



**HAL**  
open science

## An advanced genetic algorithm for designing 2-D FIR filters

Kamal Boudjelaba, Frédéric Ros, Djamel Chikouche

► **To cite this version:**

Kamal Boudjelaba, Frédéric Ros, Djamel Chikouche. An advanced genetic algorithm for designing 2-D FIR filters. 2011 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Aug 2011, Victoria, Canada. pp.60 - 65, 10.1109/PACRIM.2011.6032868 . hal-01076880

**HAL Id: hal-01076880**

**<https://hal.science/hal-01076880>**

Submitted on 23 Oct 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

# AN ADVANCED GENETIC ALGORITHM FOR DESIGNING 2-D FIR FILTERS

Kamal Boudjelaba<sup>1</sup>, Frédéric Ros<sup>1</sup>, Djamel Chikouche<sup>2</sup>

1. Prisme Laboratory, IRAuS Pole, Axis Image and vision, Polytech'Orléans, 12 rue de Blois, Orléans, France
2. Electronics Department, University of M'sila, Rue Ichebilia, M'sila, Algeria

## Abstract

*In this paper, a new genetic algorithm is proposed for designing 2-D FIR filters with the objective of being relevant and tractable. The key point of our approach stems in the capacity of our GA to adapt the genetic operators during the genetic life while remaining simple and easy to implement. It hybridizes the use of conventional and dedicated processes. The first are capable of covering a wide range of optimization problems without being sufficient. The second includes the filter design specificities and takes feedback from the current state of the genetic search. We demonstrate the method efficiency on various cases of filter design.*

**Keywords**– 2-D FIR digital filters, adaptive GA approach, real-valued chromosomes.

## 1. Introduction

In recent years, with the rapid improvement in computer technology, two dimensional (2-D) digital signal processing has become more important. Therefore, the design problem of 2-D digital filters has been receiving a great deal of attention. Digital filters can be classified into two groups, i.e., finite impulse response (FIR) filters and infinite impulse response (IIR) filters. Since FIR digital filters are inherently stable and they can have linear phase, they are often preferred over IIR filters. These 2-D FIR filters have many important applications, e.g., in radar and sonar signal processing and in image processing. In this area, it can be used to remove the effects of some degradation mechanisms or enhance the image to facilitate identification. The linear-phase filter is important in DSP applications such as image processing, where phase information must not be altered. The problem of filter design is to find a realization of the filter which meets each of the requirements to a sufficient degree to make it useful. The techniques for designing 2-D FIR digital filters have been developed extensively for several years [1]. The results of most of these techniques are given in the form of the impulse response of a 2-D filter, so the designed filter is suitable for a direct convolution realization. The classic methods are based on approximating some specified frequency response, the least-square (LS) or the minimax error criteria are usually used. By using the LS error criterion, one gets an overshoot of the frequency response at the pass band and the stop band edges caused by Gibbs phenomenon [2].

There are different methods [1,3] to find the coefficients from frequency specifications: Window design method, frequency transformation method, frequency sampling method, weighting least squares design, equiripple design. Filters generated using these approaches often contain many small ripples in the pass band, since such a filter minimizes the peak error. In the window and frequency sampling methods, it remains difficult to control cutoff frequencies of the pass-band and stop-band with accuracy. With the windowing method, truncating the Fourier series causes a phenomenon called the "Gibbs effect" a spike occurs wherever there is a discontinuity in the desired magnitude of the filter. The frequency transformation method preserves most of the characteristics of the one-dimensional filter, particularly the transition bandwidth and ripple characteristics [4]. These approaches are considered as not efficient enough for practical implementations. New design approaches [2,5] have been investigated, and more interesting results have been obtained without being completely satisfactory ; the authors deal only with very specific cases and do not argue that these methods give good results for all cases.

The popular weight least square (WLS) algorithm gives some improvements, and can acquire analytical solutions. At the opposite, it must calculate an inverse matrix which can be computationally problematic as soon as the filter's degree is very high.

