OPTIMAL RECURSIVE DIGITAL FILTER DESIGN USING IMPROVED GENETIC ALGORITHM

MORRIS ABRAHAM GNANAMUTHU EZRA, RAMAR K, ESWARAN C
Faculty of Engineering,
Multimedia University,
Jalan Multimedia, 63100 Cyberjaya, Selangor.
Malaysia.
ezra.morris@mmu.edu.my

Abstract: - In the process of designing digital recursive filters to satisfy a given magnitude response, the designer may end up with an unstable filter.. This unstable filter can be stabilized using methods such as Planar Least Square Inverse technique, Complex Ceptrum Method etc. The Planar Least Square Inverse technique is supposed to preserve the desired magnitude response which is found to be not true in all cases. Hence a new method of designing recursive filters using the improved genetic optimization algorithm is presented in this paper. The optimally designed filter will be stable and will have a magnitude response almost similar to the desired magnitude response.

Key-Words: - IIR Filters, Recursive Digital Filters, Genetic Optimization, Improved Genetic Algorithm, Signal Processing, Image Processing

1. Introduction

Two-dimensional filters have applications in image processing, geophysical signal processing, x-ray image processing etc [1][2]. Designing bounded input bounded output (stable) two dimensional digital recursive filters can be a time consuming process if one has to ensure the stability at the end of each iteration of the optimization process [3]. To overcome this, a filter can be initially designed to satisfy only the magnitude response specifications without considering the stability of the filter. If the designed filter is found to be unstable, it can be stabilized using the double planar least square inverse technique [4]-[6]. The magnitude response of the filter may get altered in the process of stabilization. A new technique using the improved genetic algorithm is presented in this paper to reduce the error between the magnitude response of the stabilized filter and that of the original filter without affecting the stability of the digital recursive filter.

2. Design Procedure using Improved Genetic Algorithm

To explain the procedure, consider a first order two-dimensional recursive digital filter as given below

$$H(z_1, z_2) = \frac{a_{00} + a_{01}z_2 + a_{10}z_1 + a_{11}z_1z_2}{b_{00} + b_{01}z_2 + b_{10}z_1 + b_{11}z_1z_2} \quad (1)$$

Let the filter represented by this transfer function be unstable. The filter can be stabilized using the double planar least square inverse technique. Let the transfer function of the resultant filter be

$$H'(z_1, z_2) = \frac{a_{00} + a_{01}z_2 + a_{10}z_1 + a_{11}z_1z_2}{b'_{00} + b'_{01}z_2 + b'_{10}z_1 + b'_{11}z_1z_2} \quad (2)$$

At the end of the stabilization process the magnitude response of the filter (1) might be different from that of the original filter. If the error in the magnitude response of the stabilized filter is found to be too large, the improved genetic algorithm technique is used to optimally reduce the error as small as possible by tuning the parameters of the numerator polynomial of (2). Let the resultant filter transfer function be

$$H''(z_1, z_2) = \frac{a'_{00} + a'_{01} z_2 + a'_{10} z_1 + a'_{11} z_1 z_2}{b'_{00} + b'_{01} z_2 + b'_{10} z_1 + b'_{11} z_1 z_2} \quad (3)$$

In the following chapter, the improved genetic algorithm technique is explained.

3. Parameter tuning using Improved Genetic Algorithm (IGA)

Genetic Algorithm is an efficient stochastic search technique wherein a population of randomly generated candidate solution evolves to an optimal solution via application of genetic operators such as selection, crossover and mutation. GA can help to find out the optimal solution globally over a domain [7]-[9].In improved GA (IGA), the standard GA is modified and new genetic operators are introduced to improve its performance. In this paper, an improved GA [10] has been used for tuning of the parameters $[a_{00} \quad a_{01} \quad a_{10} \quad a_{11}]$. First a population of chromosomes, say 'n' is created randomly. The fitness value, a non-negative figure of merit for each member of the population is computed. Next, some of the chromosomes are selected for performing genetic operations. The offspring obtained by genetic operation on the selected parent chromosomes replace the members in the initial population. This process repeats until the fitness value of a member is less than or equal to the desired fitness value.

In the genetic operation phase, the Roulette-Wheel technique [11] is used to select two chromosomes from the population for crossover operation. The crossover operation is then performed. The best offspring of the crossover operation undergoes mutation operation. Utilizing the mutation operator, three new offspring are created. These new offspring replace a few existing chromosomes having smaller fitness value. This process continues until the fitness value of any member is greater than or equal to the desired fitness value. Figure.1 gives the steps of genetic optimization.

