A GENETIC ALGORITHM FOR THE DESIGN OF FINITE WORD LENGTH ARBITRARY RESPONSE CASCADED IIR DIGITAL FILTERS

T.Arslan and D.H.Horrocks
THE UNIVERSITY OF WALES, UK

ABSTRACT

This paper describes a Genetic Algorithm (GA) for the design of optimal finite word length (FWL) infinite impulse response filters with arbitrary response functions using a cascade of second order sections. Such structures could be used for the implementation of filters for an unrestricted range of responses and are usually characterised by having low sensitivity to variations in word length and simple stability tests. A distinct feature of the GA is that it allows the single-step-design from a specified filter template hence avoiding additional errors being introduced as is the case with the two-stage-design approach. Depending on the specified template, the GA could be used to design filters with arbitrary response functions including those of the main classical filters with considerably reduced coefficient wordlengths. The paper commences by highlighting FWL effects in digital filters and reviewing existing attempts in using GAs in digital filter design. Next, the paper describes the implementation technique adopted here and reports on the results obtained using examples of 8th and 10th order filters .

1. INTRODUCTION

1.1 FWL EFFECTS IN DIGITAL FILTERS

Hardware implementation of digital filters impose a number of practical constraints since each of the theoretical infinite precision coefficients used to describe the filter must be represented by a finite number of bits limited by the size of the physical registers in hardware. Operations such as rounding and truncation of the coefficients to the restricted number of bits may cause the transfer function of the designed filter to deviate from the original specifications of the filter. A number of papers in the literature [1-4] attempt to determine an optimal set of finite word length coefficients derived from the original infinite precision coefficients. The techniques employed by the above usually utilise a search method with a probabilistic variation of each FWL coefficient with the aim of minimising the error between the desired frequency response and the response of the filter described by the coefficients. An increase in the number of coefficients will lead to significant increase in computation time making the use of the above techniques very expensive [5].

Infinite impulse response (IIR) filters are an important class of digital filters which are significantly important in cases where sharp cut-off and high throughput are desirable. Such filters, especially those based on the elliptic characteristics require fewer coefficients than finite impulse response (FIR) filters. IIR filters can be realised in one of several forms, including direct, cascade, or parallel forms [6]. The direct form is simply a full order single-stage representation of the IIR transfer function. In the cascade form, the transfer function of the IIR filter is factored and expressed as the product of second-order sections. In the parallel form, the transfer function is expanded, using partial fractions, as the sum of second-order sections. The parallel and cascade structures are the most widely used for IIR because they lead to simpler filtering algorithms and are generally far less sensitive than the direct form, to the effects of implementation using a finite number of bits. Other low sensitivity structures for the implementation of IIR filters are the wave digital filter structures [7,8] which have some passivity properties [9,10,11] allowing them to have low sensitivity with shorter word lengths. According to Dempster et. al. for a number of filter designs such as the ones in [12] the cascade provides a more superior structure to both wave and parallel. Hence, in this work cascade structures are used due to their generality [12] in addition to the relative simplicity of performing stability checks on them.

1.2 GAs AND DIGITAL FILTER DESIGN

In the past few years GAs [13,14] have been applied to a number of design issues regarding both digital and analogue filters [15]. Here, we will concentrate on the digital case only. Suckley [16] has used GAs for the design of low-pass FIR filters providing structures of

Genetic Algorithms in Engineering Systems: Innovations and Applications
12-14 September 1995, Conference Publication No. 414, © IEE, 1995

near minimal computational complexity. The designs produced by the GA are compared by evaluating the frequency over the bands over which the filter is specified [17]. In [18], this work is further extended to the design of medium multiplierless FIR filters constructed from a cascade of linear primitive sections. Dexiang et. al. [19] use a parallel GA to design optimal FWL FIR filters. The parallel GA is implemented using a hypercube, in which populations representing sets of FWL coefficients are allocated to each node. The evolution process is guided by a strategy of minimising the frequency response.