Some approaches have emerged for several years on the basis of dedicated evolutionary algorithms [6] and

particularly Genetic Algorithms (GAs). Studies related to the design of FIR filters by GAs have however been more confidential. One can find pioneer papers at the end of ninety. They are especially devoted to 1-D FIR filters and very often to the design of digital filters. Applications are more related to low order filters [7]. They however yielded solutions whose quality often significantly overcomes that obtained by methods previously applied. Design 1-D FIR filter has been discussed in detail in [8]. The idea consists in including ripple error in the minimization algorithm. Results are promising. At the opposite, they can be very changeable especially when the filter dimension becomes large.

Factually, the design of 2-D arbitrary FIR filter is more difficult as the searching space is larger [9-10]. GAs are good candidates because efficient solutions can be obtained even for complicated optimization problems involving large and complex searching spaces. However, despite these no discussing successes, applying GA for a dedicated problem is not always straightforward, and objectively their implementation and use face various problems. Diversity and elitism are the main factors to manage with GA. Population diversity enables the fruitful exploration of the search space [11]. Without enough diversity the search can be trapped prematurely in a local minimum, and without enough elitism, the search process becomes random and the algorithm cannot converge. Different approaches coming from the specialized literature still exist [12]. Niching approaches, the most popular being sharing and crowding methods, have been developed to counteract the convergence of the population to a single solution by maintaining a diverse population of individuals throughout its search. Many works [13] have also focused on parameter adaptation (mutation, crossover rates and population sizes) and on the proposals of new variants (intelligent crossover, elitist recombination...). All the ideas developed are very promising and have given interesting results. However, the amount of simultaneous parameterization and the tuning of several parameters can often necessary lead to extremely high computation costs without insuring of good results. In [14] we have presented the potential of several evolutionary algorithms for the 2-D FIR filters design and proposed a specific GA producing very interesting results compared to the state of the art. In this paper we propose a more advanced GA to improve both the run time and the result accuracies by adding more efficient mechanisms. This paper is organized as follows. Section 2 briefly describes the characteristics of digital filter design and its reformulation as a constrained minimization problem. Section 3 presents our genetic approach, and section 4 is dedicated to experimental results. Finally, Section 5 reports some concluding remarks.

## 2. Problem formulation

The design of digital filters, as with most engineering tasks, is a multistage, iterative process. The key stages are filter specification, coefficient calculation and structure realization. The set of filter coefficients has to meet performance specifications such as passband width and corresponding gain, width of the stopband and attenuation, band edge frequencies, and tolerable peak ripple in the passband and stopband. An attractive goal is to perform the optimization by several approaches in order to seek out the filter design method with the best results and the best performances.

The transfer function of a 2-D FIR filter of dimension  $N_1 \times N_2$  is given by [1, 2]:

$$H(Z_1, Z_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} h(n_1, n_2) * Z_1^{-n_1} * Z_2^{-n_2} \quad (1)$$

Where  $h(n_1, n_2)$  is its impulse response.

The frequency response of a 2-D FIR digital filter with its impulse response  $h(n_1, n_2)$  is given by:

$$H(\omega_1, \omega_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} h(n_1, n_2) * e^{-j.(n_1\omega_1+n_2\omega_2)} = M(\omega_1, \omega_2) * e^{j.\theta(\omega_1, \omega_2)} \quad (2)$$

$M(\omega_1, \omega_2)$  the magnitude response of  $H(\omega_1, \omega_2)$ , is real-valued function.

