Design of Optimal Frequency-Selective FIR Filters Using a Memetic Algorithm

Luis M. San-José-Revuelta
Dept. Signal Theory and Communications,
Universidad de Valladolid
Valladolid, Spain
lsanjose@tel.uva.es

***Abstract*—Finite Impulse Response (FIR) digital filters are widely used due to their capabilities of low computational load, stability and linear phase. Most traditional design approaches can not control with enough accuracy the frequency response of the designed filter. For this reason, we propose the use of a memetic algorithm along with a weighted fitness function, to estimate optimized filter coefficients that best approximate ideal specifications. Results have been compared to both traditional methods (mainly windowing and the Parks-McClellan algorithm) as well as to several bio-inspired techniques. Numerical results show that proposed method achieves better fit to filter specifications, a larger attenuation in the stop band and a narrower transition band, at the expense of slightly increasing the passband ripple ($0.5-0.7$ dB), the later in about 68% of the cases.**

***Index Terms*—FIR design, memetic algorithm, genetic algorithm, k-opt algorithm**

I. Introduction

Filters are important components in most computing and electronic devices. Filtering aims to extract information about the signal of interest, either by eliminating noise, extracting frequency components or by separating desired components from unwanted signals.

FIR (Finite Impulse Response) filters present many advantages such as stability and the possibility of obtaining generalized linear phase. Like any linear time-invariant system, they are completely defined by knowing the output signal –known as *impulse response*– when an impulse signal is placed as input. The length of this output is known as the filter *order*. The larger this order is, the more similar becomes the filter response to the ideal filter. This increases computational load and processing time. Optimal filter design according to a specified criterion can be seen as an optimization problem [1].

Many conventional methods allow the design of FIR filters. For instance, the method of windowing [2], the Parks and McClellan (PM) algorithm [3], etc. However, these methods do not provide accurate control of various parameters such as the bands cut-off frequencies and the width of the transition band. Besides, methods based on classical optimization have a great tendency to get trapped in local minima, due, in part, to a wrong selection of initial conditions. Their slow rate of convergence also limits its practical use, particulary in online applications. Many novel design strategies, including bio-inspired methods, aim to overcome these drawbacks. These methods have several interesting properties, such as simplicity and the ability to handle different data coding and representations [4]. However, conventional evolutionary algorithms (EAs) are only capable of identifying the high performance region at an affordable time and present inherent difficulties in performing local search. To overcome these problems, once the high performance regions of the search space are identified by the EAs, it is useful to apply a local search procedure to optimize the members of the final population [4]. The efficiency of the algorithm and the quality of the solutions are well improved in this way. These hybrid schemes with closer analogy to cultural evolution are often categorized as memetic algorithms (MAs) [5].

In this paper, a memetic algorithm that combines a standard genetic algorithm (GA) with a k-opt heuristic local search scheme to improve convergence speed and solution accuracy, is applied to design FIR filters. The proposed algorithm employs a two-phase strategy where it starts with a GA that searches for the optimal region with limited computing time. Afterwards, a k-opt heuristic local search is employed to do the fine-tuning so as to obtain the exact global optimum –whenever previous GA has converged properly–, with light computational expense [6], [7].

The thus obtained MA improves the characteristics of the designed filter, showing a good trade-off between frequency response characteristics (mainly, ripple, attenuation and transition bandwidth) and processing time. Besides, it allows a better control of frequency features than most conventional techniques which aim to reduce implementation complexity [8].

II. Literature review

Nature-inspired methods so far used for FIR filter design belong to a vast variety of strategies. This section summarizes the state-of-the-art in FIR filter design using nature-inspired methods, focusing into two big categories: evolutionary algorithms and swarm methods.

Initial approaches made use of genetic algorithms [9] and some hybrid variants [10]. Simulations showed that GAs achieved better solutions, in terms of passband ripple and group delay, although stopband attenuation was slightly poorer.

ISBN 978-90-827970-1-5

Some hybrid methods were developed in order to reduce computational cost [11].

Differential Evolution (DE) algorithms have also been proposed [12]. Simulations show that performance was close to the GA, though computational load of DE was less than that of GA and much lower than the least squares method. In [13], Simulated Annealing (SA) was proposed for the design of both FIR and IIR filters. Results were quite good although computational load was very high. Tabu Search (TS) has also been proposed for FIR design [14].