Recently, GAs have also been applied to IIR design problems. Wilson et. al. [20], consider the design of a number of IIR filter examples using cascaded second order sections. A standard GA has been applied, without regard to computational cost, in order to find a compromise between response error and adder cost. The number of coefficient bits considered is in the range 7-8. In addition, this work requires the evaluation of the ideal frequency response at the frequencies of interest, hence a multiple-stage design is followed. Furthermore, stability is guaranteed by analysing the genes and identifying root positions. If a gene describes a position of a root outside the unit circle contradicting stability or minimum phase constraints, then the root is moved by multiplication with an all pass filter (this stage is followed by quantizing the coefficients). This step is a restriction to the solutions provided by the GA. In [21,22] and [23] the use of GAs has been extended to adaptive IIR filters and lattice wave structures. In both cases success is reported on the example filters on which the GAs have been used. All of the above indicate the success of the GAs as powerful tools in digital filter design, however, the full power of the GAs in this area (and specifically in the case of IIR filters) has not yet been explored.

This paper describes a genetic algorithm for the design of optimal finite word length infinite impulse response filters with arbitrary response functions using a cascade of second order sections. The GA provides a single-stage design in that the quantised coefficients are produced directly from the frequency response template. This is likely to produce fitter designs than the conventional two-stage approach (as in [20] for example) in which the design is produced via an infinite-precession ideal polynomial transfer function which introduces further stage of approximation. In addition, it offers generality since filters could be designed which satisfy arbitrary template specifications including those which could be based on classical filter templates. The GA proves successful in satisfying severely restricted templates in relatively small number of coefficient bits. Furthermore, for the present application they also have the desirable property that stability can be guaranteed by imposing a simple coefficient constraint.

2. THE GENETIC ALGORITHM

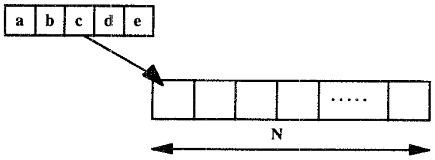
A conventional basic form of GA [13,14] was found to perform well in this work. The rest of this section describes the main tasks involved in the development of the GA and the necessary modifications for its use for the design of arbitrary response IIR filters.

REPRESENTATION METHOD

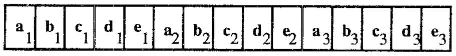
A typical second order section of the form :

$$\frac{a + bz^{-1} + cz^{-2}}{1 + dz^{-1} + ez^{-2}},$$

where a,b,c,d, and e are the multiplier coefficients. In the GA genes it can be represented as a string comprising of sequences of N bit binary numbers (N is the filter wordlength). Each of the binary numbers is associated with one of the multiplier coefficients. The representation of the above second order section is illustrated in figure (1a). A filter structure consisting of M cascaded stages will be formed by concatenating M strings each representing one of the cascade stages. A filter of three cascaded stages (M=3) is illustrated in figure (1b).



(a) A single cascade structure



a_i b_i c_i d_i e_i : Coefficients of the ith cascade stage

(b) A three-stage cascade structure

Figure 1 : Representation scheme for cascaded second order structures

THE INITIAL POPULATION

In our implementation the initial population is derived from a set of infinite precision coefficients which are produced using a set of appropriate routines within the Matlab signal processing environment [24]. Next, the above coefficients are rounded to the nearest finite word length number and then transformed to a finite range of values using a random process. The addition of a uniformly selected random number in the range [-3,+3] was found to be satisfactory in order to breed promising solutions through successive generations to fully correct designs (it could clearly be seen that such designs could not be obtained by simple rounding or truncation). In addition, a population size of a hundred was found to provide sufficient genetic diversity.

GENETIC OPERATORS

The genetic algorithm evolves from generation to generation by firstly selecting candidate parent chromosomes using *roulette wheel* selection. This consists of a weighted probability scheme with selection according to the fitness of each chromosome, such that fitter chromosomes have greater likelihood of being selected as parents. Mutation is performed by randomly selecting a member of the current population and adding/subtracting a randomly selected integer. Crossover is employed at a rate of 90% with mutation being carried out at 4%.