For its a symmetrical impulse response, the frequency response is given by:

$$H(\omega_1, \omega_2) = M(\omega_1, \omega_2) * e^{-j\left[\frac{(N_1-1)}{2}\omega_1 + \frac{(N_2-1)}{2}\omega_2\right]} \quad (3)$$

$$M(\omega_1, \omega_2) = |F(\omega_1, \omega_2)|$$

$$F(\omega_1, \omega_2) = \sum_{k_1=0}^{P_1} \sum_{k_2=0}^{P_2} a(k_1, k_2) * \cos(k_1\omega_1) * \cos(k_2\omega_2) \quad (4)$$

$$P_1 = \frac{(N_1-1)}{2} \quad \text{and} \quad P_2 = \frac{(N_2-1)}{2}$$

The objective is then the minimization of the sum-squared error over frequency points ( $m_1 \times m_2$ ):

$$E = \sum_{j=1}^{m_1} \sum_{k=1}^{m_2} [D(\omega_{1j}, \omega_{2k}) - F(\omega_{1j}, \omega_{2k})]^2 \quad (5)$$

Where  $D$  is the desired magnitude response and  $F$  the actual magnitude response.

$$E_i = \sum_{j=1}^{m_1} \sum_{k=1}^{m_2} [D(\omega_{1j}, \omega_{2k}) - C_1(\omega_{1j}) * A_i * C_2(\omega_{2k})^T]^2 \quad (6)$$

The main idea is to obtain a least-squares approximation to the given magnitude function, which leads to the optimal approximation in the Chebyshev sense.

$$C_1(\omega_{1j}) = \begin{bmatrix} 1 & \cos(\omega_{11}) & \cdots & \cos(P_1 \cdot \omega_{11}) \\ 1 & \cos(\omega_{12}) & \cdots & \cos(P_1 \cdot \omega_{12}) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \cos(\omega_{1m_1}) & \cdots & \cos(P_1 \cdot \omega_{1m_1}) \end{bmatrix} \quad (7)$$

$$C_2(\omega_{2k}) = \begin{bmatrix} 1 & \cos(\omega_{21}) & \cdots & \cos(P_2 \cdot \omega_{21}) \\ 1 & \cos(\omega_{22}) & \cdots & \cos(P_2 \cdot \omega_{22}) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \cos(\omega_{2m_2}) & \cdots & \cos(P_2 \cdot \omega_{2m_2}) \end{bmatrix} \quad (8)$$

The matrix ( $A_i$ ) formed by the filter coefficients can be characterized as:

$$A_i = \begin{bmatrix} a_{00} & a_{01} & \cdots & a_{0p_2} \\ a_{10} & a_{11} & \cdots & a_{1p_2} \\ \vdots & \vdots & \vdots & \vdots \\ a_{p_10} & a_{p_11} & \cdots & a_{p_1p_2} \end{bmatrix} \quad (9)$$

$a_{k_1 k_2}$  is ( $k_1, k_2$ )th filter coefficient.

### 3. Our proposed approach

In order to accelerate evolutionary process and reach an efficient solution with a reasonable execution time, several mechanisms have been incorporated in our GA devoted to FIR filter design: Firstly the use of an initial population presenting chromosomes close to acceptable solutions coming from last experiences obtained with standard approaches. Secondly, genetic operators, and particularly the mutation operator, that are completely adapted to the level of the genetic process and therefore limit the simultaneous tuning of several static parameters. Thirdly, a ranking selection scheme has been introduced in order to limit the promotion of extraordinary chromosomes, and then preventing premature convergences. To improve accuracy, the genetic exploration is driven by a fitness assignment strategy including the filter specificities. Driving GAs by this way is necessary. It is however not adaptable enough to continuously manage the trade-off between elitism and pressure preservation. Our algorithm is therefore reinforced by a mechanism involving a breaking process to reseed the population when necessary. The

selection process naturally promotes the best chromosomes. It integrates a function allowing the limitation of redundant chromosomes then promoting diversification. The GA is based on the assumption that it is generally difficult to pursue different optima in parallel within the theory of GAs but easier to focus attention on one optimum to pass from another one if better.