Swarm methods such as the Ant-Colony (ACO) algorithm emerged as a need to find the global optimum in medium order filters [15]. A bee-colony algorithm was recently proposed by Dwivedi [16] and the algorithm of cloud of particles (Particle Swarm Optimization, PSO) was first proposed for low-pass filter design in [17]. Recently, the algorithm of Herd of Cats (CSO) was used in [18], and [19] developed a novel Bacteria Food Search algorithm (BFO), both of them for FIR design.

Even a few attempts using memetic algorithms for digital filter design have been proposed [20], [21].

III. Theoretical Foundation

A. Digital FIR filter design

As linear systems, a FIR filter is completely defined by giving either its transfer function $H(z)$, or its corresponding difference equation. The later can be written as:

$$y[n] = h[0]x[n] + h[1]x[n-1] + \cdots + h[N-1]x[n-(N-1)]$$

where $x[n]$ and $y[n]$ stand for the input and output signal, respectively, and N is the number of filter coefficients (thus, $N-1$ is known as the *order of the filter*). The purpose of digital filter design is to obtain the coefficients of the filter impulse response $h[n]$. The values of $h[n]$ will determine the type of frequency selective filter being designed: low-pass filter (LPF), high-pass filter (HPF), band-pass filter (BPF) or band-stop filter (BSF).

Filter specifications are often given in the frequency domain as, for example, the pass-band ripple (δ_p) and maximum attenuation (A_p), the bands cut-off frequencies, the minimum attenuation of the stop band (A_s), the stop-band ripple (δ_s), and the transition bandwidth (ΔF), among others.

This work studies the design of linear-phase symmetrical FIR filters. It is well-known that linear phase is guaranteed if the corresponding impulse response satisfies one of the following symmetry conditions [2]: $h[n] = h^*[-n]$ or $h[n] = -h^*[-n]$.

Due to this symmetry, only half of the coefficients need to be estimated. At the end of the design process, coefficients are duplicated to form the other half of the impulse response. Thus, complexity of the problem is reduced to half and the desired property of linear phase is satisfied.

The design process of an optimal filter might be seen as an optimization problem where some error measure must be minimized [1]. For instance, some error function between the ideal frequency response and the real frequency response of the estimated filter coefficients.

IV. Design Methodology

A. GA encoding and fitness function

This paper proposes the use of a MA, originally developed in [4] and here extended for FIR design. The first stage of the MA is a GA where each chromosome encodes a different filter impulse response. This is represented by vector $\mathbf{x}_i$:

$$\mathbf{x}_i^t = \left[h_{i,t}(1), h_{i,t}(2), h_{i,t}(3), \ldots, h_{i,t}\left(\frac{N+1}{2}\right)\right]$$

which represents the $(N+1)/2$ coefficients of the filter, taking advantage of the imposed symmetry; index i indicates a particular chromosome from the total population of PS ($1 \le i \le PS$), and t stands for current generation. The population will consist of a total of PS possible solutions.

This work proposes to use a fitness function (FF) defined as a combination of two previously proposed FFs. A first one, that we will refer as J_1, obtained in [18], [19], and a second one, J_2, taken from [17]. This way, the FF proposed for use in the MA is,

$$\begin{aligned} J &= w_1 J_1 + w_2 J_2 \\ &= w_1 \left(\max\left(|\mathbf{E}(\omega|_{\omega \le \omega_p}) - \delta_p|\right) + \max\left(|\mathbf{E}(\omega|_{\omega \ge \omega_p}) \right.\right. \\ &\quad \left.\left. - \delta_s|\right)\right) + w_2 \left(\sum_{\omega \le \omega_p} abs\left[abs(|\mathbf{H}_d(\omega)| - 1) - \delta_p\right] \right. \\ &\quad \left. + \sum_{\omega \ge \omega_s} abs(|\mathbf{H}_d(\omega)| - \delta_s) \right) \end{aligned} \tag{1}$$

where coefficients w_1 and w_2 are adjusted in order to get a filter with better frequency characteristics. Error function $\mathbf{E}(\omega)$ is defined in [3] as

$$\mathbf{E}(\omega) = \mathbf{G}(\omega)\left[\mathbf{H}_d(\omega) - \mathbf{H}_i(\omega)\right] \tag{2}$$