FITNESS FUNCTION

Fitness is evaluated in the normalised frequency range [0,1] over which a template is specified for a given filter. The template is specified in terms of minimum and maximum limits (frequency response values in decibels) for a grid of frequency values spanning the above range. For each evaluation of frequency response, a template violation is checked for. If a violation is detected then an error value is calculated (as the deviation from the minimum/maximum limit being violated) and later squared. This is repeated for all the frequencies allowed by the grid. Finally a total error is evaluated as specified by the following:

$$\Delta = \sum_{i=0}^{1} W_i (H(f_i) - T_{\min/\max})^2$$

Where,

$H(fi)$=The frequency response at frequency fi

Tmin/max= The maximum/minimum limit violated by $H(fi)$

Wi =Error weighting function

The overall fitness is evaluated as the reciprocal of Δ. Our work indicated that grid increments of $1/fs$ (fs is the sampling frequency) provided adequate compromise between computation speed and design accuracy. In the results to be given in the next section, a uniform error weighting is assumed, $Wi = 1$.

BASIC STABILITY CHECKS

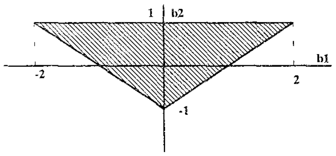


Figure 2 Stability triangle

The first step in fitness evaluation is to check for the stability of the chromosome. For each second order polynomial of the form $(1+b_1z^{-1}+b_2z^{-2})$ in the denominator, the coefficients are checked to be within the region defined by the stability triangle [6] indicated in the figure (2). The chromosomes failing the above check are awarded zero fitness value.

3. RESULTS

The GA was used to design a large number of filters. The filter designs investigated so far range from 4 up to 12 in their order. The GA was made to search for designs satisfying finite wordlength constraints of 16 down to 4 bits. Typically, the GA evolves for 10-100 generations for 'good' designs to emerge. The number of generations was governed by the difficulty of the design problem. Difficult problems include high order filters with highly restricted templates or small specified wordlengths. In addition, the number of good designes in a generation generally increased with the evolution process with the final few populations including 20-30% designs that fully satisfied the design criteria. Two GA designed filters are used to demonstrate our results. In the first, a 10th order arbitrary response filter is designed by providing the GA with a 1 db wide template. The GA converged to the stage where approximately 20% of the population were fully correct designs in 30-35 generations. A typical GA designed response (magnitude (db) vs. normalised frequency) is illustrated in figure (3a) with only 6 bits being required for the representation of the filter coefficients. Figure (3b) illustrates the difference between the infinite coefficient filter response and its corresponding GA designed filter response within a *tolerance band*, a region of allowed designs which is bound by the template. Similar results were obtained by increasing the template band to 2 - 5 dbs.

In the second design example the GA is used to design a filter based upon an 8th order elliptic template with a passband range of [0,0.6] and stopband region of [0.7,1] with attenuations of 1db and 80db respectively. The filter was designed with only 4 bits wordlength. After 30-35 generations approximately 10% of the population included fully correct designs. Two of these designs are illustrated in figure (4). The complete responses and the passband response are illustrated in figures (4a) and (4b) respectively. Both the speed of convergence of the GA and the number of good solutions in the final populations significantly increase by slight relaxation in the template specifications.

4. CONCLUSIONS

In this paper we describe a technique for the design of arbitrary response finite word length cascaded IIR filters. The technique uses a genetic algorithm in order to search a very large and complex search space of filter structures. A feature of this GA is that single-step design can be carried out directly on the specified response without the need to find intermediate approximations. Our results indicate that using direct cascaded sections the GA can implement an unrestricted number of responses even with severely restricted specifications. Our results also illustrate that with a slight relaxation in filter specifications the designs produced could be significantly reduced in complexity and could be implemented with smaller wordlengths even with higher order filters.

5. REFERENCES

[1] Kodek D.M.:"Design of optimal finite wordlength FIR filter using integer programming techniques", IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-28, pp. 304-308, 1980.

[2] Diethorn E.J.:"Finite wordlength FIR digital filter design using simulated annealing", ISCAS Proceedings, pp.217-220, 1986.

[3] Nielsen J.J.:"Design of linear-phase direct form FIR digital filters with quantized coefficients using error spectrum shaping techniques", IEEE Trans. on Accoustics, Speech, and Signal Processing, vol. 37, pp. 1020-1026, 1989.

