The Design of Analogue and Digital Filters using Genetic Algorithms

T. Arslan and D.H. Horrocks[1]

1. Introduction

The conventional approach to filter design is to select one of the standard polynomial transfer functions that satisfies the response specification, followed by the implementation of the transfer function in one of the standard circuit structures. In many cases this approach is inadequate and an optimisation approach is required. The need for this can arise in a various ways, including non-standard response specification, and the consideration of practical effects such as finite-arithmetic effects and computational complexity for digital filters, and component selection and device imperfections in analogue filters.

Classical hill-climbing optimisation methods have succeeded in certain cases but are less suited to the general design task which can be mixed discrete/continuous, can have many local minima, and have high dimensionality. Recently Genetic Algorithm (GA) optimisation methods have emerged as a powerful approach to solving the more difficult optimisation problems. Publications are emerging in the literature on GAs applied to filter design and promise to move filter design significantly forwards to the ultimate goal of a single-step design from general specification to full practical implementation. This paper aims to: give a brief tutorial on GAs; provide a set of commented references on GAs applied to filter design; and present some new work on single-step design of finite wordlength (FWL) IIR digital filters that simultaneously satisfy magnitude and phase template specifications.

2. Genetic Algorithms

GAs [1,2] are search techniques which are based on mechanics of natural selection and the principle of *survival of the fittest.* They operate on a population of structures which are fixed length strings representing possible solutions of a given optimisation problem. A population of solutions is maintained and successive *generations* are produced by manipulating the solutions in the current population. Each solution has a *fitness* that measures its *competence*. New solutions are formed typically by merging two previous ones via a *crossover* operator. Other new solutions are simply modifications of previous ones, using a *mutation* operator. Successive generations are produced with new solutions probabilistically replacing older ones based on relative fitness. An ad hoc termination condition is often used and the best solution is usually reported. GAs have several advantages over traditional search and optimisation algorithms. These advantages stem in part from its ability to maintain simultaneously information about a variety of points in the solution space. This helps prevent the GA from being trapped at inferior local minima. Another feature of GAs is their use of building blocks in creating new solutions. This allows a GA to take advantage of the high quality sub-solutions that may already be present in existing solutions.

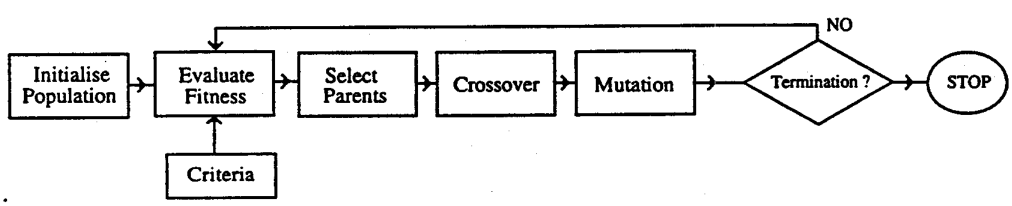


Figure 1: A Simple Genetic Algorithm

[1]The authors are with the University of Wales College of Cardiff, Cardiff.

3.GAs and Filter Design

Relatively little work has been published so far on GAs applied to analogue filters. A number of practical issues are important in analogue filter design. One is the choice of component values. On the grounds of cost, it can be cheaper to use components selected from a preferred set , such as the well known twelve-series of resistors and capacitors rather than special-valued components. In [3] GAs were successfully applied to second order active filter sections, to produce designs having gain constant, centre frequency and Q-factor that were closest to specified values. The approach is extended in [4] for higher order LC filters. The fitness function operated directly on the transfer function magnitude response rather than the polynomial function, thereby achieving a single-stage design process. GAs produce a family of solutions, and it is shown in [5] that the choice of filter topology has a marked affect on the number of solutions that can be obtained. Specifically the Leapfrog type of structures are preferred to the FDNR-type of structures, from this view point.

However, earliest application of GAs was to digital filtes. This started with the work of Suckley [6] who used GAs for the design of low-pass FIR filters providing structures of near minimal computational complexity. The designs produced by the GA are compared by evaluating the frequency response over the bands over which the filter is specified [7]. In [8], this work is further extended to the design of medium-order multiplierless FIR filters constructed from a cascade of linear phase primitive sections. Dexiang et. al. [9] 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. Recent work with FIR filters [10] uses GAs for rounding the coefficients of a filter represented using floating point numbers.

