Journal of King Saud University – Computer and Information Sciences (2012) 24, 83–92

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

ORIGINAL ARTICLE

Design of optimal linear phase FIR high pass filter using craziness based particle swarm optimization technique

Sangeeta Mandal [a], Sakti Prasad Ghoshal [a], Rajib Kar [b], Durbadal Mandal [b,*]

[a] Dept. of Electrical Engg., National Institute of Technology, Durgapur, India
[b] Dept. of Electronics & Communication Engg., National Institute of Technology, Durgapur, India

Received 31 May 2011; revised 13 September 2011; accepted 19 October 2011
Available online 7 November 2011

KEYWORDS

FIR filter;
PM algorithm;
RGA;
PSO;
Magnitude response;
High Pass (HP) filter

Abstract In this paper, an optimal design of linear phase digital high pass FIR filter using Craziness based Particle Swarm Optimization (CRPSO) approach has been presented. FIR filter design is a multi-modal optimization problem. The conventional gradient based optimization techniques are not efficient for such multi-modal optimization problem as they are susceptible to getting trapped on local optima. Given the desired filter specifications to be realized, the CRPSO algorithm generates a set of optimal filter coefficients and tries to meet the desired specifications. In birds' flocking or fish schooling, a bird or a fish often changes directions suddenly. This is described by using a "craziness" factor and is modeled in the CRPSO technique. In this paper, the realizations of the CRPSO based optimal FIR high pass filters of different orders have been performed. The simulation results have been compared to those obtained by the well accepted classical optimization algorithm such as Parks and McClellan algorithm (PM), and evolutionary algorithms like Real Coded Genetic Algorithm (RGA), and conventional Particle Swarm Optimization (PSO). The results justify that the proposed optimal filter design approach using CRPSO outperforms PM, RGA and PSO, in the optimal characteristics of frequency spectrums.

1. Introduction

Digital filters are used in numerous applications from control systems, systems for audio and video processing, and communication systems to systems for medical applications to name just a few. They can be implemented in hardware or software and can process both real-time and off-line (recorded) signals. Beside the inherent advantages, such as, high accuracy and reliability, small physical size, and reduced sensitivity to component tolerances or drift, digital filters allow one to achieve certain characteristics not possible with analog implementations such as exact linear phase and multi-rate operation. Digital filtering can be applied to very low frequency signals, such as those occurring in biomedical and seismic applications very

* Corresponding author. Tel.: +91 9474119721.
E-mail addresses: spghoshalnitdgp@gmail.com (S.P. Ghoshal), durbadal.bittu@gmail.com (D. Mandal).

Peer review under responsibility of King Saud University.
doi:10.1016/j.jksuci.2011.10.007

Production and hosting by Elsevier

efficiently. In addition, the characteristics of digital filters can be changed or adapted by simply changing the content of a finite number of registers, thus multiple filters are usually used to discriminate a frequency or a band of frequencies from a given signal which is normally a mixture of both desired and undesired signals. The undesired portion of the signal commonly comes from noise sources such as power line hum or other signals which are not required for the current application. There are mainly two types of filter algorithms. They are Finite Impulse Response filter (FIR), Infinite Impulse Response filter (IIR). In case of a FIR filter, the response due to an impulse input will decay within a finite time. But for IIR filter, the impulse response never dies out. It theoretically extends to infinity. FIR filters are commonly known as non-recursive filters and IIR filters are known as recursive filters. Implementation of FIR filters is easy, but it is slower when compared to IIR filters. Though IIR filters are fast, practical implementation is complicated compared to FIR filters (Litwin, 2000). FIR filter is an attractive choice because of the ease in design and stability. By designing the filter taps to be symmetrical about the centre tap position, the FIR filter can be guaranteed to have linear phase. FIR filters are known to have many desirable features such as guaranteed stability, the possibility of exact linear phase characteristic at all frequencies and digital implementation as non-recursive structures. Traditionally, different techniques exist for the design of FIR filters and its implementation (Yuksel et al., 2003; Filho et al., 2000). Out of these, windowing method is the most popular (Yuksel et al., 2003). In this method, ideal impulse response is multiplied with a window function. There are various kinds of window functions (Butterworth, Chebyshev, Kaiser, etc.), depending on the requirements of ripples on the pass band and stop band, stop band attenuation and the transition width. These various windows limit the infinite length impulse response of ideal filter into a finite window to design an actual response. But windowing methods do not allow sufficient control of the frequency response in the various frequency bands and other filter parameters such as transition width. The most frequently used method for the design of exact linear phase weighted Chebyshev FIR digital filter is the one based on the Remez-exchange algorithm proposed by Parks and McClellan (Parks and McClellan, 1972). Further improvements in their results have been reported in McClellan et al. (1973), Rabiner (1973).

The classical gradient based optimization methods are not suitable for FIR filter optimization because of the following reasons: (i) highly sensitive to starting points when the number of solution variables and hence the size of the solution space increase, (ii) frequent convergence to local optimum solution or divergence or revisiting the same suboptimal solution, (iii) requirement of continuous and differentiable objective cost function (gradient search methods), (iv) requirement of the piecewise linear cost approximation (linear programming), and (v) problem of convergence and algorithm complexity (non-linear programming). So, evolutionary methods have been employed in the design of digital filters to design with better parameter control and to better approximate the ideal filter. Different heuristic optimization algorithms such as simulated annealing algorithms (Chen et al., 2000), genetic algorithm (GA) Mastorakis et al., 2003 have been widely used to the synthesis of design methods capable of satisfying constraints which would be unattainable. When considering global optimization methods for digital filter design, the GA seems to be the promising one. Filters designed by GA have the potential of obtaining near global optimum solution. Although standard GA (herein referred to as Real Coded GA (RGA)) has a good performance for finding the promising regions of the search space, but finally, RGA is prone to revisiting the same suboptimal solutions.