[4] Nakayama K.: "A discrete optimization method for high-order FIR filters with finite wordlength coefficients", EEE Trans. on Accoustics, Speech, and Signal Processing, vol. 35, pp. 1215-1217, 1987.

[5] Chan S.K. and Rabiner L.R.: "Analysis of quantisation errors in the direct form for FIR filters", IEEE Trans. Audio Electroacoust., vol. 212, pp.354-366,1973.

[6] Ifeachor E.C. and Jervis, B.W: "Digital signal processing a practical approach", Addison-Weseley, Wokingham, England, 1993.

[7] Fetteweis A: "Wave digital filters: theory and practice", IEEE Proceedings, 74(2), pp.270-327, Feb 1986.

[8] Renner K. and Gupta S. R.: "On the design of wave digital filters with a minimal number of multipliers", IEEE Trans. on circrcuits and Systems, vol. 21, pp.137-145, 1974.

[9] Fettweis A.: "Wave digital lattice filters", Int. J. Circuit Theory Appl., vol 2, pp.203-211, 1974 , vol CT-19, pp.668-673, Nov 1972.

[10] Fettweis A.: "Pseudopassivity, sensitivity, and stability of wave digital filters", IEEE Trans. Circuit Theory , vol CT-19, pp.668-673, Nov 1972.

[11] Viadyanathan P. P., Mitra S. K., and Neuvo Y.:"A new approach to the realisation of low sensitivity IIR digital filters", IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-34, pp. 350-361, 1986.

[12] Dempster A.G. and Macleod M.D.: "Multiplier blocks and the complexity of IIR structures", Electronics Letters, vol.30, pp.1841-1842, 1994.

[13] Holland, J.H.: "Adaptation in Natural and Artificial Systems," Univ. of Michigan Press, Ann Arbor, 1975.

[14] Golberg, D.E.: "Genetic algorithms in search, optimisation and machine learning," Addison Wesley, Reading, 1989.

[15] Horrocks D.H. and Khaliffa Y.M.: "Genetically derived filter circuits using preferred value components", IEE Colloquimum on Analogue Signal Processing, Oxford, UK, pp.4/1-4/5, 1994.

[16] Suckley D.: "Genetic algorithms in the design of FIR filters", IEE Proceedings-G, vol. 138, pp. 234-238, 1991.

[17] Wade G., Van-Eetvelt P, and Darwen H.: "Synthesis of effecient low-order FIR filters from primitive sections", IEE proc. G, pp.367-372, 1990.

[18] Roberts A. and Wade, G.:"A structured GA for FIR filter design", Proceedings of the workshop on "Natural algorithms in signal processing", vol. 1, pp.16/1-16/8, Chelmsford, U.K., Nov. 1993.

[19] Dexiang X.J. and Delay M.L.: "Design of finite word length digital filtrs using a parallel genetic algorithm", IEEE conference, pp.834-837, 1992.

[20] Wilson P.B. and Macleod M.D.: "Low implementation cost IIR digital filter design using genetic algorithms", Proceedings of the workshop on "Natural algorithms in signal processing", vol. 1, pp.4/1-4/8, Chelmsford, U.K., Nov. 1993.

[21] White M.S. and Flockton S.J.: "A comparative study of natural algorithms for adaptive IIR filtering", Proceedings of the workshop on "Natural algorithms in signal processing", vol. 1, pp.22/1-22/8, Chelmsford, U.K., Nov. 1993.

[22] Etter D.M., Hicks M.J., and Cho K.H.: "Recursive adaptive filter design using an adaptive genetic algorithm", Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, pp.635-638. 1982.

[23] Wicks T. and Lawson S.:"Genetic algorithm design of wave digital filters with a restricted coefficient set", Proceedings of the workshop on "Natural algorithms in signal processing", vol. 1, pp.17/1-171/7, Chelmsford, U.K., Nov. 1993.

[24] Little J.N. and Shure L.: "Signal processing toolbox for use with matlab", The mathworks Inc., July 1992.

