International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:5, No:12, 2011

Linear Phase High Pass FIR Filter Design using Improved Particle Swarm Optimization

Sangeeta Mondal, Vasundhara, Rajib Kar, Durbadal Mandal, S. P. Ghoshal

***Abstract*—**This paper presents an optimal design of linear phase digital high pass finite impulse response (FIR) filter using Improved Particle Swarm Optimization (IPSO). In the design process, the filter length, pass band and stop band frequencies, feasible pass band and stop band ripple sizes are specified. FIR filter design is a multi-modal optimization problem. An iterative method is introduced to find the optimal solution of FIR filter design problem. Evolutionary algorithms like real code genetic algorithm (RGA), particle swarm optimization (PSO), improved particle swarm optimization (IPSO) have been used in this work for the design of linear phase high pass FIR filter. IPSO is an improved PSO that proposes a new definition for the velocity vector and swarm updating and hence the solution quality is improved. A comparison of simulation results reveals the optimization efficacy of the algorithm over the prevailing optimization techniques for the solution of the multimodal, non-differentiable, highly non-linear, and constrained FIR filter design problems.

***Keywords*—**FIR Filter, IPSO, GA, PSO, Parks and McClellan Algorithm, Evolutionary Optimization, High Pass Filter

I. Introduction

DIGITAL filter is an important part of digital signal processing (DSP). The popularity of the DSP can be owed to the extraordinary behavior of the filters. Filters basically serve two purposes of signal separation and signal restoration. Signal separation is needed when a signal has been contaminated with interference, noise or other signal. Signal restoration is needed when a signal has been distorted in some way. Both of these problems can be rectified by both analog and digital filters. Though analog filters are cheap, fast and have large dynamic ranges both in amplitude and frequency, digital filters are vastly superior in the level of performance that can be achieved. A digital filter takes a digital input, gives a digital output, and consists of digital components. On the other hand an analog filter operates directly on the analog input and is built entirely with analog components, such as resistors, capacitors, and inductors. In analog filters limitations are imposed by the electronic components such as accuracy and stability of the resistors and capacitors, whereas no such restrictions are there in the digital filter. Due to the definite nature of the coefficients of the digital filters, they are able to achieve much more complex and selective designs. With the digital filter one can achieve the target of a lower pass band ripple, faster transition, and higher stop band attenuation. Digital filter may be more expensive than an equivalent analog filter due to their increased complexity, but they make many practical designs that are impractical or impossible as analog filters. Thus digital filter outperforms analog filter in many aspects.

Digital filter usually comes in two categories FIR and IIR. A finite impulse response (FIR) filter is a type of digital filter whose impulse response is of finite duration, since the impulse response settles to zero within a finite amount of time. Whereas an IIR filter has impulse response function that is non-zero over an infinite length of time. The impulse response never dies out [1]. FIR filter has a number of useful properties, which gains a lot of preferences over the IIR filter. FIR filter requires no feedback, which makes its implementation simpler. The lack of feedback ensures that the impulse response will be finite [2]. Since there is no required feedback, all the poles are located at the origin and thus are located within the unit circle. FIR filters have only zeros (no poles), hence also known as all-zero filters. FIR filters also known as feed forward or non recursive, or transversal filters. FIR filters are inherently stable. FIR filters can be easily designed to have linear phase by making the coefficients symmetric. There is a great flexibility in shaping their magnitude response. They are easy and convenient to implement. FIR filters are dependent upon linear-phase characteristics, whereas IIR filters are used for applications which are not linear. FIR's delay characteristic is much better but they require more memory. IIR filters are dependent on both input and output whereas FIR filters are only dependent on the input. IIR filters can become difficult to implement, and also delay and distortion adjustments can alter the poles and zeros, which make the filters unstable, whereas FIR filters always remain stable. FIR filters are used for tapping of a higher-order, and IIR filters are better for tapping of lower-orders, since IIR filters may become unstable with tapping higher orders.

Traditionally, there are many well known methods of filter design such as the window method, frequency sampling method and the optimal filter design methods. The windowing method simply consists of truncating or windowing a theoretically ideal filter impulse response by some suitably chosen window function. The window method for digital filter design is fast, convenient, robust but generally suboptimal. A window is a finite array of coefficients selected to satisfy the desirable requirements.