The approach detailed in this paper takes advantage of the power of the stochastic global optimization technique called particle swarm optimization. Although the algorithm is adequate to applications in any kind of parameterized filters, it is chosen to focus on real-coefficient FIR filters. Particle Swarm Optimization (PSO) is an evolutionary algorithm developed by Eberhart et al. Kennedy and Eberhart (1995), Eberhart and Shi (2000). Several attempts have been made towards the optimization of the FIR Filter (Ababneh and Bataineh, 2008; Luitel and Venayagamoorthy, 2008; Sarangi et al., 2011) using PSO algorithm. The PSO is simple to implement and its convergence may be controlled via few parameters. The limitations of the conventional PSO are that it may be influenced by premature convergence and stagnation problem (Ling et al., 2008; Biswal et al., 2009). In order to overcome these problems, the PSO algorithm has been modified and called as craziness based PSO (CRPSO) in this paper and is employed for FIR filter design.

This paper describes the FIR HP digital filter design using CRPSO. CRPSO algorithm tries to find the best coefficients that closely match the desired frequency response. Based upon this improved PSO approach, this paper presents a good and comprehensive set of results, and states arguments for the superiority of the algorithm.

The rest of the paper is arranged as follows. In Section 2, the FIR filter design problem is formulated. Section 3 briefly discusses on RGA, conventional PSO and the proposed CRPSO algorithm. Section 4 describes the simulation results obtained for FIR HP digital filter using RGA, PSO, PM algorithm and the proposed CRPSO. Finally, Section 5 concludes the paper.

2. High pass FIR filter design

A digital FIR filter is characterized by,

$$H(z) = \sum_{n=0}^{N} h(n)z^{-n}, \quad n = 0, 1 \dots N \tag{1}$$

where N is the order of the filter which has $(N + 1)$ number of coefficients. $h(n)$ is the filter's impulse response. The values of $h(n)$ will determine the type of the filter e.g. low pass, high pass, band pass, etc. The values of $h(n)$ are to be determined in the design process and N represents the order of the polynomial function. This paper presents the optimal design of even order HP filter with even symmetric $h(n)$ coefficients. The length of $h(n)$ is $N + 1$ and the number of coefficients is also $N + 1$. In the optimization algorithm, the individual represents $h(n)$. In each iteration, a population of such individuals is updated based on updated error fitnesses. Error fitness is the error between the frequency responses of the ideal and the actual filters. An ideal filter has a magnitude of one on the pass band and a magnitude of zero on the stop band. Comparative optimization is done using RGA, conventional PSO and CRPSO.

The individuals that have lower error fitness values represent the better filter i.e., the filter with better frequency response. Result obtained after a certain number of iterations or after the error is below a certain limit is considered to be the optimal result. Because the coefficients are symmetrical, the dimension of the problem reduces by a factor of 2. The $(N + 1)/2$ coefficients are then flipped and concatenated to find the required $N + 1$ coefficients.

Various filter parameters which are responsible for the optimal filter design are the stop band and pass band normalized frequencies (ω_s, ω_p), the pass band and stop band ripples (δ_p, δ_s), the stop band attenuation and the transition width. These parameters are mainly decided by the filter coefficients, which is evident from transfer function in (1).

Several scholars have investigated and developed algorithms in which N, δ_p, and δ_s are fixed while the remaining parameters are optimized (Herrmann and Schussler, 1970). Other algorithms were originally developed by Parks and McClellan (PM) in which N, w_p, w_s, and the ratio δ_p/δ_s are fixed (Parks and McClellan, 1972). In this paper, swarm and evolutionary optimization algorithms are applied in order to obtain the actual filter response as close as possible to the ideal response.

Now for (1), coefficient vector $\{h_0, h_1, \ldots, h_N\}$ is represented in $N + 1$ dimensions. The particles are distributed in a D dimensional search space, where $D = N + 1$ for the case of FIR filter.

The frequency response of the FIR digital filter can be calculated as,

$$H(e^{jw_k}) = \sum_{n=0}^{N} h(n)e^{-jw_k n} \quad (2)$$

where $w_k = \frac{2\pi k}{N}$; $H(e^{jw_k})$ is the Fourier transform complex vector. This is the FIR filter frequency response. The frequency is sampled in $[0, \pi]$ with N points; the positions of the particles in this D dimensional search space represent the coefficients of the transfer function. In each iteration of evolutionary optimization, these particles find new positions, which are the new sets of coefficients.

An error fitness function given by (3) is the approximate error used in Parks–McClellan algorithm for filter design (Parks and McClellan, 1972).

$$E(\omega) = G(\omega)[H_d(e^{j\omega}) - H_i(e^{j\omega})] \quad (3)$$

where $G(\omega)$ is the weighting function used to provide different weights for the approximate errors in different frequency bands, $H_d(e^{j\omega})$ is the frequency response of the desired HP filter and is given as,

$$\begin{aligned} H_d(e^{j\omega_k}) &= 0 \quad \text{for } 0 \leqslant \omega \leqslant \omega_c; \\ &= 1 \quad \text{otherwise} \end{aligned} \quad (4)$$

$H_i(e^{j\omega})$ is the frequency response of the approximate filter.

$$\begin{aligned} H_d(\omega) &= [H_d(\omega_1), H_d(\omega_2), H_d(\omega_3), \ldots H_d(\omega_K)]^T \quad \text{and} \quad H_i(\omega) \\ &= [H_i(\omega_1), H_i(\omega_2), H_i(\omega_3) \ldots, H_i(\omega_K)]^T \end{aligned}$$

The major drawback of PM algorithm is that the ratio of δ_p/δ_s is fixed. To improve the flexibility in the error fitness function to be minimized, so that the desired level of δ_p and δ_s may be specified, the error fitness function given in (5) has been considered as fitness function in many literatures (Ababneh and Bataineh, 2008, Sarangi et al., 2011).