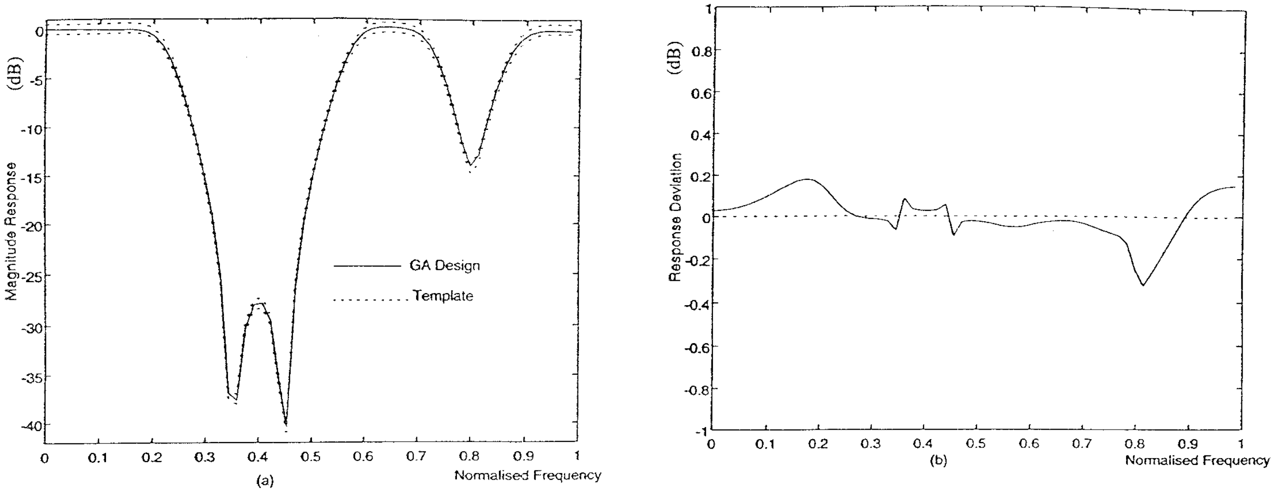


Figure 3 Response of GA designed arbitrary response 10th order filter.
(a) Complete response with template. (b) Tolerance band.

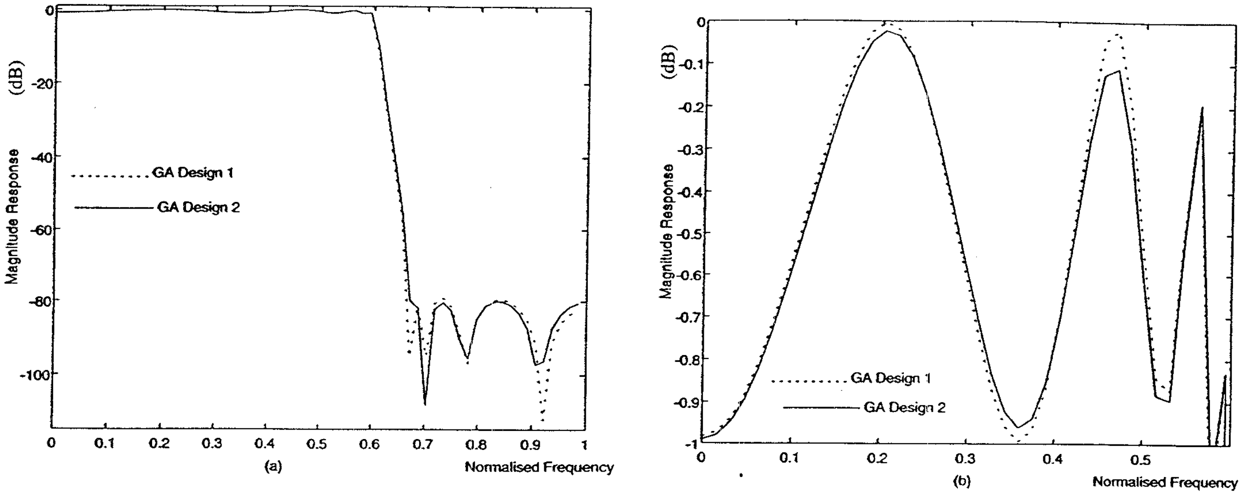


Figure 4 Response of two different GA designed 8th order elliptic IIR filter.
(a) Complete response. (b) Pass band.