The development of methodologies such as primitive operators [11] provided an effecient framework for the application of genetic algorithms, such as the work in [12]. In addition, effecient reduced complexity implementation of two-dimentional FIR filters [13] allowed the subsequent use of genetic algorithms [14].

Recently, GAs have also been applied to IIR design problems. Wilson et. al. [15], 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 [16-18] 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. The work in [19] exploits the multiple criterion optimisation ability of GAs in investigating the optimisation of the individual sections involved in the design of cascaded IIR filters.

In [20] the authors describe a GA for the design of optimal FWL IIR filters with arbitrary magnitude response functions using a cascade of second order sections. This 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 [15] 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 proved successful in satisfying severely restricted templates in relatively small number of coefficient bits.

The work described in this paper is an extension of [20] in that the single-step design approach is extended to include phase as well as magnitude response of the IIR filter in the fitness calculation.

4. FWL IIR Digital Filters with Specified Magnitude/Phase Responses

4.1 The Genetic Algorithm - General Features

A conventional basic form of GA [1, 2] was found to perform well in this work. 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 are 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. 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 (2).

a_1	b_1	c_1	d_1	e_1	a_2	b_2	c_2	d_2	e_2	a_3	b_3	c_3	d_3	e_3

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

Figure 2 : Representation scheme for a three-stage cascaded structure

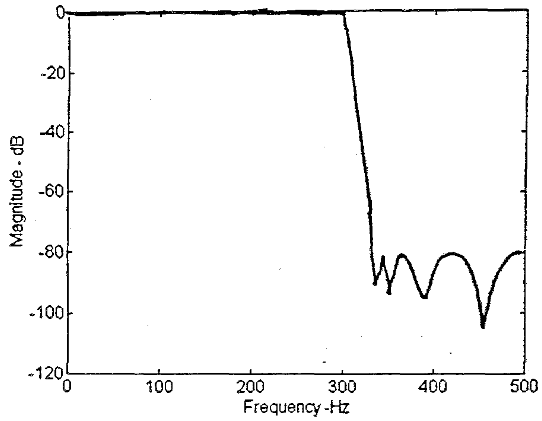
The initial population is derived from a set of infinite precision coefficients which are produced using a set of appropriate routines within the Matlab signal processing environment [20]. Next, the above coefficients are rounded to the nearest finite word length number and then transformed to a finite 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). A population size of a hundred was found to provide sufficient genetic diversity. Uniform two-point crossover is employed with a mutation procedure which involves the addition/subtraction of a random number to a random position in the chromosome. The GA performed best at crossover and mutation rates of 90% and 4% respectively.

4.2 Fitness Evaluation

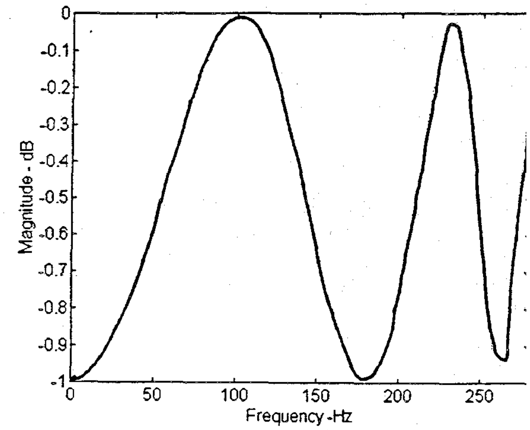
Fitness is evaluated in the normalised frequency range [0,1] over a uniform grid of frequency points. A template specification is defined in dBs and radians for the magnitude and phase responses. Separate total errors are defined for magnitude and phase responses. For the total magnitude error, this is a frequency-weighted sum-squared error across all frequency grid points, with the error at any frequency being the amount that the magnitude response is outside the template, and zero otherwise. Thus the total magnitude error will be zero if it fully satisfies the magnitude template. To avoid divide-by-zero errors as a consequence, if the total error is less than some small value, ε, it is replaced by ε, which we set at 10^{-5}. The reciprocal of the resulting total magnitude error was taken as the value for magnitude fitness. The phase fitness is calculated in the same way from the phase response. The fitness of the complete response is defined as the sum of magnitude and phase fitnesses. Thus any transfer function that completely satisfies the template can be identified by it having a combined fitness equal to $1/2\varepsilon$.