The error to be minimized is defined as:

$$J_1 = \max_{\omega \leqslant \omega_p}(|E(\omega)| - \delta_p) + \max_{\omega \geqslant \omega_s}(|E(\omega)| - \delta_s) \quad (5)$$

where δ_p and δ_s are the ripples in the pass band and stop bands, respectively, and ω_p and ω_s are pass band and stop band normalized edge frequencies, respectively. Eq. (5) represents the error fitness function to be minimized using the evolutionary algorithms. The algorithms try to minimize this error. Since the coefficients of the linear phase filter are matched, the dimension of the problem is halved. By only determining one half of the coefficients, the filter may be designed. This greatly reduces the complexity of the algorithms.

3. Evolutionary techniques employed

3.1. Real Coded Genetic Algorithm (RGA)

Real Coded Genetic Algorithm (RGA) is mainly a probabilistic search technique, based on the principles of natural selection and evolution. At each generation, it maintains a population of individuals where each individual is a coded form of a possible solution of the problem at hand called chromosome. Chromosomes are constructed over some particular alphabet, e.g., the binary alphabet $\{0, 1\}$, so that chromosomes' values are uniquely mapped onto the real decision variable domain. Each chromosome is evaluated by a function known as fitness function, which is usually the objective func-

Table 1 Steps for RGA.

Step 1	Initialize the real chromosome strings of n_p population, each consisting of a set of HP filter coefficients. Size of the set depends on the number of the filter coefficients for a particular order of the filter to be designed
Step 2	Decoding the strings and evaluation of error of each string
Step 3	Selection of elite strings in order of increasing error fitness values from the minimum value
Step 4	Copying the elite strings over the non-selected strings
Step 5	Crossover and mutation generate the off-springs
Step 6	Genetic cycle updating
Step 7	The iteration stops when the maximum number of cycles is reached. The grand minimum error fitness and its corresponding chromosome string or the desired solution of optimal $h(n)$ coefficients of HP filter are finally obtained

tion of the corresponding optimization problem (Mandal and Ghoshal, 2010; Mandal et al., 2010).

The basic steps of RGA are shown in Table 1.

3.2. Particle Swarm Optimization (PSO)

PSO is a flexible, robust population-based stochastic search/optimization technique with implicit parallelism, which can easily handle with non-differential objective functions, unlike traditional optimization methods (Mandal et al., 2010,2009). PSO is less susceptible to getting trapped on local optima unlike GA, Simulated Annealing, etc. Eberhart et al. Kennedy and Eberhart (1995), Eberhart and Shi (2000) developed PSO concept similar to the behavior of a swarm of birds. PSO is developed through simulation of bird flocking in multidimensional space. Bird flocking optimizes a certain objective function. Each particle (bird) knows its best value so far (*pbest*). This information corresponds to personal experiences of each particle. Moreover, each particle knows the best value so far in the group (*gbest*) among *pbests*. Namely, each particle tries to modify its position using the following information:

- The distance between the current position and the *pbest*.
- The distance between the current position and the *gbest*.

Similar to GA, in PSO techniques also, real-coded particle vectors of population n_p are assumed. Each particle vector consists of components as required number of normalized HP filter coefficients, depending on the order of the filter to be designed.

Mathematically, velocities of the particle vectors are modified according to the following equation:

$$V_i^{(k+1)} = w * V_i^{(k)} + C_1 * rand_1 * \left(pbest_i^{(k)} - S_i^{(k)}\right) + C_2 * rand_2 * \left(gbest^{(k)} - S_i^{(k)}\right) \quad (6)$$

where $V_i^{(k)}$ is the velocity of ith particle at kth iteration; w is the weighting function; C_1 and C_2 are the positive weighting factors; $rand_1$ and $rand_2$ are the random numbers between 0 and 1; $S_i^{(k)}$ is the current position of ith particle vector at kth iteration; $pbest_i^{(k)}$ is the personal best of ith particle vector at kth iteration; $gbest^{(k)}$ is the group best of the group at kth iteration. The searching point in the solution space may be modified by the following equation:

$$S_i^{(k+1)} = S_i^{(k)} + V_i^{(k+1)} \quad (7)$$

The first term of (6) is the previous velocity of the particle vector. The second and third terms are used to change the velocity of the particle. Without the second and third terms, the particle will keep on "flying" in the same direction until it hits the boundary. Namely, it corresponds to a kind of inertia represented by the inertia constant, w and tries to explore new areas.

3.3. Craziness based Particle Swarm Optimization (CRPSO)

The global search ability of above discussed conventional PSO is improved with the help of the following modifications. This modified PSO is termed as Craziness based Particle Swarm Optimization (CRPSO).

The velocity in this case can be expressed as follows (Mandal and Ghoshal, 2010):

$$V_i^{(k+1)} = r_2 * sign(r_3) * V_i^{(k)} + (1 - r_2) * C_1 * r_1 * \left\{pbest_i^{(k)} - S_i^{(k)}\right\} + (1 - r_2) * C_2 * (1 - r_1) * \left\{gbest^{(k)} - S_i^{(k)}\right\} \quad (8)$$

where r_1, r_2 and r_3 are the random parameters uniformly taken from the interval [0, 1] and $sign(r_3)$ is a function defined as:

$$\begin{aligned} sign(r_3) &= -1 \quad \text{where } r_3 \leqslant 0.05 \\ &= 1 \quad \text{where } r_3 > 0.05 \end{aligned} \quad (9)$$

The two random parameters $rand_1$ and $rand_2$ of (6) are independent. If both are large, both the personal and social experiences are over used and the particle is driven too far away from the local optimum. If both are small, both the personal and social experiences are not used fully and the convergence speed of the technique is reduced. So, instead of taking independent $rand_1$ and $rand_2$, one single random number r_1 is chosen so that when r_1 is large, $(1 - r_1)$ is small and vice versa. Moreover, to control the balance between global and local searches, another random parameter r_2 is introduced. For birds' flocking for food, there could be some rare cases that after the position of the particle is

Table 2 Steps of CRPSO.