$$\mathbf{G}(\omega) = \begin{cases} \delta_s/\delta_p & 0 \le \omega \le \omega_p \\ \delta_p/\delta_s & \omega_s \le \omega \le \pi \end{cases} \tag{3}$$

where $\mathbf{G}(\omega)$ is a weighting function and ω_p and ω_s represent the pass- and stop-band limits, respectively. Notice that frequencies are normalized to sampling frequency. $H_d(\omega)$ represents the frequency response of the designed filter, and $H_i(\omega)$ represents the frequency response of the ideal desired filter. This way, error function $\mathbf{E}(\omega)$ is higher in that band where tolerances are narrower. Notice that $w_1 J_1$ weights the maximum errors obtained in pass- and stop-bands, measured with respect to maximum allowed ripples, δ_p and δ_s, respectively. This expression allows to differently weight errors in pass- and stop-bands, since $\mathbf{E}(\omega)$ depends on $\mathbf{G}(\omega)$, which is given by Eq. (3), where k allows to weight asymmetrically each filter band. On the other hand, term $w_2 J_2$ gathers together errors at every frequency within both pass- and stop-bands (and not only the maximum error values) and references them with respect to maximum deviations, δ_p and δ_s, respectively[1]. Finally, notice that error within transition bandwidth is taken into account in second term of J_1.

[1]This expression assumes an ideal amplitude value equal to 1 within passband, and equal to 0 within stopband.

ISBN 978-90-827970-1-5 © EURASIP 2018

B. Steps of the MA

Proposed MA combines a standard GA and a heuristic local search method. First, the GA estimates a potential quasi-optimum solution, then, the k-opt local search algorithm is applied to the GA fittest individual so as to quickly estimate a precise solution (always assuming that the GA has reported a solution estimate near the global optimum.

1) Genetic algorithm (Step 1): The metric given in Eq. (1) is used as the fitness function in the GA. In contrast to [4], the here proposed GA makes use of two genetic operators: mutation and crossover, though the later is applied with low probability. Mutation is implemented using the scheme proposed in [4], i.e., offspring vector o_i is calculated as

$$o_{ik} = sign(p_{ik} + N_k(0, \sigma)), \quad k = 1, 2, , K \tag{4}$$

where o_{ik} and p_{ik} denote the k-th element of the i-th offspring and parent chromosomes, respectively, $N_k(\mu, \sigma)$ represents a Gaussian random variable with mean μ and standard deviation σ. A new random number is generated for each value of k. Standard deviation controls the proximity of the offspring around the parents.

Next, crossover is performed with probability $P_c = 0.01$. The fittest PS individuals from the parents and offspring are selected to obtain next generation. GA stops when the algorithm starts converging. The heuristic local search of Step-2 is then called to fine-tune the best individual found so far.

2) k-opt Local Search (Step-2): Local search methods are improvement heuristics that search in the neighborhood of the current solution for a better one until no better solution is found. Neighborhood of a binary encoded vector is defined as the set of solutions which can be reached by changing the bits value within the vector.

The smallest neighborhood can be achieved by flipping a single bit of the solution vector. The so-called 1-opt neighborhood, searches for a flip with the highest associated gain in fitness ($g = f(\mathbf{b}') - f(\mathbf{b})$) in each iteration. An expression for the gain of flipping bit k in current solution is given in [4]. The k-opt neighborhood is obtained by flipping one up to k elements in the solution vector simultaneously:

$$N_{k-opt}(\mathbf{b}) = \mathbf{b}' \in S / d_H(\mathbf{b}, \mathbf{b}') \le k \tag{5}$$

where S is the solution space and d_H denotes the hamming distance between bit vectors. The k-opt neighborhood size, $|N_{k-opt}(\mathbf{b})| = \sum_{i=0}^{k} \binom{k}{i}$, grows exponentially with k.

To efficiently search a subset of the k-opt neighborhood, the main ideas of the Lin-Kernighan methods can be used [6], [7]. The main aim is to find a better solution by flipping a variable number of bits in the current solution in every iteration. A sequence of k solutions is produced in each iteration by flipping the bit with the highest associated gain. Every bit of the solution is only flipped once. The best solution from PS solutions is then selected as input for next iteration. Thus, a variable number of bits are flipped in each iteration to find a better solution in the neighborhood of current solution.

