DOI: 10.21917/ijsc.2010.0003

USING GENETIC ALGORITHMS FOR DESIGNING OF FIR DIGITAL FILTERS

Alireza rezaee
Islamic Azad University-Hashtgerd Branch, Karaj, Iran
E-mail: arrezaee@yahoo.com

Abstract
In this paper, a new technique is presented for the design and optimization of digital FIR filters with coefficients that are presented in canonic signed-digit (CSD) format. Since such implementation requires no multipliers, it reduces the hardware cost and lowers the power consumption. The proposed technique considers three goals, the optimum number of coefficients, the optimum wordlength, and the optimum set of coefficients which satisfies the desirable frequency response and ensures the minimum hardware cost by minimizing the number of nonzero digits in CSD representation of the coefficients using Genetic Algorithms (GA). Comparing with equiripple method, the proposed technique results in about 30-40 percent reduction in hardware cost.

Keywords:
FIR Filter, Genetic Algorithm, Optimization, CSD, Hardware

1. INTRODUCTION

Finite impulse response (FIR) digital filters are preferred for their desirable features such as guaranteed stability, linear phase, and low coefficient sensitivity [1]. However, the price paid is the large number of arithmetic operations involved in their implementation that reduces the speed and increases the power consumption of the filter [2,3,4]. Considering the demands for low power consumption in modern applications such as wireless portable devices, reducing the power dissipation is one of the most important factors in the design and optimization process [1]. Besides, high speed digital signal processing (DSP) systems demand for low hardware complexity to ensure higher speeds.

The use of CSD multiplier coefficients reduces digital multiplication to a minimal sequence of shifts and adds [5,6,7]. The filter design problem is now finding an optimum number of coefficients, an optimum wordlength, and an optimum set of coefficients, which satisfies the desirable frequency response and minimize hardware complexity simultaneously. Hardware complexity can be translated to the number of nonzero digits in CSD representation of the coefficients which determines the number of adders and shifters that are necessary for implementation of the filter. The wordlength can also be an important factor in hardware complexity. In this paper, we present a novel technique for design and optimization of the FIR digital filters considering aforementioned issues. First, we determine the minimum number of coefficients and the minimum wordlength that are necessary to achieve desirable frequency response using equiripple method [8]. The reason of using equiripple method is that we have shown for every desirable frequency response, this algorithm obtains the optimal number of coefficients and also the optimal set of coefficients. This conclusion has been achieved by solving the design problem by GA for several examples and comparing the results with the results of the equiripple method. This comparison shows that the results of GA converge to the results of equiripple method. Since GA is a global optimum finder, we conclude that if we just consider the design and desirable frequency response and no hardware complexity optimization is made, the equiripple method achieves the optimal result. Therefore we set the initial number of coefficients and the wordlength with the values that are obtained from equiripple method.

The main idea of the proposed technique is simultaneous design and optimization of the filter using GA. Using trade offs between the number of coefficients and the wordlength, we have observed that by increasing the number of coefficients the wordlength can be decreased, resulting in considerable reduction in the number of nonzero digits that corresponds to reducing the hardware complexity. The results show that this reduction is about 30-40% that can considerably improve performance and power consumption of the filter.

This paper is organized as follows: in Section II, the CSD representation, genetic algorithm and equiripple method are introduced. In Section III, the proposed technique is presented. Section IV is dedicated to the design example. The conclusions are drawn in section V.

2. BACKGROUND

2.1 CANONIC SIGNED-DIGIT COEFFICIENTS

In the proposed technique, every filter coefficient is represented in CSD format. In this format each individual digit has a sign. The sign and value of the overall number is then determined by the weighted sum of these signed digits. These number systems can have any base but in its simplest form it is a ternary number system where the digits can take on the values 0, 1 or -1. A signed-digit number is represented by a string of ternary digits. Each of the ternary digits corresponds to a power-of-two (POT) number. For example, a W (W implies the wordlength) ternary digit number would be represented by the ternary digits t_0 through t_{W-1} and would have a value given by

$$value = [t_0 t_1 t_2 ... t_{W-1}] = \sum_{i=0}^{W-1} t_i \times 2^{-i} \qquad (1)$$

If the ternary sequence has no adjacent nonzero digits, it is called canonic signed-digit (CSD) representation. It is suitable for high throughput filters as it has the fewest number of nonzero digits [5, 6, and 7].