### 3.1. The genetic algorithm

The GA is a real-coded genetic algorithm (RCGA). A chromosome solution is directly represented by a matrix of coefficients representing the filter itself. It means that for a filter of  $N_1 \times N_2$  dimension, it is necessary to determine  $N_1 \times N_2$  coefficients, the goal being to find them in order to minimize the sum-squared error  $E$  over  $m_1 \times m_2$  frequency points. The fitness function is defined as the inverse of the error  $E$  presenting the goodness to be maximized. For a given shape filter, peak ripples are generally more present in certain regions than others making their attenuation more difficult. To face this situation, we suggest as novelty to weight the sum-squared error by a matrix  $W$  of  $m_1 \times m_2$  dimensions and therefore compensate this imbalance.

The selected scheme is classically driven by the different operators of crossover, mutation, selection and reproduction as illustrated in Fig. 1. It however includes some major differences detailed later in the way of applying the successive operations. The  $n$  chromosomes from generation  $t$  produce  $2n$  chromosomes via the crossover operator. In this scheme, there is no need to fix a crossover rate. A copy of the parent and children population is kept. Parents and children are put together and submitted to the mutation operator. Each chromosome component has a  $P_M$  probability to be mutated. The  $P_M$  value is calculated for each chromosome individually as well as the level of mutation and is dependent on the genetic advance of the algorithm. An enlarged population of  $4n$  chromosomes is then submitted to the selection operator that discards the redundant and worse chromosomes to obtain a population of size  $n$ . When this process generates less than  $n$  chromosomes, the population is completed with new chromosomes generated from the last population having submitted a large mutation.

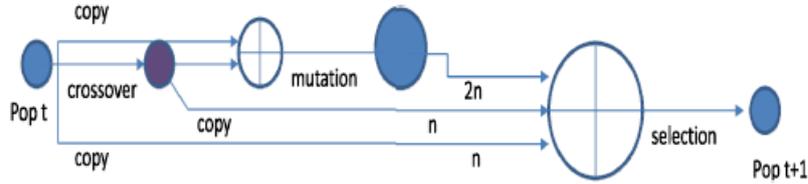


Fig.1. Genetic process

### 3.2. Genetic operations and dedicated mechanisms to improve convergence

Some researchers put considerable efforts into finding parameter values which give reasonable performance for a wide range of problems. Specific problems however require dedicated parameters stressing the need for additional mechanisms allowing good parameter tuning methods. Our algorithm hybrid the two approaches.

*Crossover* is the primary operator in the GAs and is the key to the success of GAs. The operator generates two children  $pop_k^{child}$  from two parents  $i$  and  $j$  according to the following rule:

$$\begin{cases} pop^{child1} = (pop^i - pop^j) * \lambda_{c1} + pop^i \\ pop^{child2} = (pop^j - pop^i) * \lambda_{c2} + pop^j \end{cases} \quad (10)$$

where  $pop^i$  ( $pop^j$ ) presents the individual  $i$  ( $j$ ),  $\lambda_{c1}$  and  $\lambda_{c2}$  are random number between 0 and 1. Crossover enables the generation of new chromosomes without introducing any new genetic features into the population at the gene level.

*Mutation* operator arbitrarily alters the gene value according to a predetermined probability. Mutation operator introduces diversity and reflects features which are not presented in the current population. We suggest a non-uniform adaptive mutation directly applicable to each chromosome of the population. The mutation rate is limited for the best chromosomes and encouraged for the worse. For a given chromosome,  $P_M$  is applied to each gene.

$$P_M = \begin{cases} \frac{k_1 * (f_{max} - f)}{(f_{max} - f_{avg})} & f \geq f_{avg} \\ k_2 & f < f_{avg} \end{cases} \quad (11)$$