V. Results

This section shows the results of the simulations carried out with Matlab(R) and the effectiveness of the proposed filter design method. An odd number of samples is used for all filters designed in this work and the condition of symmetry of coefficients is assumed. Therefore, any type of filter (LPF, HPF, BPF and BSF) can be designed without restrictions. $N = 20$ unless otherwise specified. Results are mean values after 50 independent runs of the algorithm.

After several initial simulations, values selected for the different parameters were: *PS* = 30, maximum number of generations in GA *NaxGen* = 250, crossover probability $P_c = 0.2$, mutation probability $P_m = 0.01$, $w_1 = 0.25$ and $w_2 = 0.75$.

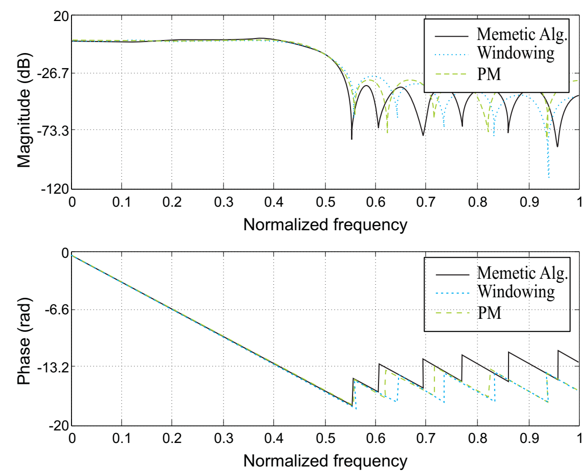

Fig. 1. Frequency response of the three compared 20-order low-pass filters: MA, PM [3] and windowing (Kaiser with $\beta = 2.48$) [2].

A. Low pass filter design

1) Fixed filter order N: The parameters of the desired LPF to be designed are: $F_s = 2$ Hz, $L = 512$, $(N-1) = 20$, $\delta_p = 0.1$, $\delta_s = 0.01$, normalized pass-band cut-off frequency $\omega_p = 0.45$, stop-band normalized cut-off frequency $\omega_s = 0.55$, and $\Delta F = 0.1$.

This LPF design will be performed using the following methods: windowing [2], PM [3], and MA. In Fig. 1 frequency responses are depicted. Notice that MA gets the filter with the narrowest ΔF, clearly outperforming the other two methods. In contrast, MA slightly increases δ_p.

The filter designed with MA presents higher A_s than those obtained with PM and windowing. Notice the linear phase of the three methods in the pass-band.

Table I shows the results in terms of maximum ripples δ_p and δ_s, minimum A_s, and ΔF. It can be observed that, for the case of windowing, if a reduced ΔF is required, attenuation A_s must be reduced (lower β). This fact involves that δ_p increases, as well as the maximum attenuation in this band. Also, MA implemented with proposed FF provides a better frequency response than PM, in terms of ΔF and A_s. However, it slightly increases δ_p and, therefore, the maximum attenuation produced in this band.

ISBN 978-90-827970-1-5 © EURASIP 2018

TABLE I
TRANSITION BANDWIDTH, MAXIMUM PASSBAND AND STOPBAND RIPPLES, AND STOPBAND ATTENUATION FOR A 20-ORDER LPF DESIGNED WITH MA, PM [3] AND WINDOWING [2]. VALUES OBTAINED AFTER 50 INDEPENDENT RUNS.

Method used	Max. δ_p $(0-0.45\pi)$		Max. δ_s $(0.55\pi-\pi)$		Min. A_s (dB)		Transition band (ΔF)	
	Max.	Min. $\pm$ s.d.	Max.	Mean $\pm$ s.d.	Max.	Mean $\pm$ s.d.	Max.	Mean $\pm$ s.d.
PM	0.025	0.025 ± 0	0.025	0.025 ± 0	32.03	32.03 ± 0	0.102	0.102 ± 0
Windowing (Kaiser,$\beta = 2.48$)	0.031	0.031 ± 0	0.018	0.018 ± 0	34.68	34.68 ± 0	0.121	0.121 ± 0
Windowing (Kaiser,$\beta = 1.7$)	0.052	0.052 ± 0	0.034	0.034 ± 0	29.38	29.38 ± 0	0.098	0.098 ± 0
MA	0.163	0.145 ± 0.01	0.018	0.016 ± 0.1	36.82	36.98 ± 0.16	0.089	0.095 ± 0.003

2) Variable order: Let us see what happens if both filter order and transition bandwidth are modified.