There are various kinds of window functions (Butterworth, Chebyshev, Kaiser, and Hamming) available depending on the filter specifications to be met like ripples in pass band and stop band, stop band attenuation and transition width. Its major disadvantage is the lack of precise control of the critical frequencies such as pass band cutoff frequency and stop band cutoff frequency. These values, in general, depend on the type of the window and the filter length. Remez Exchange algorithm proposed by Parks and McClellan, is used for the

Vasundhara, Rajib Kar, Durbadal Mandal are with the Department of Electronics and Communication Engg., National Institute of Technology, Durgapur, India (phone: +91-9434788056, e-mail: rajibkarece@gmail.com).

Sangeeta Mondal, S. P. Ghoshal are with the Department of Electronics and Communication Engg., National Institute of Technology, Durgapur, India (phone: +91-9434788056, e-mail: rajibkarece@gmail.com).

design of exact linear phase weighted Chebyshev FIR filter [3]. Further a computer program has been developed for the design of FIR digital filter by Parks McClellan [4]. The basic problem which limits the use of this particular method is that the relative values of the amplitude error in the frequency bands are specified by means of the weighting functions and not by deviations themselves. The program has to be iterated many times in order to meet the filter specifications in terms of stop band deviation, cutoff frequency and filter length [5].

Filter designing is a multimodal optimization problem, thus making it a quite interesting and innovative research field [6]. Now a day, FIR filter is designed with the evolutionary techniques, which provides far better control of parameters and more nearly approximate ideal filter [7]. Different heuristics and stochastic optimization methods have been developed, which have proved themselves quite efficient for the design of FIR filter like GA [7-9], simulated annealing [10], Tabu Search [11] and artificial bee colony optimization [12] etc. FIR filter has also been designed using differential evolution [13].

GA proves itself to be far more efficient in terms of obtaining local optimum while maintaining its moderate computational complexity but they are not very successful in determining the global minima in terms of convergence speed and solution quality [14].

In this paper, the benefits of designing the FIR filter using a more stochastic technique known as Particle Swarm Optimization has been explored. The PSO proves itself to be far more efficient than the previously discussed techniques in many aspects. Particle Swarm Optimization is an evolutionary optimization technique developed by Eberhart *et al.* [15]. The merits of PSO lie in its simplicity to implement as well as its convergence can be controlled via few parameters. Several works have already been done in order to explore the flexibility of FIR filter design provided by PSO. The comparison of GA and PSO has been already made [16]. Several modifications of the already existing PSO technique have been made so that its efficiency can be increased. PSO is used with the differential evolution [17] to obtain a hybrid optimization algorithm. The inertial weights and acceleration coefficients are the parameters of PSO whereas scaling factor and the recombination probability are the parameters of DE. With the use of this method, the optimization algorithm becomes insensitive to the parameters of PSO as well as DE. In [18], the hybrid differential evolution approach (HDE), which is derived from both differential evolution and PSO is used. The inertial weight concept and the neighbor topology of PSO are used with the concept of the DE, which avoids the trapping of the solution in local minima as well as it speeds up the convergence process. In DEPSO, new offspring is created by the mutation of global best, which is taken as one of the parent and Gaussian distribution is used [19]. PSO based on velocity differential mutation is used for avoiding the local minima [20]. In this paper velocity is mutated rather than the particle's position. PSO uses the concept of mutation to increase the convergence speed and the global search ability.

Quantum-behaved Particle Swarm Optimization (QPSO) which was proposed by Sun, is a novel algorithm based on the PSO and quantum model [21]. In this concept each particle has a quantum behavior. In quantum mechanics, a particle, instead of having position and velocity, has a wave function. By using this concept, one cannot find the positions and velocities of the particles of search space exactly so the algorithm gets modified accordingly. Discrete Particle Swarm Optimization method along with the concept of Quantum evolution can be used for combinatorial optimization problem [22]. In this method slight modification in the velocity update is done so that it can update adaptively and can avoid local minima. A new Quantum based PSO which uses hyper-chaotic discrete system equation, as hc-QPSO is also used [23]. The main concept in confining all the particles in identical particle system is to remove the seasonal fluctuation, which helps in better updating of the particle's position. Averaging out the search length is used to avoid the local minima and a 2-dimensional hyper chaotic sequence theory is used. QPSO is used for the design of FIR filters [24]. This algorithm reduces the computational time, converges to the global optima and proves to be more efficient than the other evolutionary techniques like GA and PSO. The new concept of Quantum infused PSO is also utilized for the design of digital filters [25].The global best is selected by comparing the global best obtained from the conventional PSO and the offspring obtained from the QPSO. By merging the two techniques of PSO and QPSO, the best of both the methods can be extracted so as to obtain better results.