Where  $f_{max}$  and  $f_{avg}$  are the maximum and average fitness values in the current population, respectively, and  $f$  is the fitness value of the current chromosome.  $k_1$  and  $k_2$  are weighing parameters with  $k_1$  and  $k_2 \leq 1$ . We propose a novelty that lies in the conditions of use thereof, i.e., where the population evolves in a normal manner and that the population is diverse. The simple idea consists of restarting a process of evolution that has stalled by refreshing more chromosomes.  $P_M$  is increased one time via the  $k_1$  and  $k_2$  values each time the population tends to get stuck in a local optimum and decreased when the population is scattered in the search space. The level of mutation also takes feedback from the current state of the search and modifies the chromosomes accordingly. We suggest adopting a two-case version described as follows:

$$pop^{new} = \begin{cases} pop^{old} * (1 + \lambda_{m1}) & \text{if local min} \\ pop^{old} * (1 + \lambda_{m2}) & \text{else} \end{cases} \quad (12)$$

Where  $\lambda_{m1}$ ,  $\lambda_{m2}$  are parameters depending on the genetic state ( $\lambda_{m1} \gg \lambda_{m2}$ ).

Our *selection* scheme aims at improving the quality of the current population by giving chromosomes of higher quality a higher probability to be inserted into the next population without reducing the population diversity:

$$P_i = f_i / \sum_{j=1}^n f_j \quad (13)$$

where  $f_j$  is the fitness value of the  $j^{th}$  chromosome.

The scheme includes the genetic materials and prevents very fit individuals from gaining dominance early at the expense of less fit ones. The algorithm starts with the best chromosome called *current*. This chromosome is automatically selected and firstly compared to the second called *inspected* via the fitness function. If there is no match between them, the second becomes *current* and the procedure continues. Otherwise, the comparison is done via a gene metric to avoid discarding two chromosomes having similar fitness values while being genetic different. *Inspected* is discarded if there is no match between them in this space. In this case the first chromosome is still *current*. The comparison is then done with the third chromosome which becomes *inspected*. The procedure continues until the whole population is inspected.

$$\begin{cases} \text{if } \Delta_{fitness}(ch_{current}, ch_{inspected}) \geq f_\epsilon \text{ continue else} \\ \text{if } \Delta_{gene}(ch_{current}, ch_{inspected}) \geq g_\epsilon \text{ continue else} \\ \text{discard } ch_{inspected} \end{cases} \quad (14)$$

$ch_{current}$  ( $ch_{inspected}$ ) stands for chromosome *current* as  $\Delta_{fitness}$  (*resp.*  $\Delta_{gene}$ ) stands for distance in the fitness space, and  $f_\epsilon$  (*resp.*  $g_\epsilon$ ) is a predefined threshold. The second calculation is processed only when *current* and *inspected* fit very well in the fitness space. This speeds up the process.

### 3.3. Breaking processes to prevent premature convergence

There exists no manageable model for controllable selection pressure. Premature convergence is then naturally possible. It has to be detected and corrected. The correction is done via a breaking mechanism capable of refreshing the active population without losing its current advance. The detection consists of identifying an evolution process that has stalled materialized by the presence of a lot of similar chromosomes within the population.

According to a given metric, one way aims at calculating or estimating the percentage of similar chromosomes within the population and reseeding the population accordingly. There is the possibility to work directly on the fitness space by comparing the best chromosome and the average fitness function. In both cases, the reseeding is submitted to the level of the genetic advance.

$$\begin{cases} \text{if } \Delta f_{fitness}(f(ch_{best}), f(ch_{ave})) \leq \Delta f_{min} & \text{and} \\ f(ch_{best}) \leq f_{end} & \text{breaking process active} \\ \text{else} & \text{breaking process inactive} \end{cases} \quad (15)$$

Where  $\Delta f_{min}$  and  $f_{end}$  are the different thresholds and  $f(x)$  the fitness of chromosome  $x$ .

#### 4. Experimental results

The objective of this section is to assess the contribution of the different mechanisms introduced in the GA. The mechanisms “Initialization”, “Weighting function”, “Adaptive mutation”, “Selection with diversification” and “Breaking process” are respectively named  $M_1, M_2, \dots, M_5$ .