Step 1:	Initialization: Population (swarm size) of particle vectors, $n_P = 120$; maximum iteration cycles = 200; number of filter coefficients ($h(n)$), filter order, $nvar$ = 20 or 30 or 40; fixing values of C_1, C_2 as 2.05; $P_{cr} = 0.3$; $v^{craziness} = 0.0001$; minimum and maximum values of filter coefficients, $h_{\min} = -2$, $h_{\max} = 2$; number of samples = 128; $\delta_p = 0.1$, $\delta_s = 0.01$; initialization of the velocities of all the particle vectors
Step 2:	Generate initial particle vectors of filter coefficients ($nvar/2 + 1$) randomly within limits; computation of initial error fitness values of the total population, n_P
Step 3:	Computation of population based minimum error fitness value and computation of the personal best solution vectors (*hpbest*), group best solution vector (*hgbest*)
Step 4:	Updating the velocities as per (8) and (10); updating the particle vectors as per (7) and checking against the limits of the filter coefficients; finally, computation of the updated error fitness values of the particle vectors and population based minimum error fitness value
Step 5:	Updating the *hpbest* vectors, *hgbest* vector; replace the updated particle vectors as initial particle vectors for step 4
Step 6:	Iteration continues from step 4 till the maximum iteration cycles or the convergence of minimum error fitness values; finally, *hgbest* is the vector of optimal FIR HP filter coefficients ($nvar/2 + 1$); form complete *nvar* coefficients by copying (because the filter has linear phase) before getting the optimal frequency spectrum

Table 3 RGA, PSO, CRPSO Parameters.

Parameters	RGA	PSO	CRPSO
Population size	120	120	120
Iteration cycles	700	600	200
Crossover rate	0.8	–	–
Crossover	Two point crossover	–	–
Mutation rate	0.001	–	–
Selection probability	1/3	–	–
C_1	–	2.05	2.05
C_2	–	2.05	2.05
$v_i^{\min}$	–	0.01	0.01
$v_i^{\max}$	–	1.0	1.0
$w_{\max}$	–	1.0	–
$w_{\min}$	–	0.4	–
P_{cr}	–	–	0.3
$v_{\text{craziness}}$	–	–	0.0001

Table 4 Optimized coefficients of FIR HP filter of order 20.

$h(n)$	RGA	PSO	CRPSO
$h(1) = h(21)$	−0.041381962615633	−0.025121713619127	−0.020824324104180
$h(2) = h(20)$	0.021673804280370	0.001492366452015	−0.011047898251996
$h(3) = h(19)$	0.047895764196488	0.043733075825729	0.042748242336850
$h(4) = h(18)$	−0.013229952769354	0.011335425063877	0.017065576110215
$h(5) = h(17)$	−0.043587982592581	−0.073252559596244	−0.062020554590447
$h(6) = h(16)$	−0.038782190560531	−0.015345172678055	−0.026716425667514
$h(7) = h(15)$	0.072174643529098	0.054576040647757	0.057963246158695
$h(8) = h(14)$	0.079085537629130	0.086223797404124	0.077835011483033
$h(9) = h(13)$	−0.057763146408972	−0.053614279222201	−0.040945127266925
$h(10) = h(12)$	−0.317157761265512	−0.319103050490381	−0.335967934822772
$h(11)$	0.575150960310068	0.575150960310068	0.575150960310068

Table 5 Optimized coefficients of FIR HP filter of order 30.

$h(n)$	RGA	PSO	CRPSO
$h(1) = h(31)$	−0.023490271736148	−0.021936936290133	−0.016668419495247
$h(2) = h(30)$	0.009665423079993	0.009894574634720	−0.001729383539829
$h(3) = h(29)$	0.023468772204978	0.024447574147984	0.015773136968762
$h(4) = h(28)$	−0.004912075133580	0.003141365867601	0.008953869670687
$h(5) = h(27)$	−0.011199071100982	−0.026453238596167	−0.008216024088105
$h(6) = h(26)$	−0.036859205525860	−0.016376104143824	−0.028290358523715
$h(7) = h(25)$	0.025419571045152	0.014726120434996	0.009865261180708
$h(8) = h(24)$	0.051297248225981	0.038071092404889	0.039857534186325
$h(9) = h(23)$	−0.020236717391193	−0.009608574065867	−0.004686550027946
$h(10) = h(22)$	−0.048229979616962	−0.047629842434778	−0.040880848492781
$h(11) = h(21)$	−0.015780111856345	−0.016700813278929	−0.037798341591623
$h(12) = h(20)$	0.076501409631873	0.062358850767844	0.059172893056331
$h(13) = h(19)$	0.057614929086958	0.076547973471596	0.110601322934505
$h(14) = h(18)$	−0.057332360581376	−0.070167621281884	−0.101967156482842
$h(15) = h(17)$	−0.311933920441146	−0.310940982209537	−0.300736729061916
$h(16)$	0.575357591017140	0.575267503881518	0.575357591017140

changed according to (7), a bird may not, due to inertia, fly towards a region at which it thinks is most promising for food. Instead, it may be leading toward a region which is in opposite direction of what it should fly in order to reach the expected promising regions. So, in the step that follows, the direction of the bird's velocity should be reversed in order for it to fly back to the promising region. $sign(r_3)$ is introduced for this purpose. In birds' flocking or fish schooling, a bird or a fish often changes directions suddenly. This is described by using a ''craziness'' factor and is modeled in the technique by using a craziness variable. A craziness operator is introduced in the proposed technique to ensure that the particle would have a predefined craziness probability to maintain the diversity of the particles. Consequently, before updating its position the velocity of the particle is crazed by,

Table 6 Optimized coefficients of FIR HP filter of order 40.