PSO is used for the design of FIR digital filters by using LMS and Minimax strategies for different populations and number of iterations [26]. PSO is not only being used for filter design but also for various other optimization purposes like in electrical systems, antenna etc. PSO with little amount of modification as constrained PSO (CPSO) is used for the designing of a nonlinear MIMO system identification, where two types of kernels one linear and another Gaussian is used. For parameter optimization, CPSO is used to obtain optimal free parameters [27]. PSO is also used for the design of two folded reflected array antenna for 77 GHz, by optimizing maximum power in the direction of main beam as well as obtaining proper antenna diagram from the reflector configuration [28]. Beam forming system which uses adaptive array antennas is very useful in mobile communication. K.A. Papadopoulos proposes PSO and GA approaches for the optimization of multiple constraints like beam direction, suppression of side lobes and null placement and control [29]. Basically this is a multi-objective problem but all the objectives have been converted to a single one with the help of weighting factors, whose proper selection poses an important task [30]. PSO is used to determine the control parameters of proportional-integral (PI) or proportional-integral derivative (PID) for speed control of a field oriented control (FOC) induction motor. PSO in this case proves to be advantageous in terms of improving the step response characteristics in speed control as well as speed tracking of a FOC induction motor [31].

In this paper, a new method for the design of FIR filters has been discussed known as Improved Particle Swarm Optimization (IPSO) [32]. The basic concept of the conventional PSO has been modified so as to overcome the drawbacks encountered in the conventional PSO such as premature convergence and stagnation problem. The

simulation results discussed latter in this paper makes the scenario completely clear, thus justifying the superiority of the IPSO over the conventional PSO. The ability of the PSO as a multimodal optimization problem and its flexibility can be seen as its use is not only restricted for DSP but in many other fields as well [33-37].

The rest of the paper is arranged as follows. In section II, the FIR high pass filter design problem is formulated. Section III briefly discusses the algorithms of RGA, conventional PSO and the IPSO algorithm. Section IV describes the simulation results obtained for high pass FIR digital filter using PM algorithm, RGA, PSO and the proposed IPSO approach. Finally, section V concludes the paper.

II. High Pass FIR Filter Design

The main advantage of the FIR filter structure is that it can achieve exactly linear-phase frequency responses. That is why almost all design methods described in the literature deal with filters with this property. Since the phase response of linear-phase filters is known, the design procedures are reduced to real-valued approximation problems, where the coefficients have to be optimized with respect to the magnitude response only.

A digital FIR filter is characterized by,

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

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 even order FIR filter design with h(n) as positive even symmetric. The number of coefficients h(n) is N+1. But, because the h(n) coefficients are symmetrical, the dimension of the problem is halved. Thus, (N/2+1) number of h(n) coefficients are actually optimized, which are finally concatenated to find the required (N+1) number of filter coefficients. An ideal filter has a magnitude of one on the pass band and a magnitude of zero on the stop band. Error fitness is the error between the frequency responses of the ideal filter and the designed approximate filter. In each iteration of any optimization algorithm, error fitness values of particle vectors are calculated and used for updating the particle vectors with new coefficients h(n). The final particle vector obtained after a certain number of iterations or after the error fitness is below a certain limit is considered to be the optimal result, yielding an optimal filter. 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 and δ_s), the stop band attenuation and the transition width. These parameters are mainly decided by the filter coefficients which are 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 [6]. Other algorithms were originally developed by Parks and McClellan (PM) [3] in which *N*, w_p, w_s, and the ratio δ_p/δ_s are fixed.

In this paper, evolutionary optimization algorithms RGA, conventional PSO and IPSO are individually applied to obtain the actual designed filter response as close as possible to the ideal response.