By hybridizing their use, different algorithm versions (see Table 1) have been implemented. Our advanced GA is  $GA_1$ .

Table 1. Details of the different GAs

	$M_1$	$M_2$	$M_3$	$M_4$	$M_5$
$GA_1$	Yes	Yes	Yes	Yes	Yes
$GA_2$	Yes	Yes	Yes		Yes
$GA_3$	Yes	Yes		Yes	Yes
$GA_4$	Yes	Yes	Yes		
$GA_5$	Yes	Yes		Yes	
$GA_6$	Yes	Yes			

Six of them have been selected to concentrate the discussion on three attributes as the role of the remaining is obvious. The GAs have been applied to various classical filters including high-pass, low-pass, band-pass and band-cut filters of different orders with the same parameter set. The number of frequency points ( $m_1 \times m_2$ ) has been put to  $20 \times 20$ . Numerical results (See Table 2) are detailed only for the  $7 \times 7$  low pass filter and Fig. 5 illustrates the results of other filters. The population size  $P_s$  has been fixed to 100, a maximum number of genetic generations to 1500 and an implicit probability of crossover to 100%.

The initial chromosomes were generated by considering solutions obtained by more conventional approaches mutated via elementary statistics. The parameters  $f_\varepsilon, g_\varepsilon,$  and  $f_{end}$  were respectively fixed to 0.00001, 0.0001, and 0.008.  $\Delta f_{min} = 0.07 * fit_{ave}, fit_{ave}$  being the fitness average of the whole population. The parameters  $k_1$  and  $k_2$  have been respectively fixed to (0,001) and (0,01) for the normal genetic advance and increased to 0,1 and 0,3 when a sign of premature convergence has been detected. The parameters  $\lambda_{m1}$  and  $\lambda_{m2}$  have been put to  $0.5 * randn$  and  $0.05 * randn$ . The weight matrix  $W$  is composed of two values:  $c_0=1$  in the passband and  $c_1=5/8$  outside.

Table 2. Results obtained for a low-pass filter

	Error	Peak Error	Width	CPU	Ite
$GA_1$	0.0092	0.0803	very small	49.74s	298
$GA_2$	0.0116	0.0874	very small	61.83s	371
$GA_3$	0.0200	0.1055	Middle	75.63s	458
$GA_4$	0.0193	0.1092	Small	59.62s	359
$GA_5$	0.0709	0.8597	Middle	247.86s	1500
$GA_6$	0.0212	0.1086	Middle	205.42s	1380
FT	0.0566	0.1668	Middle	< 2s	
FS	0.0551	0.2203	Very small	< 2s	
W	0.0400	0.4920	Large	< 2s	

We clearly see that  $GA_1$  produces better results than the classic methods and the other versions for almost all attributes.  $GA_5$  is less relevant than all the others. According to the “Error” attribute that is directly linked to the

fitness function, GA<sub>1</sub> is the most efficient. We should notice the particular role of the couple “Adaptive mutation rate” and “Breaking process”. From two errors [0,0116; 0,0874] obtained with GA<sub>2</sub>, we move to [0,0709; 0,8597] for GA<sub>5</sub>. The added value of the “Breaking mechanism” taken alone can be seen from the GA<sub>5</sub> to GA<sub>3</sub> version. The “Adaptive mutation” affords a significant contribution denoted by analyzing the GA<sub>4</sub> performances. The “Selection with diversification” mechanism has no effect by itself as proved by the GA<sub>5</sub> results. It however contributes to the performances when included with the other mechanisms. For GA<sub>1</sub>, we can observe that both the performances and CPU time is improved. Only a small number of genetic iterations (298) is needed to reach the smallest errors (0,0092 for the average error and 0,0803 for the peak error). Fig. 2 shows the frequency responses of the different versions and Fig. 3 shows those of the classic methods.