$h(n)$	RGA	PSO	CRPSO
$h(1) = h(41)$	−0.006506181211789	−0.007953432321525	−0.004081460107780
$h(2) = h(40)$	−0.012704844680354	0.002016648911279	−0.000142698920317
$h(3) = h(39)$	0.016487494961960	−0.000707251385234	0.002924233454458
$h(4) = h(38)$	0.013759426667389	0.012106972734307	0.008371924795646
$h(5) = h(37)$	−0.017963304796543	−0.002852214605626	−0.001119535488860
$h(6) = h(36)$	−0.011080946472428	−0.019937884874224	−0.017488381099820
$h(7) = h(35)$	0.001772074702866	0.001892701032620	−0.000472797199271
$h(8) = h(34)$	0.013738458111875	0.020122717096913	0.017110843717723
$h(9) = h(33)$	0.001660297871590	0.016363469812963	0.015533149833372
$h(10) = h(32)$	−0.008049662482544	−0.029477180344732	−0.022387937608300
$h(11) = h(31)$	−0.029710127279378	−0.014450566726646	−0.030463986771984
$h(12) = h(30)$	0.003041522418197	0.002770038339279	0.025844369874274
$h(13) = h(29)$	0.061516868140425	0.057265315321892	0.035927214985687
$h(14) = h(28)$	−0.011463073354542	−0.017256607690928	−0.001993514568381
$h(15) = h(27)$	−0.064157282493422	−0.061201762521351	−0.061733504594963
$h(16) = h(26)$	−0.014274121015648	−0.005155495982084	−0.021041659372839
$h(17) = h(25)$	0.054199911814203	0.058052052534273	0.085247146569075
$h(18) = h(24)$	0.092325869561206	0.068780863311564	0.047790530313293
$h(19) = h(23)$	−0.093381731985864	−0.061925216030327	−0.053670827801687
$h(20) = h(22)$	−0.302954617595263	−0.312978198273881	−0.308816183447219
$h(21)$	0.575818514350800	0.575818514350800	0.575818514350800

Table 7 Comparison summery of stop band attenuations for different orders and different algorithms.

Order	Maximum stop-band ripple (dB)			
	PM	RGA	PSO	CRPSO
20	−15.58	−19.35	−20.76	−22.83
30	−20.49	−22.19	−23.76	−28.78
40	−25.09	−25.74	−29.10	−31.89

Table 8 Comparison summery of the parameters of interest of order 20 for different algorithms.

Algorithm	Order 20				
	Maximum stop band attenuation (dB)	Maximum pass band ripple (normalized)	Maximum stop band ripple (normalized)	Transition width	Execution time per 100 cycles
PM	−15.58	0.166	0.1657	0.0574	–
RGA	−19.35	0.119	0.1078	0.0698	3.5783
PSO	−20.76	0.134	0.09163	0.0754	2.5347
CRPSO	−22.83	0.126	0.0722	0.0794	2.6287

Table 9 Comparison summery of the parameters of interest of order 30 for different algorithms.

Algorithm	Order 30				
	Maximum stop band attenuation (dB)	Maximum pass band ripple (normalized)	Maximum stop band ripple (normalized)	Transition width	Execution time per 100 cycles
PM	−20.49	0.095	0.0948	0.0463	–
RGA	−22.19	0.120	0.0777	0.0517	4.6733
PSO	−23.76	0.073	0.0649	0.0575	3.6125
CRPSO	−28.78	0.139	0.03638	0.0620	3.7328

Table 10 Comparison summery of the parameters of interest of order 40 for different algorithms.

Algorithm	Order 40				
	Maximum stop band attenuation (dB)	Maximum pass band ripple (normalized)	Maximum stop band ripple (normalized)	Transition width	Execution time per 100 cycles
PM	−25.09	0.056	0.05557	0.0407	–
RGA	−25.74	0.134	0.04678	0.046	5.8867
PSO	−29.10	0.130	0.03508	0.0505	4.7082
CRPSO	−31.89	0.153	0.02544	0.0543	4.8875

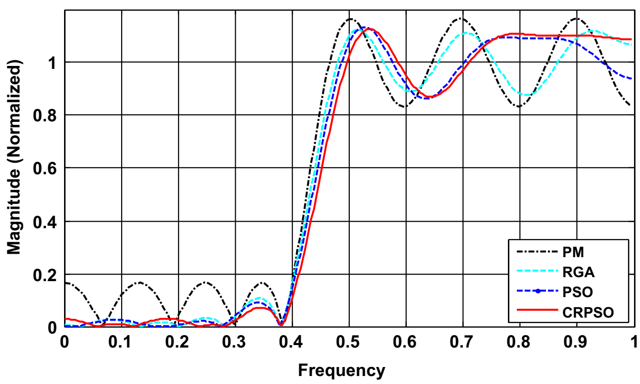

Figure 1 Normalized frequency response for the FIR HP filter of order 20.

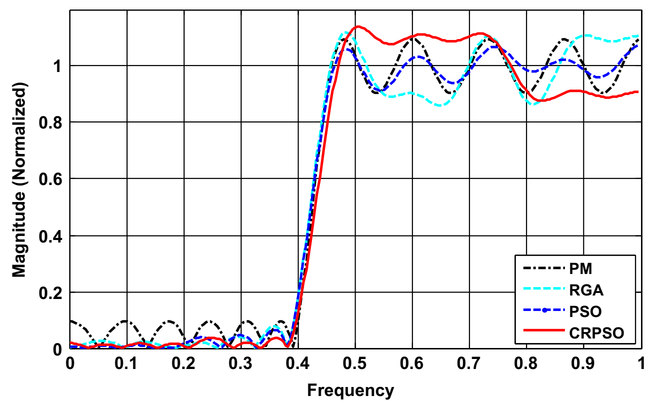

Figure 3 Normalized frequency response for the FIR HP filter of order 30.