Now for (1), the particle i.e. the coefficient vector $\{h_0, h_1 \ldots h_n\}$, which is optimized, is represented in (N/2+1) dimension instead of (N+1) dimension.

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

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

where $w_k = \frac{2\pi k}{N}$; $H\left(e^{jw_k}\right)$ is the Fourier transform complex vector. This is the FIR filter's frequency response. The frequency is sampled in $[0, \pi]$ with N points. Different kinds of error fitness functions have been used in different literatures. An error function given by (3) is the approximate error used in Parks–McClellan algorithm for filter design [3].

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

where $H_d\left(e^{j\omega}\right)$ is the frequency response of the designed approximate filter; $H_i\left(e^{j\omega}\right)$ is the frequency response of the ideal filter; $G(\omega)$ is the weighting function used to provide different weights for the approximate errors in different frequency bands. For ideal HP filter, $H_i\left(e^{j\omega}\right)$ is given as,

$$H_i\left(e^{j\omega}\right)=0 \quad for \quad 0\le\omega\le\omega_c; \quad =1 \quad \text{otherwise} \quad (4)$$

where ω_c is the cut-off frequency. The major drawback of PM algorithm is that the ratio of δ_p/δ_s is fixed. To improve the flexibility in the error function to be minimized, so that the desired level of δ_p and δ_s may be specified, the error function given in (5) has been considered as fitness function in many literatures [13] [14] [37]. The error fitness to be minimized using the evolutionary algorithms, is defined as:

$$J_1=\max_{\omega\le\omega_p}\left(|E(\omega)|-\delta_p\right)+\max_{\omega\ge\omega_s}\left(|E(\omega)|-\delta_s\right) \quad (5)$$

where δ_p and δ_s are the ripples in the pass band and stop band; and ω_p and ω_s are pass band and stop band normalized cut-off frequencies, respectively. Since the coefficients of the linear phase positive symmetric even order filter are matched, the dimension of the problem is halved. This greatly reduces the computational complexity of the algorithms.

A. Real Coded Genetic Algorithm (RGA)

Standard genetic algorithm (also known as real coded GA) 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 fitness function or the objective function of the corresponding optimization problem. Steps of RGA as implemented for optimization of h(n) coefficients are [34-35]:

- Initialization of real chromosome strings of n_p population, each consisting of a set of (N/2+1) number of h(n) coefficients.
- Decoding of strings and evaluation of error fitness of each string.
- Selection of elite strings in order of increasing error fitness values from the minimum value.
- Copying of the elite strings over the non-selected strings.
- Crossover and mutation to generate off-springs.
- Genetic cycle updating and repeat from the step of evaluation error fitness value of each string.

The iteration stops when the maximum number of genetic cycles is reached. The grand minimum error fitness value, its corresponding chromosome string or the desired optimal solution vector h(n) having positive symmetric (N/2+1) coefficients are obtained. Finally, (N+1) number of coefficients is formed to obtain the optimal frequency spectrum.

B. Conventional 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. PSO is less susceptible to getting trapped on local optima unlike GA, Simulated Annealing etc. Eberhart *et al.* [15-16] developed PSO concept similar to the behavior of a swarm of birds. PSO is developed through simulation of bird flocking in multi-dimensional 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 vector h(n) 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 or sub-strings as required number of normalized 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 i[th] particle vector at k[th] 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 i[th] particle vector h(n) at k[th] iteration; $pbest_i^k$ is the personal best of the i[th] particle at the k[th] iteration; $gbest^k$ is the group best of the group at the k[th] 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 vector. Without the second and third terms, the particle vector 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.

C. Improved Particle Swarm Optimization (IPSO)

The global search ability of conventional PSO is very much enhanced with the help of the following modifications. This modified PSO is termed as IPSO [32].

i) 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 of 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 changed according to (6), a bird may not, due to inertia, fly toward a region at which it thinks is most promising for food. Instead, it may be leading toward a region which is in the 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 into promising region. $sign(r_3)$ is introduced for this purpose. Both cognitive and social parts are modified accordingly. Other modifications are described below.

ii) A new variation in the velocity expression (6) is made by splitting the cognitive component (second part of (6)) into two different components. The first component is called *good experience component*. That is, the particle has a memory about its previously visited best position. This component is exactly the same as the cognitive component of the conventional PSO. The second component is given the name *bad experience component*. The bad experience component helps the particle to remember its previously visited worst position. The inclusion of the worst experience component in the behavior of the particle gives additional exploration capacity to the swarm. By using the bad experience component, the bird (particle vector) can bypass its previous worst position and always try to occupy a better position.