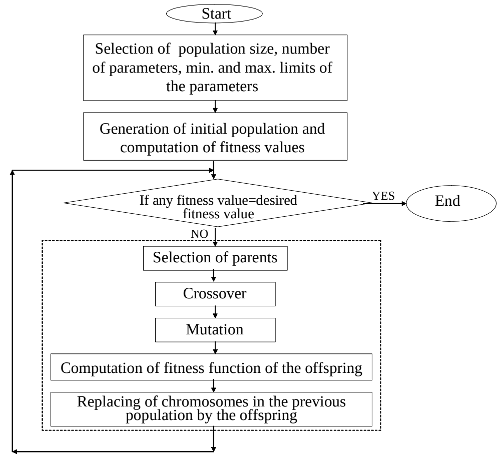

Fig 1: Steps of Genetic Optimization

3.1 Initial population

The initial population of size *n* is generated randomly as follows. Let.

$$P = \{ p_1, p_2, \ldots\ldots, p_n \} \quad (4)$$

and

$$p_i = \left[p_{i_1} \;\; p_{i_2} \;\; \cdots p_{i_j} \cdots p_{i_{no\_vars}} \right]$$
$$i = 1,2,\ldots, n; \quad (5)$$
$$p_{j\min} \le p_{i_j} \le p_{j\max} \quad (6)$$

For the filter problem,
n = population size
no_vars = number of variables to be tuned
p_{i_j} = parameters to be tuned
$p_{j\min}$ = min. value of the parameter p_{i_j}
$p_{j\max}$ = max. value of the parameter p_{i_j}

3.2 Evaluation

Each chromosome in the population is evaluated by the defined fitness function. The fitness function is defined as:

$$fitness\ function,\ F = \frac{1}{1 + \|e\|} \quad (7)$$

where

$$e = y^d - y$$
$$\|e\| \triangleq Max\left(\left|y^d - y\right|\right) \quad (8)$$

where

y^d = desired magnitude response

y = actual magnitude response

4. Genetic Operations

4.1 Selection

1. Starting with the first member of the existing population, for each member, i find $fit\_sum_i = \sum_{j=1}^{i} F_i \quad i = 1, 2, ..., n$ where F_i is the fitness value of i^{th} member as defined in (7).
2. Let the total of all the values of fitness functions, fit_sum_n be fit_sum.
3. Generate a random real number, *rand_sum* between 0 and *fit-sum*.
4. If $fit\_sum_i > rand\_sum$, then the i^{th} member is selected as a parent. Similarly, by generating another *rand_sum* the second parent is selected. Let par_1 and par_2 be the two selected parents.

4.2 Crossover

The crossover operation is mainly for exchanging information from the two parents. The two parents will produce four offspring as follows:

$$ch1 = \frac{par_1 + par_2}{2} \qquad (10)$$

$$ch2 = p_{max}(1-w) + \max(par_1, par_2)w \qquad (11)$$

$$ch3 = p_{min}(1-w) + \min(par_1, par_2)w \qquad (12)$$

$$ch4 = \frac{(p_{max} + p_{min})(1-w) + (par_1 + par_2)w}{2} \qquad (13)$$

$$p_{max} = [a_{00\,max} \quad a_{01\,max} \quad a_{10\,max} \quad a_{11\,max}] \qquad (14)$$

$$p_{min} = [a_{00\,min} \quad a_{01\,min} \quad a_{10\,min} \quad a_{11\,min}] \qquad (15)$$

where, w is the weight to be determined by the user. Among *ch1-ch4*, the one with the best (largest) fitness value is used as the offspring of the crossover operation and is denoted as

$$os = [a_{00} \quad a_{01} \quad a_{10} \quad a_{11}] \qquad (16)$$

$\max(par_1, par_2)$ is defined as the maximum value of the parameters in both the parents.