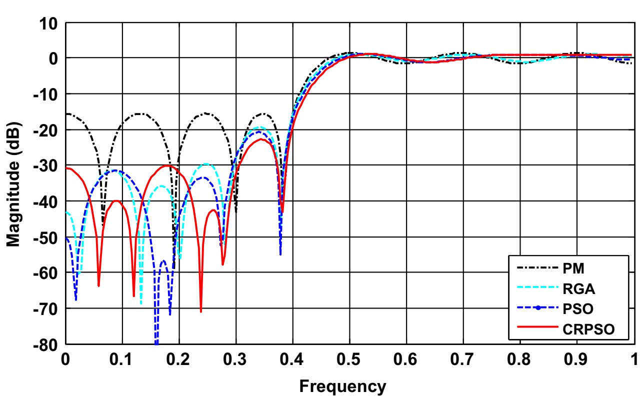

Figure 2 Gain (dB) plot of the FIR HP filter of order 20.

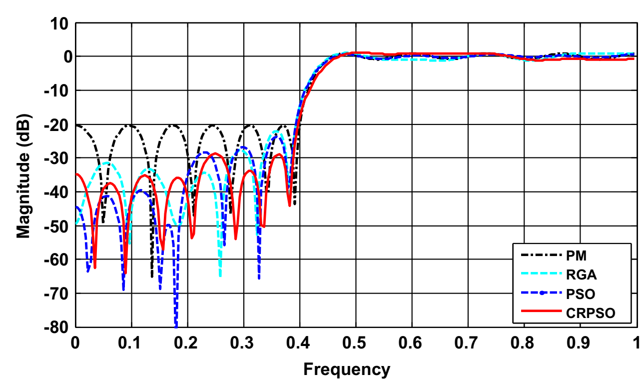

Figure 4 Gain (dB) plot of the FIR HP filter of order 30.

$$V_i^{(k+1)} = V_i^{(k+1)} + P(r_4) * sign(r_4) * v^{craziness} \quad (10)$$

where r_4 is a random parameter which is chosen uniformly within the interval [0, 1];

$v^{craziness}$ is a random parameter which is uniformly chosen from the interval $\left[v_i^{\min}, v_i^{\max}\right]$; and $P(r_4)$ and $sign(r_4)$ are defined, respectively, as:

$$\begin{aligned} P(r_4) &= 1 \quad \text{when } r_4 \leqslant P_{cr} \\ &= 0 \quad \text{when } r_4 > P_{cr} \end{aligned} \quad (11)$$

$$\begin{aligned} sign(r_4) &= -1 \quad \text{when } r_4 \geqslant 0.5 \\ &= 1 \quad \text{when } r_4 < 0.5 \end{aligned} \quad (12)$$

where P_{cr} is a predefined probability of craziness and *iter* means iteration cycle number.

The steps of CRPSO algorithm are given in Table 2. The values of the parameters used for the RGA, PSO and CRPSO techniques are given in Table 3.

4. Results and discussion

4.1. Analysis of magnitude response of FIR HP filters

In order to demonstrate the effectiveness of the proposed filter design method, several examples of FIR filter are constructed using PM, RGA, PSO and CRPSO algorithms. The MATLAB

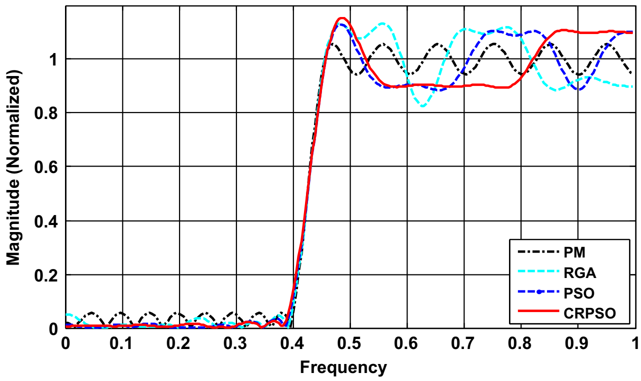

Figure 5 Normalized frequency response for the FIR HP filter of order 40.

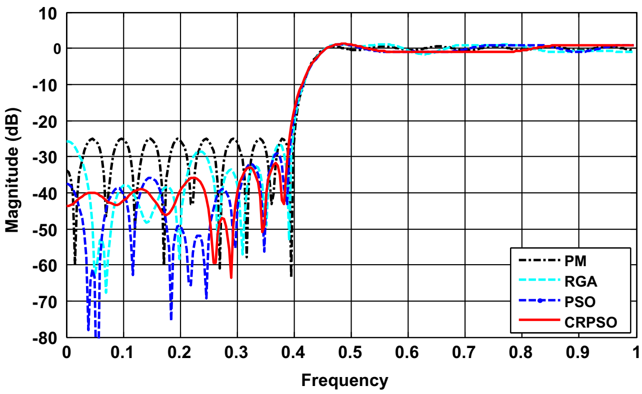

Figure 6 Gain (dB) plot of the FIR HP filter of order 40.