4.3 Results

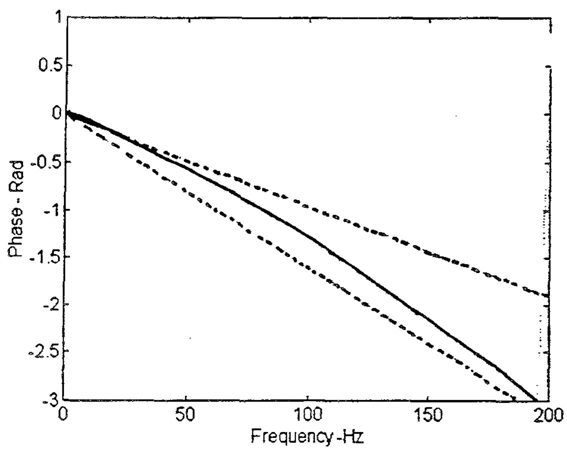
We demonstrate our results with an low-pass filter with the following specifications: Pass/stop band ripples 1dB/-75dB, and band edges at 280Hz/340Hz and a sampling frequency of 1000Hz. The phase response is specified by requiring a max/min time delay of 1.5/2.5 ms over the frequency range 0Hz to 90% of the passband edge. This requires a wedge shaped template as indicated in the phase response in figure 3 (c). The GA produced one solution that satisfied both magnitude and phase templates with four cascaded sections, in 18 generations. As the evolutionary process continued the number of correct designs increased up to 30 (in a population of 100) in 35 generations. Figure 3 illustrates one of the designs satisfying the given specification.



(a) Overall Magnitude Response



(b) Pass-band



(c) Phase response and template

Figure 3: Results for the low-pass filter example

References

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

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

[3] Horrocks, D.H. and Spittle M.C.: "Component Value Selection for Active Filters Using Genetic Algorithms", Proc. IEE/IEEE Workshop on Natural Algorithms in Signal Processing, Chelmsford, UK, November 1993, vol. 1, pp. 13/1-13/6.

[4] Horrocks, D.H. and Khalifa, Y.M.A.: "Genetically Derived Filter Circuits using Preferred Value Components", IEE Colloquium on Analogue Signal Processing, Oxford, UK, Oct. 1994, pp 4/1-4/5.

[5] Horrocks, D.H. and Khalifa, Y.M.A.: "Genetically Evolved FDNR and LeapFrog Filters using Preferred Component Values", Proc. European Conference on Circuit Theory and Design, Istanbul, Turkey, Aug. 1995, pp 359-362.

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

[7] 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.

[8] 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.

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

[10] Karaboga, N., Horrocks, D.H., and Karaboga, D.:" Design of FIR Filters using Genetic Algorithms", Proc. European Conference on Circuit Theory and Design, Istanbul, Turkey, Aug. 1995, pp 553-556.

[11] Bull, D.R. and Horrocks, D.H.:"Primitive Operator Digital Filters", IEE Proceedings-G, Vol. 138, No. 3, June 1991, pp 401-412.

[12] Bull, D.R. and Aladjidi, A.:"The Optimisation of Multiplier-Free Directed Graphs: an Approach using Genetic Algorithms", Proc. Int. Symposium on Circuits and Systems, London, England, June 1994, pp 93-96.

[13] Bull, D.R. and Horrocks, D.H.:" The Implementation of Reduced Complexity 2-D FIR Filters", Primitive Operator Digital Filters", roc. Int. Symposium on Circuits and Systems, Singapore, June 1991, pp 1653-1656.

[14] Sriranganathan, S., Bull, D.R., and Redmill, D.W.: "Design of 2-D Multiplierless FIR Filters Using Genetic Algorithms", Proceedings of IEE/IEEE Int. Conf. on Genetic Algorithms in Engineering Systems, Sheffield, UK, Sept. 1995, pp 282-286.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] Harris, S. P., and Ifeachor, E.C.: "Automating IIR Filter Design by Genetic Algorithms", Proceedings of IEE/IEEE Int. Conf. on Genetic Algorithms in Engineering Systems, Sheffield, UK, Sept. 1995, pp 271-275.

[20] Arslan, T., and Horrocks, D.H., "A Genetic Algorithm for the Design of Finite Word Length Arbitrary Response Cascaded IIR Digital Filters", Proceedings of IEE/IEEE Int. Conf. on Genetic Algorithms in Engineering Systems, Sheffield, UK, Sept. 1995, pp 276-281.