4.3 Mutation

The best offspring of the crossover operation will then undergo the mutation operation. The mutation operation is to change the genes of the chromosomes. Three new offspring will be generated by the following mutation operation

$$mut_j = [a_{00} \quad a_{01} \quad a_{10} \quad a_{11}] + [b_1\Delta m_1 \quad b_2\Delta m_2 \quad b_3\Delta m_3 \quad b_4\Delta m_4]$$

$$j = 1,2,\text{and } 3. \qquad (17)$$

where b_1, b_2, b_3, and b_4, can take the value of either 0 or 1. Δm_1, Δm_2, Δm_3 and Δm_4, are randomly generated numbers such that $(p_{i\,min} - os_i) \le (\Delta m_i) \le (p_{i\,max} - os_i)$. The first mutated offspring mut_1 is obtained according to (17) by assigning 1 to one of the b_i by random selection and all other three b_i values are set to zero. The second mutated offspring mut_2 is obtained by assigning 1 to two of the b_i's by random selection. The third offspring, mut_3 is obtained with all $b_i = 1$. These three new offspring will then be evaluated using the fitness function (8).

A method of selecting mutated offspring to replace one or more chromosomes in the population is given next. Let $p_a \in [0\ 1]$. The value p_a is the probability of accepting a bad offspring in order to reduce the chance of converging to a local optimum and is kept small. A real number is generated randomly and compared with p_a. If the number is smaller than p_a, then the one with the largest fitness value among the three new offspring will replace the chromosome in the population with the smallest fitness value. If the real number is larger than p_a and if $fitness(mut_1)$ > the smallest fitness value f_s, then the first offspring mut_1 will replace the chromosome with the smallest fitness value in the population. The second and the third offspring are tried successively. It may be noted that if

$$f_s > Max\begin{Bmatrix} fitness(mut1), \\ fitness(mut2),\ fitness(mut3) \end{Bmatrix}$$

then there will be no replacement in the initial population. After the operation of selection, crossover, and mutation, a new population is generated. This genetic process is continued until the required accuracy is achieved.

5. Example:

Consider the following 1st order two-dimensional digital recursive filter function

$$H(z_1, z_2) = \frac{1 + z_2 + z_1 + z_1 z_2}{9.3547 + 9.1690 z_2 + 4.1027 z_1 + 8.9365 z_1 z_2} \quad (18)$$

The magnitude response of this filter function is shown in figure 1.

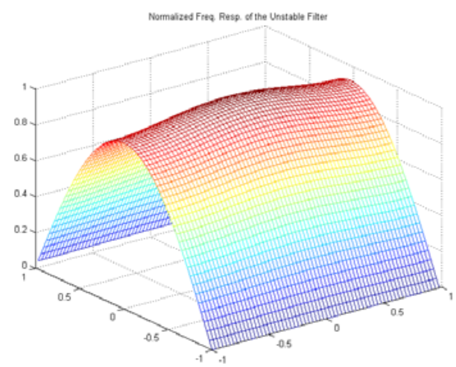

Fig 2: Normalized Magnitude Response of the Unstable Filter

The above filter function is found to be unstable. Hence we stabilize the filter using the double planar least square inverse technique. The resultant transfer function of the stabilized filter is found to be

$$H(z_1, z_2) = \frac{1 + z_2 + z_1 + z_1 z_2}{1.3309 + 0.4667 z_2 + 0.4595 z_1 + 0.2378 z_1 z_2} \quad (19)$$

The magnitude response of this filter is shown in figure 2. Figure 3 shows the error in the magnitude response of the original filter compared with that of the stabilized filter.

The parameters of the numerator polynomial of the stable filter transfer function shown in equation (19) is tuned using the Improved Genetic Algorithm discussed in sections 3 and 4.

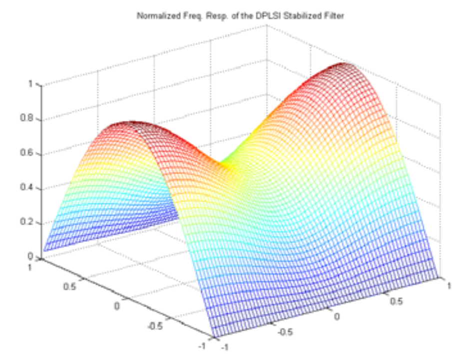

Fig 3: Normalized Magnitude Response of the Stable Filter

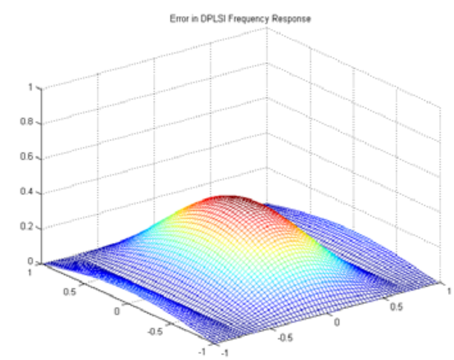

Fig 4: Error between the Unstable and Stable Magnitude Responses

In the algorithm, the following parameters are used:

Population size = 20

No. of variables = 4

Max. value of all the variables = 3

Min. value of all the variables = -3

Weight parameter w = 0.99

Desired fitness value = 0.995

The optimized filter transfer function is given by

$$H(z_1, z_2) = \frac{2.2870 + 0.8052 z_2 + 2.4910 z_1 + 0.8790 z_1 z_2}{9.3547 + 9.1690 z_2 + 4.1027 z_1 + 8.9365 z_1 z_2} \quad (20)$$

The magnitude response of the above filter function and the error plot are shown in figures 5 and 6 respectively.

It may be noted that the magnitude response of the optimized filter is almost same as that of the original unstable filter.

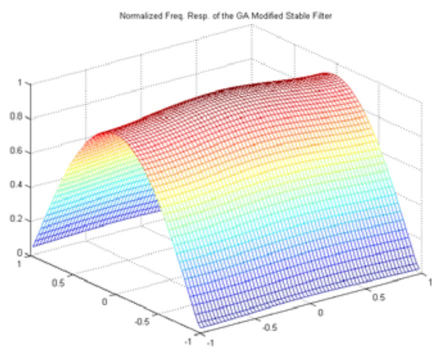

Fig 5: Normalized Magnitude Response of the IGA Modified Stable Filter

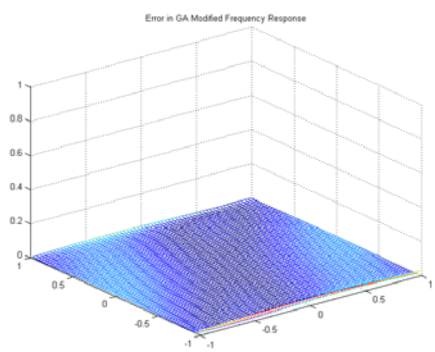

Fig 6: Error between the Unstable and IGA Modified Stable Filter Magnitude Responses

6. Conclusion:

In this paper, a new method for an optimal design of two dimensional digital recursive filters using improved genetic algorithms is presented. This method can be used to filters that were stabilized using any of the known stabilization techniques. The improved Genetic algorithm technique helps to modify the magnitude response of the stable filter so that it is very close to those of the original unstable filter.

References:

[1] K.R. Castleman, *Digital Image Processing*, Englewood Cliffs, NJ; Prentice-Hall 1996.

[2] R. A. Stein and N. R. Bartley, "Continuously time-variable recursive digital band-pass filters for seismic signal processing," *Geophysics*, Vol. 48, No. 6, pp. 702-712, June 1983.

[3] Jae S. Lim, *Two Dimensional Signal and Image Processing*, Englewood Cliffs, NJ, Prentice Hall, 1990.

[4] J. L. Shanks and S. Treitel, "Stability and synthesis of two dimensional recursive filters", *IEEE Transactions on Audio and Electroacoustics*, Vol. 20, No. 2, pp. 115-128, June 1972.

[5] P. S. Reddy, D. Raghu Rami Reddy and M.N.S. Swamy, "Proof of a modified form of Shanks' conjecture on the stability of 2-D planar least square inverse polynomials and its implications", *IEEE Trans. Circuits and Syst.*, Vol. 31, pp. 1009-1015, Dec. 1984.

[6] P. Delsarte, Y. Genin and Y. Kemp, "Comments on proof of a modified form of Shanks' conjecture on the stability of 2-D planar least square inverse polynomials and its implications", *IEEE Trans. Circuits and Syst.*, Vol. 32, pp. 966, Sept. 1985.

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

[8] D.T. Pham and D. Karaboga, "Intelligent Optimization Techniques, Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks", New York: Springer-Verlang, 2000.

[9] Z. Michalewics, "Genetic Algorithm+Data Structures = Evolution Programs", 2nd Extended ed. New York: Springer-Verlang, 1994.

[10] Frank H.F. Leung, H.K. Lam, S.H. Ling and Peter K.S. Tam, "Tuning of the Structure and Parameters of a Neural Network Using an Improved Genetic Algorithm", IEEE Trans. Neural Networks, Vol. 14, No. 1, pp. 79 -87, January 2003.

[11] Kamepalli, H.B., "The Optimal Basics for GAs", IEEE Potentials, Vol. 20, Issue 2, pp. 25-27, 2001.