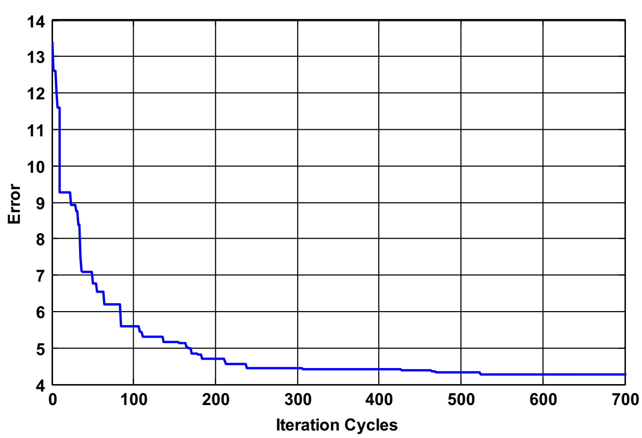

Figure 7 Convergence profile for RGA in case of 40th order FIR HP filter.

simulation has been performed extensively to realize the FIR HP filters of the order of 20, 30 and 40, respectively. Hence, the lengths of the filter coefficients are 21, 31, and 41, respectively. The sampling frequency has been chosen as $f_s = 1$ Hz. Also, for all the simulations the number of sampling points

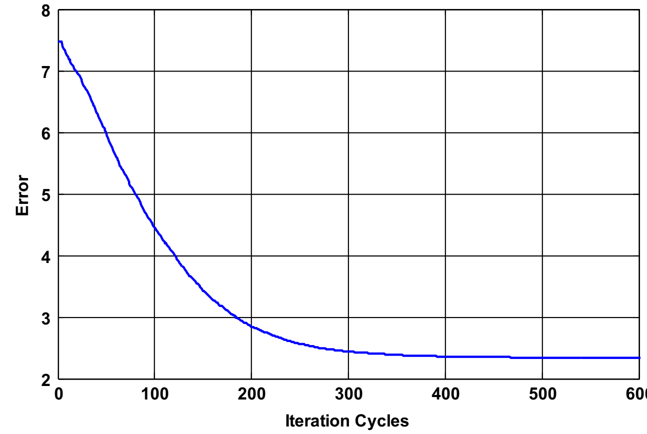

Figure 8 Convergence profile for PSO in case of 40th order FIR HP filter.

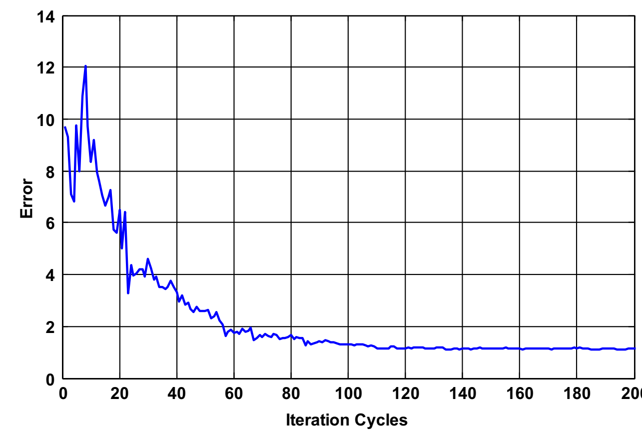

Figure 9 Convergence profile for CRPSO in case of 40th order FIR HP filter.

is taken as 128. Algorithms are run for 30 times to get the best solutions. The best results are reported in this work.

Table 3 shows the best chosen parameters used for different optimizations algorithms.

The parameters of the filter to be designed are as follows:

- Pass band ripple (δ_p) = 0.1.
- Stop band ripple (δ_s) = 0.01.
- Pass band (normalized) edge frequency (ω_p) = 0.45.
- Stop band (normalized) edge frequency (ω_s) = 0.40.
- Transition width = 0.05.

The best optimized coefficients for the designed FIR HP filters with the order of 20, 30 and 40 have been calculated by PM algorithm, RGA, PSO and CRPSO and are given in Tables 4–6, respectively. Tables 7–10 summarize the results of different performance parameters obtained using PM, RGA, PSO and CRPSO algorithms for HP filters of order 20, 30 and 40, respectively.

Figs. 1, 3 and 5 show the normalized frequency responses of the FIR HP filters of orders 20, 30 and 40, respectively. Figs. 2, 4 and 6 show the magnitude (dB) plots for the FIR HP filters of orders 20, 30 and 40, respectively.

The proposed CRPSO based approach for 20th order HP filter design results in 22.83 dB stop band attenuation, maximum pass band ripple (normalized) = 0.126, maximum stop band ripple (normalized) = 0.0722, transition width 0.0794. The proposed CRPSO based approach for 30th order HP filter design results in 28.78 dB stop band attenuation, maximum pass band ripple (normalized) = 0.139, maximum stop band ripple (normalized) = 0.03638, transition width 0.0620. The simulation results show that the proposed CRPSO based approach for 40th order HP filter design results in 31.89 dB stop band attenuation, maximum pass band ripple (normalized) = 0.153, maximum stop band ripple (normalized) = 0.02544, transition width is 0.0543. The novelty of the proposed filter design approach is also justified by the comparison made with (Sarangi et al., 2011). The particle swarm optimization with quantum infusion (PSO-QI) model proposed in Sarangi et al. (2011) reveals no improvement with respect to the PM algorithm, whereas, the proposed filter design technique shows 7.25 dB, 8.29 dB, 6.8 dB improvement as compared to PM for the HP filter of orders 20, 30 and 40, respectively.

From the diagrams and above discussions it is evident that with almost same level of the transition width, the proposed CRPSO based filter design approach produces the highest stop band attenuation (dB) and the lowest stop band ripple at the cost of very small increase in the pass band ripple compared to those of PM algorithm, RGA and conventional PSO. So, in the stop band region, the filters designed by the CRPSO results in the best responses. From Tables 7–10, one can finally infer that the CRPSO based filter design approach is the best among those of the literatures available for this purpose.