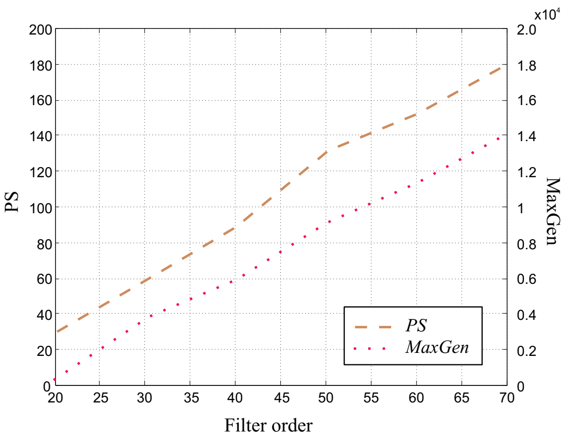

Fig. 2. Population size (*PS*) and number of iterations (*MaxGen*) required for MA convergence for different orders of a low pass filter.

Fig. 2 shows the relationship between order N and both the population size and the number of iterations required for the design. This figure is valid as a guide for parameter tuning for HPFs and BPFs, as well. BSFs require an increase of $\sim 33\%$ in both parameters.

3) Comparison with other nature-inspired algorithms: Once the comparison with traditional methods has been made, comparisons to other nature-inspired algorithms will be performed.

Fig. 3 shows the frequency responses of MA as well as BFO, PSO and RGA [19]. MA shows the best A_s without an increase of ΔF. In the meanwhile, δ_p is very similar to those obtained with other design methods. If we look at the phase of the designed filters, all of them present a linear phase in the pass-band, in accordance to the properties of symmetrical FIR filters described in Section III.

For the shake of brevity, results concerning the high-pass filter have not been included here. However, similar conclusions to those of the low-pass filter are valid.

B. Band-pass filter (BPF) and band-stop filter (BSF)

This section gathers the results related to both BPF and BSF filters. Common specifications are: $F_s = 2$ Hz, $L = 512$, $(N-1) = 20$, $\delta_p = 0.1$, $\delta_s = 0.01$. Cut-off frequencies for BPF: inferior stop-band cut-off frequency $\omega_{s1} = 0.25$, inferior pass-band cut-off frequency $\omega_{p1} = 0.35$, upper pass-band frequency $\omega_{p2} = 0.65$, upper stop-band cutoff frequency $\omega_{s2} = 0.75$. For BSF: $\omega_{p1} = 0.25$, $\omega_{s1} = 0.35$, $\omega_{s2} = 0.75$, $\omega_{p2} = 0.85$. BSF is implemented with *PS*=50 and *MaxGen*=300, while BSF requires *PS*=60 and *MaxGen*=500.

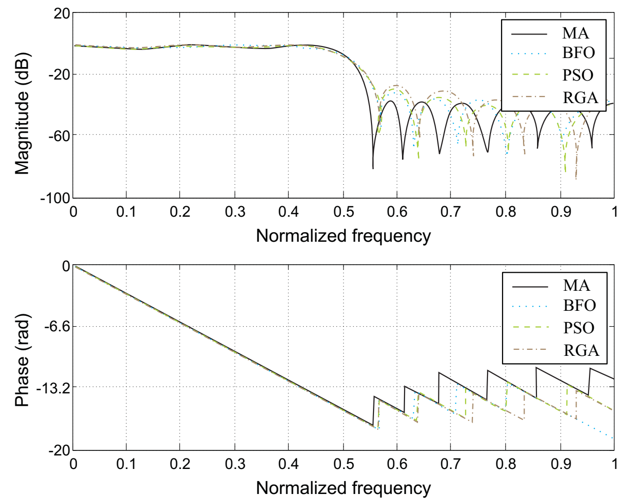

Fig. 3. Frequency response, magnitude (dB) and phase (rad), for a 20-order low pass filters designed with different nature-inspired algorithms: MA (proposed), BFO, PSO and RGA [19].

