dimensional filters. Figure 5 shows results for the design of 2D linear-

---

[2]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.
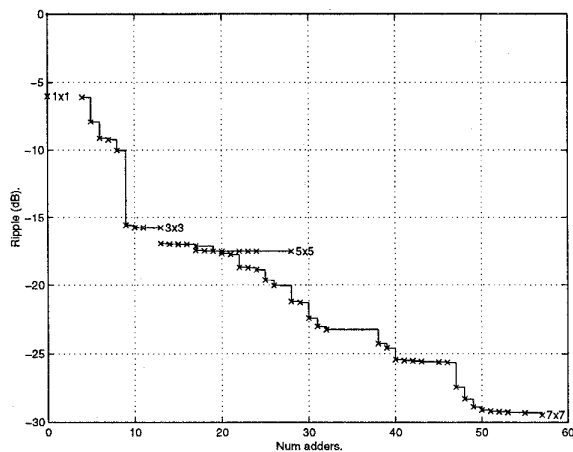
V-431

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

phase circular symmetric low pass filters with an annular transition band between 0.15 and 0.35. These results can be compared with those of [7, 8, 9, 10, 11, 12], and demonstrate filters with both superior complexity and performance for sizes of 5x5, 7x7 and 9x9. For example, in [11, 12], a 53 adder 7x7 filter with a maximum ripple of -28.8dB, was presented. Figure 5 shows that a slightly better performance (-29.0dB) can be achieved with only 48 adders. The improvement obtained by using larger filters becomes less significant in two dimensions, since an increase in size implies a much larger increase in complexity, due to the larger shift register structure.

4. CONCLUSIONS

This paper has examined the design of low complexity FIR filters using a primitive operator directed graph (PODG) representation. Genetic algorithms have been used to perform a joint optimization of both filter performance and complexity. This method provides a range of filters covering the various trade-offs between performance, complexity and filter order.

Example minimax designs have been presented for both one and two dimensional linear phase filters, and shown to provide superior results to previous methods. The improvements are due to both the use of a PODG representation, and the joint optimization strategy.

5. REFERENCES

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

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

[3] A. G. Dempster and M. D. Macleod. Use of minimum-adder multiplier blocks in FIR digital filters. *IEEE Transactions on Circuits and Systems-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 Transactions on Signal Processing*, 40:323–332, 1992.

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

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

[7] L. Banzato. A design method for 2-D multiplierless FIR digital filters for video applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 2:273–284, 1992.

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

[9] S. Sriranganathan, D. R. Bull, and D. W. Redmill. Design of 2-D multiplierless FIR filters using genetic algorithms. In *First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications*, 1995 Sheffield UK. 12-14 September 1995.

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

[11] 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.

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

[13] D. R. Bull and A. Aladjidi. The optimization of multiplier-free directed graphs: An approach using genetic algorithms. In *Proceedings IEEE International Symposium on Circuits and Systems*, pages 456–459, 1994.

[14] G. Wade, A. Roberts, and G. Williams. Multiplier-less fir filter design using a genetic algorithm. *IEE proceedings, Vision, Image and Signal Processing*, 81:175–180, 1994.

[15] P. B. Wilson and M. D. Macleod. Low implementation cost IIR filter design using genetic algorithms. In *Natural Algorithms in Signal Processing Workshop. Chelmsford, Essex: IEE, Nov 1993*, pages 4/1–4/8.