4.2. Comparative effectiveness and convergence profiles of RGA, PSO, and CRPSO

In order to compare the algorithms in terms of the error fitness value, Figs. 7–9 show the convergences of error fitnesses obtained when RGA, PSO and the CRPSO are employed, respectively. The convergence profiles are shown for the HP filter of order 40. Similar plots have also been obtained for the HP filters of orders of 20 and 30, which are not shown here. The CRPSO converges to much lower error fitness as compared to RGA, and PSO which yield suboptimal higher values of error fitnesses. As shown in Figs. 7–9, in case of HP filter of order 40, RGA converges to the minimum error fitness value of 4.27 in 41.2069s; PSO converges to the minimum error fitness value of 2.34 in 28.2492s; whereas, CRPSO converges to the minimum error fitness value of 1.1 in 9.775s. The above-mentioned execution times may be verified from Tables 8–10. Similar observations hold good for HP filters of orders 20 and 30 as shown in the same tables.

For all HP filters of different orders, the CRPSO algorithm converges to the least minimum error fitness values in finding the optimum filter coefficients in less number of iteration cycles. Fig. 7 shows that RGA converges to the minimum error fitness value of 4.27 in more than 500 iteration cycles; Fig. 8 shows that PSO converges to the minimum error fitness value of 2.34 in more than 450 iteration cycles; whereas, Fig. 9 shows that the proposed CRPSO algorithm converges to the minimum error fitness value of 1.1 in less than 200 iteration cycles.

With a view to the above fact, it may finally be inferred that the performance of CRPSO algorithm is the best among all algorithms. All optimization programs were run in MATLAB 7.5 version on core (TM) 2 duo processor, 3.00 GHz with 2 GB RAM.

5. Conclusions

In this paper, a novel Craziness based Particle Swarm Optimization (CRPSO) is applied to the solution of the constrained, multi-modal, non-differentiable, and highly nonlinear FIR HP filter design with optimal filter coefficients. With almost same level of the transition width, the CRPSO produces the highest stop band attenuation and the lowest stop band ripple at the cost of very small increase in the pass band ripple compared to those of PM algorithm, RGA and PSO. It is also evident from the results obtained by a large number of trials that the CRPSO is consistently free from the shortcoming of premature convergence exhibited by the other optimization algorithms. The simulation results clearly reveal that the CRPSO may be used as a good optimizer for the solution of obtaining the optimal filter coefficients in a practical digital filter design problem in digital signal processing systems.

References

Ababneh, J.I., Bataineh, M.H., 2008. Linear phase FIR filter design using particle swarm optimization and genetic algorithms. Digital Signal Process. 18, 657–668.

Biswal, B., Dash, P.K., Panigrahi, B.K., 2009. Power quality disturbance classification using fuzzy C-means algorithm and adaptive particle swarm optimization. IEEE Trans. Ind. Electron. 56 (1), 212–220.

Chen, S., 2000. IIR Model identification using batch-recursive adaptive simulated annealing algorithm. In: Proceedings of 6th Annual Chinese Automation and Computer Science Conference. pp. 151–155.

Eberhart, R., Shi, Y., 2000. Comparison between genetic algorithms and particle swarm optimization. In: Proc. 7th Ann. Conf. Evolutionary Computation, San Diego.

Filho, S., et al., 2000. Graphical User Interactive interface for digital filter design and DSP implementation. In: Proceedings of ICSPAT'2000, pp. 885–889

Herrmann, O., Schussler, W., 1970. Design of non-recursive digital filters with linear phase. Electron. Lett. 6, 329–330.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proc. IEEE Int. Conf. Neural Network.

Ling, S.H., Iu, H.H.C., Leung, F.H.F., Chan, K.Y., 2008. Improved hybrid particle swarm optimized wavelet neural network for modeling the development of fluid dispensing for electronic packaging. IEEE Trans. Ind. Electron. 55 (9), 3447–3460.

Litwin, L., 2000. FIR, IIR digital filters. IEEE Potentials 0278–6648, 28–31.

Luitel, B., Venayagamoorthy, G.K., 2008. Differential evolution particle swarm optimization for digital filter design. 2008 IEEE Congress Evolutionary Computation (CEC 2008), 3954–3961.

Mandal, D., Ghoshal, S.P., Bhattacharjee, A.K., 2010. Radiation pattern optimization for concentric circular antenna array with central element feeding using craziness based particle swarm optimization. In: International Journal of RF and Microwave Computer-Aided Engineering, vol. 20(5). John Wiley & Sons, Inc., pp. 577–586 (September).

Mandal, D., Ghoshal, S.P., Bhattacharjee, A.K., 2009. Comparative optimal designs of non-uniformly excited concentric circular antenna array using evolutionary optimization techniques. In: IEEE Second International Conference on Emerging Trends in Engineering and Technology, ICETET'09. pp. 619–624.

Mandal, D., Ghoshal, S.P., Bhattacharjee, A.K., 2010. Application of evolutionary optimization techniques for finding the optimal set of concentric circular antenna array. Expert Syst. Appl. (Elsevier) 38, 2942–2950.

Mastorakis, N.E., Gonos, I.F., Swamy, M.N.S., 2003. Design of two dimensional recursive filters using genetic algorithms. IEEE Trans. Circ. Syst. I – Fundam. Theory Appl. 50, 634–639.

McClellan, J.H., Parks, T.W., Rabiner, L.R., 1973. A computer program for designing optimum FIR linear phase digital filters. IEEE Trans. Audio Electro Acoust. AU-21, 506–526.

Parks, T.W., McClellan, J.H., 1972. Chebyshev approximation for non recursive digital filters with linear phase. IEEE Trans. Circ. Theory CT-19, 189–194.

Rabiner, L.R., 1973. Approximate design relationships for low-pass FIR digital filters. IEEE Trans. Audio Electro Acoust. AU-21, 456–460.

Sarangi, A., Mahapatra, RK., Panigrahi, SP., 2011. DEPSO and PSO-QI in digital filter design. Expert Syst. Appl. 38 (9), 10966–10973.

Yuksel, Ozbay., Bekir, Karlik., Kavsaoglu, Resit, A., 2003. A windows-based digital filter design. Math. Comput. Appl. 8 (3), 287–294.