2.2 GENETIC ALGORITHMS

Genetic Algorithm optimizers are robust, stochastic search methods, modeled on the principles and concepts of natural selection and evolution. As an optimizer, the powerful heuristic of the GA is effecting in solving complex, combinatorial and related problems. GA optimizers are particularly effective when the goal is to find an approximate global minimum in a high-dimension, multi-model function domain, in a near-optimal manner [9].

In general, a GA optimizer must be able to perform six basic tasks [9]:

Encode the solution parameters as genes,

Create a string of the genes to form a chromosome,

Initialize s starting population,

Evaluate and assign fitness values to individuals in the population,

Perform reproduction through the fitness-weighted selection of individuals from the population, and

Perform recombination and mutation to produce members of the next generation.

A generic flow of a GA is shown in Fig.1.

2.3 EQUIRIPPLE METHOD

A digital FIR filter is characterized by the following z-transfer function:

$$H(z)=\sum_{n=0}^{N-1}h(n)z^{-n}. \tag{2}$$

where $h(n)$ is discrete time domain impulse response and N is the number of coefficients. The frequency response of the filter is given by:

$$H(e^{j\omega})=\sum_{n=0}^{N-1}h(n)e^{-j\omega n}. \tag{3}$$

The digital filter design problem is then finding a set of filter coefficients which satisfies the desired frequency response. There exists a number of methods for the design of linear phase FIR filters including window based design, Parks McClellan equiripple algorithm, least squares, and linear programming [11].

Considering two filter frequency response curves, $D(\omega)$ as the response of the ideal filter, and $H(\omega)$ as the actual frequency response, we have:

$$E(\omega)=W(\omega)[H(\omega)-D(\omega)] \tag{4}$$

where $E(\omega)$ is the error between the ideal and actual filter response values and $W(\omega)$ is the weighting factor, the optimal filter design problem is to determine an $H(\omega)$ that minimizes some measure or norm of $E(\omega)$ given a particular weighting function $W(\omega)$ and a desired response $D(\omega)$. Usually, the L_p norm is used to measure the error[12,13]. This norm is given by:

$$\int_{\Omega}[E[\omega]]^{p} \tag{5}$$

and this is the quantity we minimize. In practice the two most commonly used norms are *L2* and $L\infty$. Filter designs that minimize the $L\infty$ are attractive because they lead to equiripple solutions. When P goes to infinity, $L\infty$ norm simplifies to:

$$\max|E(\omega)|_{\omega\in\Omega} \tag{6}$$

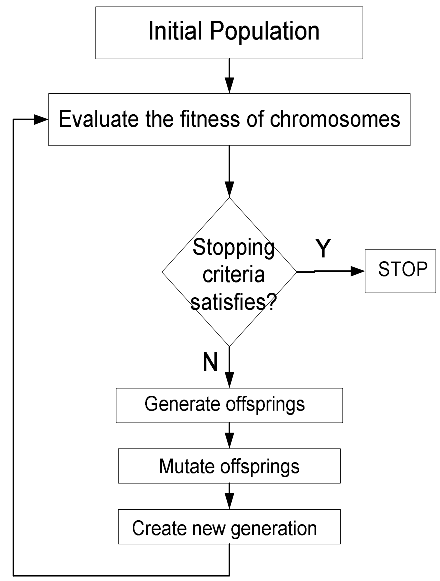

Fig.1 Typical flow of a genetic algorithm

meaning that when P equals ∞, the optimal design minimizes the maximum magnitude of the weighted error. Hence, it yields a minimax solution. The equiripple characteristics tend to produce the lowest order filter that satisfies some prescribed specifications [8].