Like in previous sections, MA is first compared to traditional approaches. Fig. 4 shows frequency responses of the three methods for stopband filters. It can be seen that the method with highest A_s is the proposed MA, with a ΔF very similar to the rest of the methods, being, on the contrary, worse in terms of δ_p, which is slightly higher than the ones retrieved using windowing and PM.

Numerical simulations show that MA with J gets a larger A_s as well as a narrower ΔF. This does not occur with traditional techniques where, despite having a lower attenuation, ΔF is much higher. However, δ_p is much lower. Similar conclusions are valid for BSFs.

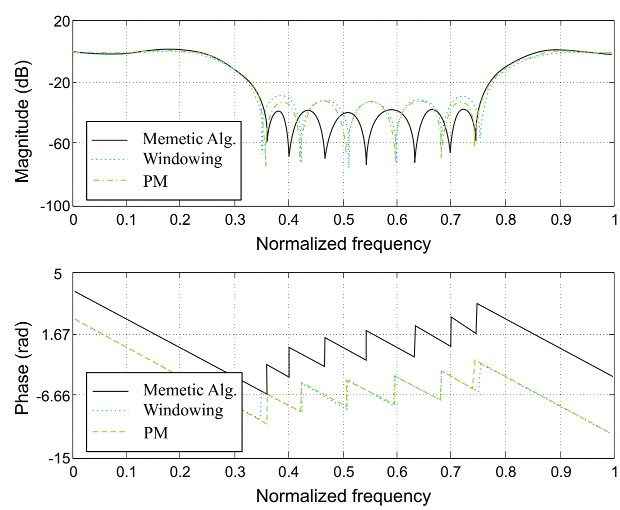

Fig. 4. Frequency response, magnitude (in dB) and phase (in rad), of the BSFs obtained using MA, PM [3] and windowing (Kaiser with $\beta = 1.9$) [2].

Table II compares frequency parameters with those obtained with other nature-inspired algorithms (BFO and CSO [19]). MA clearly outperforms the other approaches since it achieves the largest A_s, the smallest δ_p and δ_s, while keeping transition bandwidth within specs tolerance.

TABLE II
FREQUENCY FEATURES OF BAND-STOP FILTERS OBTAINED WITH DIFFERENT NATURE-INSPIRED DESIGN METHODS IN COMPARISON TO THOSE OBTAINED WITH PROPOSED MA. DESIGN OF A 20-ORDER BSF.

Design Method	Max. A_s (dB)	Max. δ_p	Max. δ_s	ΔF
BFO [19]	33.84	0.161	0.02033	0.1080
CSO [18]	34.47	0.163	0.01891	0.1006
MA	37.91	0.1499	0.01381	0.094

This way, proposed MA with FF J given in Eq. (1), consolidates as a good alternative design method to other strategies for any type of frequency-selective digital filter.

VI. CONCLUSIONS

This paper explores the possibility of using a memetic algorithm for designing FIR filters. The proposed memetic algorithm consists in two stages. First, a GA is applied for determining the region of the solutions space where the optimal solution is, and, afterwards, a local search technique based on the k-opt algorithm is applied.

MA has been compared to both traditional methods for filter design (windowing [2], the minimum squares method and the algorithm of Parks-McClellan [3]), and several nature-inspired methods, mainly CSO [19], [18], RGA [19], BFO [19] and PSO [22].

Numerical results show that MA improves the frequency characteristics of filters obtained by conventional techniques. The frequency response complies with enough accuracy specifications of transition bandwidth and cut-off frequencies of stop and pass bands, giving priority to the attenuation of the bands over the width of transition bands. MA obtained a greater attenuation of the stopband (4-6 dB higher) in comparison to traditional methods, reducing transition band width, as well. However, a slight increase in the passband ripple is observed in about 68% of cases, and, therefore, an increment in the attenuation of that band around 0.5-0.7 dB.

On the other hand, when MA is compared to other nature-inspired techniques it achieves satisfactory results, as well. Simulations carried out indicate that the proposed method obtains a better performance than other techniques in terms of frequency parameters: a greater attenuation of the stop-bands, a suitable transition bandwidth and a pass-band ripple very close to those obtained with other similar techniques.

REFERENCES

[1] Z. Milivojevic, *Digital filter design*. MikroElektronika. Pleiades Publishing. Russia., 2009.

[2] A. V. Oppenheim, *Discrete-time signal processing*. Pearson Education India, 1999.