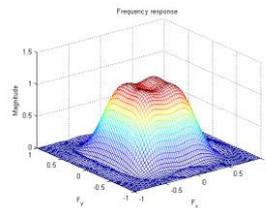


Fig.2.1. Adaptive GA<sub>1</sub>

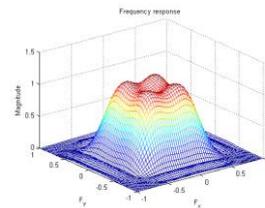


Fig.2.2. Adaptive GA<sub>2</sub>

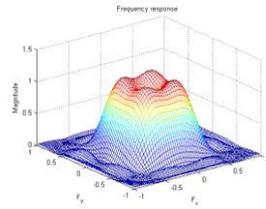


Fig.2.3. Adaptive GA<sub>3</sub>

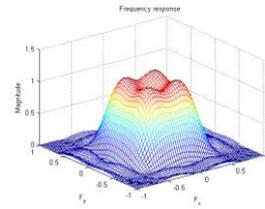


Fig.2.4. Adaptive GA<sub>4</sub>

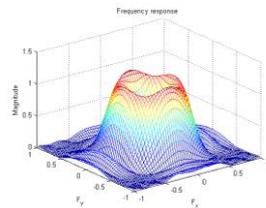


Fig.2.5. Adaptive GA<sub>5</sub>

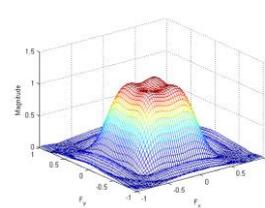


Fig.2.6. Standard GA<sub>6</sub>

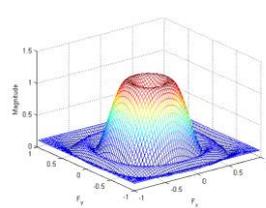


Fig.3.1. Frequency transformation method

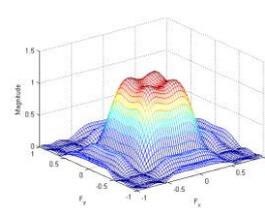


Fig.3.2. Frequency sampling method

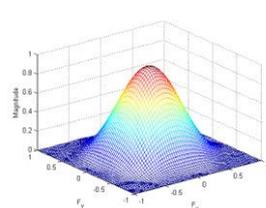


Fig.3.3. Windowing method

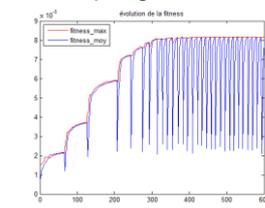


Fig.4. Fitness evolution for GA<sub>2</sub>

Except for GA<sub>5</sub>, the peak errors of the adaptive GAs can be considered significantly better than those of classic methods. Adaptive GA<sub>1</sub> has the best response for the passband region. The filters constructed using the Gas, the frequency transformation and frequency sampling methods have sharper transition band responses than that produced by the windowing method. For the stopband region, our GA<sub>1</sub> produces a better response. We can observe

specifically (Fig. 2,3) that the different versions of GAs have similar magnitude responses for the stopband regions. The frequency sampling method produces a filter whose response is a bit worse than the others. Supports of the filters designed by these algorithms are respected except for those designed by classic methods (frequency transformation method and windowing) and  $GA_5$ . We can observe that GAs are more time consuming than the classical ones. In Fig. 4, the effect of the couple “Breaking process” and “Adaptive mutation” is pointed out. We can observe a “sort of landing process”. Each time a sign of premature convergence is detected, the population is reseeded that affects the fitness average without losing the best chromosome. This speeds up the genetic advances and improves the final convergence. The contribution of the adaptive selection can be evaluated via the comparison between  $GA_1$  and  $GA_2$ .  $GA_1$  is better within most of attributes that underlines its interesting role. We can note that the diversification selection does not prevent the breaking process but shortens the population reseeding as the population is more diverse per nature.  $GA_1$  has been used for designing different filters under various specifications. Fig. 5 shows the frequency responses for three cases. We should note that the error in the designed filter response from the specified one is very small which promotes the role of adaptive GAs for 2-D FIR filter design.