Solving several examples by GA and comparing the results with the results of the Equiripple algorithm, we have found that the results of GA converge to the results of equiripple algorithm. Since GA is a global optimizer, we can consequent that for every desirable frequency response, equiripple algorithm obtains the optimal set of coefficients. So considering design issues only and having no optimization trends, equiripple algorithm can get the optimum solution.

However, if we want to consider hardware complexity and optimization constraints, equiripple algorithm can't furthermore be essential. Therefore, in the proposed technique we use equiripple method at the start point and then utilize GA to design an optimum filter which satisfies the desirable frequency response and tries to reduce hardware complexity and power dissipation simultaneously.

3. PROPOSED TECHNIQUE

In the proposed technique for the design and optimization of a digital FIR filter we consider following goals:

Finding the optimum number of coefficients.

Finding the optimum wordlength.

Finding the optimum set of coefficients.

As explained in part IV, we use equiripple algorithm at the start point to determine the initial values of the number of coefficients and the wordlength. Then we utilize GA to optimize these values. In running GA our fitness function considers two aspects, one is evaluating frequency response and the other is reducing the hardware complexity and power dissipation.

3.1 PRESENTATION OF THE PROPOSED TECHNIQUE

Algorithmic representation of the novel technique is as following:

Compute the initial number of coefficients and the wordlength which are necessary to satisfy the desired frequency response using equiripple algorithm.

Increase the number of coefficients by the minimum possible step.

Decrease the wordlength by one.

Run the genetic algorithm.

If the result of the step 4 is better than previous result (for the first iteration compare with the result of running equiripple algorithm), go to step 3.

Increase the number of coefficients by the minimum possible step.

Run the genetic algorithm.

If the result of the step 7 is better than previous result go to step 3.

End.

The main idea that is considered in the flow of the algorithm is the trade off exists between the number of coefficients and the wordlength. Considering this trade off, the algorithm attempts to find the optimum filter coefficients that satisfy the desired frequency response and have the lower number of nonzero digits in CSD representation of the filter coefficients.

3.2 PRESENTATION OF THE GENETIC ALGORITHM

Genetic algorithms are stochastic search methods and there are no deterministic rules to set the operators and options of a genetic algorithm to obtain the best algorithm considering convergence speed and the goodness of the obtained results. For every problem, corresponding to the nature of the problem, the operators and options are set differently to give the best algorithm.

Here we explain the genetic algorithm which we have used in the proposed technique. Every operator and option has been chosen to have the best algorithm. These selections have been done by solving an example by genetic algorithm with different kinds of operators and compare the speed of convergence and the goodness of obtained answers. The goodness criterion of an answer is determined by calculating its minimax error. As shown in [1] varying probabilities of crossover and mutation in different evolution stage improves the performance of the GA by preventing premature convergence and speeds up the convergence. We vary these two parameters by ±20%.

The objective function is formulated according to the minimax error criterion,

$$f = Max[H(e^{j\omega}) - H(e^{j\omega})] \quad (7)$$

To minimize the hardware cost and reduce the power dissipation, the information related to the hardware complexity is added to the fitness function as an optimization criterion. The fitness function is then defined as

$$fitness = 1/f + k(1/SoNZ). \quad (8)$$

while *SoNZ* is the sum of nonzero digits in the CSD representation of the coefficients and k is a positive number that controls the contribution of complexity criterion in the fitness function. We set k by 0.02. Table I shows the genetic algorithm options that we use in our proposed technique.