[3] T. Parks and J. McClellan, "Chebyshev approximation for nonrecursive digital filters with linear phase," *IEEE Trans. on Circuit Theory*, vol. 19, no. 2, pp. 189–194, 1972.

[4] H. S. Lim and B. Venkatesh, "An effective memetic algorithm for the optimum multiuser detection problem," in *Spread Spectrum Techniques and Applications, 8th Int. Symp. on*. IEEE, 2004, pp. 885–889.

[5] P. Moscato, "Memetic algorithms: A short introduction, new ideas in optimization, d," *Corne, M. and F. Glover, McGrawHill, London*, 1999.

[6] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *The Bell system technical journal*, vol. 49, no. 2, pp. 291–307, 1970.

[7] S. Lin and B. W. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Operations research*, vol. 21, no. 2, pp. 498–516, 1973.

[8] B. Luitel and G. K. Venayagamoorthy, "Differential evolution particle swarm optimization for digital filter design," in *Proc. of the 2008 IEEE Congress on Evolutionary Computation*, 2008, pp. 3954–3961.

[9] D. Suckley, "Ga in the design of fir filters," *IEE Proceedings G (Circuits, Dev. and Syst.)*, vol. 138, no. 2, pp. 234–238, 1991.

[10] S. Ahmad, "Design of digital filters using genetic algorithms," *PhD Thesis. University of Victoria*, 2008.

[11] K. Boudjelaba, F. Ros, and D. Chikouche, "An efficient hybrid genetic algorithm to design finite impulse response filters," *Expert Systems with Applications*, vol. 41, no. 13, pp. 5917–5937, 2014.

[12] Q. Zhao and G. Meng, "Design of digital fir filters using differential evolution algorithm based on reserved gene," in *Proc. IEEE Int. Conf. on Inform. Science and Manag. Engineering*, 2010, pp. 177–180.

[13] R. Kacelenga, P. Graumann, and L. Turner, "Designing digital fir filters using tabu search algorithm," in *Proc. of 1990 IEEE Int. Symp. on Circuits and Systems*, 1990, pp. 642–645.

[14] K. Watcharasitthiwat, J. Koseeyaporn, and P. Wardkein, "Designing digital fir filters using multiple tabu search algorithm," in *Proc.IEEE Int. Conf. on Comm., Circuits and Systems*, 2006, pp. 171–175.

[15] S. Tsutsumi and K. Suyama, "Design of fir filters with discrete coefficients using ant colony optimization," *IEEJ Trans. on Electronics, Information and Systems*, vol. 132, pp. 1066–1071, 2012.

[16] A. K. Dwivedi, S. Ghosh, and N. D. Londhe, "Low power fir filter design using modified multi-objective artificial bee colony algorithm," *Engineering Applications of Artificial Intell.*, vol. 55, pp. 58–69, 2016.

[17] J. I. Ababneh and M. H. Bataineh, "Linear phase fir filter design using particle swarm optimization and genetic algorithms," *Digital Signal Processing*, vol. 18, no. 4, pp. 657–668, 2008.

[18] S. P. Ghoshal, S. K. Saha, R. Kar, and D. Mandal, "Cat swarm optimization algorithm for optimal linear phase fir filter design," *ISA Transactions*, vol. 52, no. 6, pp. 781–794, 2013.

[19] S. K. Saha, R. Kar, D. Mandal, and S. P. Ghoshal, "Bacteria foraging optimisation algorithm for optimal fir filter design," *Int. J. of Bio-Inspired Computation*, vol. 5, no. 1, pp. 52–66, 2013.

[20] C. Liu, G. Zhang, X. Zhang, and H. Liu, "A memetic algorithm based on p systems for iir digital filter design," in *Dependable, Autonomic and Secure Computing, 2009. DASC'09. 8th IEEE Int. Conf. on*, 2009, pp. 330–334.

[21] Y. Wang, B. Li, and T. Weise, "Two-stage ensemble memetic algorithm: Function optimization and digital iir filter design," *Information Sciences*, vol. 220, pp. 408–424, 2013.

[22] M. Najjarzadeh and A. Ayatollahi, "Fir digital filters design: particle swarm optimization utilizing lms and minimax strategies," in *Proc. 8th IEEE Int. Symp. on Sig. Proc. and Inform. Technol.*, 2008, pp. 129–132.