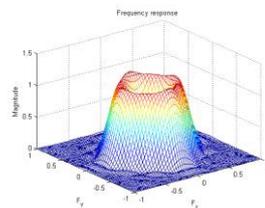


Fig.5.1. Band-pass 9\*9

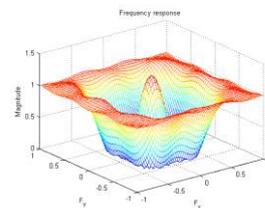


Fig.5.2. Band-cut 7\*7

## 5. Conclusion

This paper presents the optimal design methods of 2-D FIR digital filter based on GAs. By minimizing a quadratic measure of the error in the frequency band, real-valued solutions are evolved to get the filter coefficients. Our adaptive GA can produce filters with good response characteristics while greatly reducing the error criteria and CPU time. Future work will focus on the possibility of hybridizing the GA with other classic mathematical and genuine heuristic techniques.

## 6. References

- [1] D. E. Dudgeon and R. M. Mersereau, *Multidimensional Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1984.
- [2] D. Bhattacharya, A. Antoniou, "Design of 2-D FIR filters by a feedback neural network," *Multidimensional Systems and Signal Processing 10*, Boston, 1999, pp. 319-330.
- [3] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-time Signal Processing*, Prentice-Hall, Upper Saddle River, NJ, 1999.
- [4] Lim, Jae S., *Two-Dimensional Signal and Image Processing*, Englewood Cliffs, NJ, Prentice Hall, 1990.
- [5] Y.-H. Yang, J.-H. Lee, "Design of 2-D recursive digital filters using nonsymmetric half-plane allpass filters," *IEEE Transactions on Signal Processing 55*, 2007, pp. 5604–5618.
- [6] F.Ros, S.Guillaume, M.Pintore, J.R. Chrétien, "Hybrid Genetic Algorithm for Dual Selection," *Journal of Pattern Analysis and application (Theoretical Advances)*, Vol 11, 2008, pp. 179-198.
- [7] S-T Tzeng, "Genetic algorithm approach for designing 2-D FIR digital filters with 2-D symmetric properties," *Elsevier, Signal Processing 84*, 2004, pp.1883–1893.
- [8] H.-C. Lu, S.-T. Tzeng, "Design of arbitrary FIR log filter by genetic algorithm approach," *Elsevier, Signal Processing 80*, Netherlands, 2000, pp.497-505.
- [9] Shian-Tang Tzeng, "Design of 2-D FIR digital filters with specified magnitude and group delay responses by GA approach," *Signal Processing 87*, 2007, pp. 2036–2044.
- [10] Ling Cen, "A hybrid genetic algorithm for the design of FIR filters with SPoT coefficients," *Elsevier, Signal Processing 87*, 2007, pp. 528–540.

- [11] F.Ros, S.Guillaume, "An efficient nearest classifier," *Book Chapter of Hybrid Evolutionary Systems, Studies in Computational Intelligence*, Vol 75, Springer Verlag, 2007, pp. 131-147.
- [12] Coello CA, Van Veldhuizen, Lamont GB, *Evolutionary algorithms for solving multi-objective problems*, Kluwer, New York, 2002.
- [13] Boeringer D.W. Werner, D.H. Machuga, D.W., "A simultaneous parameter adaptation scheme for genetic algorithms with application to phased array synthesis," *IEEE Transactions on Antennas and Propagation*, Vol. 53, 2005, pp. 356-371.
- [14] K. Boudjelaba, D. Chikouche, F. Ros, "Evolutionary techniques for the synthesis of 2-D FIR filters," accepted in *IEEE Workshop on Statistical Signal Processing (SSP'11)*.