4. DESIGN EXAMPLE

To illustrate the proposed technique, we consider the design of a FIR halfband low-pass filter that meets the following specifications:

Normalized passband edge: 0.4
Normalized stopband edge: 0.6
Maximum passband ripple: 0.01
Maximum stopband attenuation: 40dB

Using MATLAB software to design a halfband FIR digital filter by equiripple method, an FIR filter with 23 coefficients and of wordlength 11 satisfies the given specifications. The number of nonzero digits in the CSD representation of the coefficients is 43. These results are shown in Table II.

If such a filter is designed with the proposed technique, an FIR filter with 27 taps and of wordlength 8 satisfies the given specifications. The number of nonzero digits in the CSD representation of the coefficients is 27. Therefore using the new technique, 37% reduction in the number of nonzero digits is achieved for this example. The results are shown in Table III. Fig.2 shows the frequency response of the designed filter and Fig.3 shows that of the designed filter using equiripple method for N=23 and N=21.

5. CONCLUSION

A new technique for the design and optimization of the digital FIR filters is presented. Filter coefficients represented in CSD format to reduce the hardware cost and lowers the power consumption by replacing the adders and shifters instead of multipliers. The proposed technique achieves the optimum number of coefficients, the optimum wordlength, and the optimum set of coefficients that satisfies the desired frequency response. Comparing with equiripple method, the proposed technique results in about 30-40 percent reduction in hardware cost.

Table.1 Genetic Algorithm Options

Population type	**Double Vector**
Population size	40
Creation function	Uniform
Initial range	[-1,1]
Fitness scaling	Rank
Selection	Roulette
Elite count	4
Mutation	Uniform(0.15)
Crossover	Two point
Crossover Fraction	0.6-0.9
Stopping criteria-stall generation	500

Table.2 Filter Coefficients Obtained by Equiripple Method

Coefficients	**Values**
c(1),c(23)	$-2^{-7}+2^{-10}$
c(22)	0
c(3),c(21)	$2^{-6}-2^{-9}-2^{-11}$
c(4),c(20)	0
c(5),c(19)	$-2^{-5}+2^{-7}-2^{-9}-2^{-11}$
c(6),c(18)	0
c(7),c(17)	$2^{-4}-2^{-6}+2^{-9}-2^{-11}$
c(8),c(16)	0
c(9),c(15)	$-2^{-3}+2^{-5}-2^{-9}-2^{-11}$
c(10),c(14)	0
c(11),c(13)	$2^{-2}+2^{-4}+2^{-9}+2^{-11}$
c(12)	2^{-1}

Table.3 Filter Coefficients Obtained by New Technique

Coefficients	**Values**
c(1),c(27)	-2^{-8}
c(22),c(26)	0
c(3),c(25)	-2^{-7}
c(4),c(24)	0
c(5),c(23)	2^{-6}
c(6),c(22)	0
c(7),c(21)	$-2^{-5}+2^{-8}$
c(8),c(20)	0
c(9),c(19)	$2^{-4}-2^{-6}+2^{-8}$
c(10),c(18)	0
c(11),c(17)	$-2^{-3}+2^{-5}-2^{-8}$
c(12),c(16)	0
c(13),c(15)	$2^{-2}+2^{-5}$
c(14)	2^{-1}

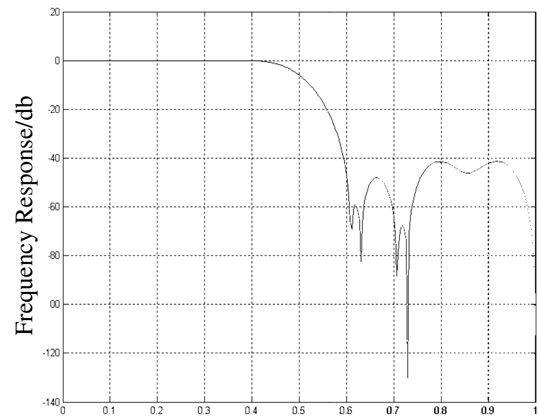

Fig.2. Frequency Response of the Designed Filter Using the Proposed Technique

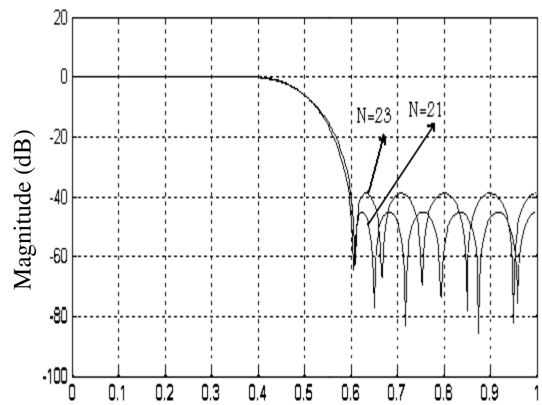

Fig.3. Frequency Response of the Designed Filter Using Equiripple Method for N=21 and N=23.

REFERENCES

[1] Ling Cen and Yong Lian, "A modified micro-genetic algorithm for the design of multiplierless digital FIR filters," IEEE Region 10 Conf. TENCON, Singapore, Vol. A, pp. 52-55, 2004.

[2] Ling Cen and Yong Lian, "Complexity reduction of high-speed FIR filters using micro-genetic algorithm," First Int. Symposium on Control, Communication and Signal Processing, Singapore, pp. 419-422, 2004.

[3] Yong Lian and Ling Cen, "A genetic algorithm for the design of low power high-speed FIR filters," Proc. Seventh Int. Symposium on Signal Processing and its Applications, Singapore, Vol. 1, pp. 181-184, 2003.

[4] G. Zhang and H. liu and W. Jin and L. Hu, "Multi-criterion satisfactory optimization method for designing FIR digital filters," Proc. Int. Conf. on Robotics, intelligent Systems and Signal Processing, China, Vol. 2, pp. 1339-1344, 2003.

[5] Fuller and B. Nowrouzian and F. Ashrafzadeh, "Optimization of FIR digital filters over the canonical

signed-digit coefficient space using genetic algorithms," Proc. Midwest Symposium on Circuits and Systems, Canada, pp. 456-459, 1998.

[6] T. Williams and M. Ahmadi and W. C. Miller, "Genetic algorithms for the design of digital filters using canonic signed digit coefficients," Proc. Int. Conf. on Signal Processing, Canada, Vol. 1, pp. 9-12, 2004.

[7] Fuller and B. Nowrouzian, "A novel technique for optimization over the canonical signed-digit number space using genetic algorithms," IEEE Int. Symposium on Circuits and Systems, Canada, Vol. 2, pp. 745-748, 2001.

[8] MATLAB Inc., Version 7.0.0, MATLAB Help, 2004.

[9] J. M. Johnson and Y. Rahmat-Samii, "Genetic algorithms in electromagnetic engineering," IEEE Antennas and Propagation Magazine, Vol. 39, pp. 7-21, 1997.

[10] M. Oner, "A genetic algorithm for optimization of linear phase FIR filter coefficient," Conf. Record of the Thirty-Second Asilomar Conf. on Signals, Systems & Computers, Turkey, Vol. 2, pp. 1397-1400, 1998.

[11] M. Oner and M. Askar, "Incremental design of high complexity FIR filters by genetic algorithms," Proc. Of the Fifth Int. Symposium on Signal Processing and its Applications, Turkey, Vol. 2, pp. 1005-1008, 1999.

[12] S. T. Tzeng, "Design of 2-D FIR digital filters with symmetric properties by genetic algorithm approach," IEEE Asia-Pacific Conf. on Circuits and Systems, Taiwan, Vol. 1, pp. 417-420, 2004.

[13] D. W. Redmill and D. R. Bull, "Automated design of low complexity FIR filters," Proc. IEEE Int. Symposium on Circuits and System, UK, Vol.5, pp. 429-432, 1998.