Finally, with all modifications, the modified velocity of the i[th] particle vector at the (k+1)[th] iteration is expressed as (8).

$$V_i^{(k+1)} = r_2 * sign(r_3) * V_i^k + (1-r_2) * C_1 * r_1 * \left\{pbest_i^k - S_i^k\right\} + (1-r_2) * C_2 * (1-r_1) * \left\{gbest^k - S_i^k\right\} + (1-r_2) * c_1 * r_1 \left(S_i^k - pworst_i^k\right) \quad (8)$$

where $sign(r_3)$ is a function defined as:

$sign(r_3) = -1$ where $r_3 \le 0.05$
$= 1$ where $r_3 > 0.05$

V_i^k is the velocity of the i[th] particle vector at the k[th] iteration; r_1, r_2 and r_3 are the random numbers between 0 and 1; S_i^k is the current position of the i[th] particle at the k[th] iteration; $pbest_i^k$ and $pworst_i^k$ are the personal best and the personal worst of the i[th] particle, respectively ; $gbest^k$ is the group best among all pbests for the group. The searching point in the solution space is modified by the equation (7) as usual. The steps of IPSO are given in Table I.

TABLE I
STEPS OF IPSO

Step 1: Initialization: Population (swarm size) of particle vectors, n_P=120; maximum iteration cycles=200; number of filter coefficients (h(n)), filter order, N=20; fixing values of C_1, C_2 as 2.05; minimum and maximum values of filter coefficients, hmin=-2, hmax= 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 (N/2+1) randomly with limits; Computation of initial 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 (pbest), group best solution vector (gbest).
Step 4: Updating the velocities as per (8); 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 pbest vectors, gbest 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, gbest is the vector of optimal filter coefficients (N/2+1); Form complete (N+1) coefficients by copying (because the filter has linear phase) before getting the optimal frequency spectrum.

III. RESULTS AND DISCUSSIONS

A. Analysis of Magnitude Response of High Pass FIR Filters

The simulation results discussed in this section clearly justifies the superiority of this new method IPSO over the other traditional methods like RGA and conventional PSO. The simulations have been performed in the MATLAB environment. For the purpose of designing high pass FIR filter, the order of the filter has been taken as 20, which ensures that the length of the coefficient vector will be 21. The sampling frequency has been fixed to f_s=1Hz. For all the simulation works carried out, the number of sampling points taken is 128. In order to extract the best results out of all the iterations, all the algorithms are made to run for 40 times.
Table II shows the parameters chosen in order to run different evolutionary algorithms. The proper selection of these parameters plays an important role in the convergence profile of the respective algorithm. First column shows the parameters chosen to run the RGA. The crossover rate has been fixed to 1. Two point crossover has been done. Gaussian Mutation is used with the mutation rate of 0.01. The selection probability is kept as 1/3.

TABLE II
RGA, PSO, IPSO PARAMETERS

Parameters	RGA	Conventional PSO	IPSO
Population size	120	25	25
Iteration Cycle	800	350	200
Crossover rate	1	-	-
Crossover	Two Point Crossover	-	-
Mutation rate	0.01	-	-
Mutation	Gaussian Mutation	-	-
Selection	Roulette	-	-
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	-

Table III shows the optimized coefficients of FIR high pass filter of order 20. This table justifies the statement that the IPSO is more efficient in optimizing the filter coefficients.
Table IV shows the maximum stop band attenuations achieved by all three evolutionary algorithms for the design of FIR high pass filter of order 20.
In designing the FIR high pass filter using the evolutionary methods, the focus is kept on optimizing the stop band attenuation as far as possible. Table IV clearly shows that the target of achieving the maximum stop band attenuation is attained by IPSO easily. The comparisons have been made in order to clear the point between Park McClellan, RGA, conventional PSO and IPSO.
The maximum stop band attenuation achieved by IPSO is 29.59dB as compared to 28.52 dB of conventional PSO, 27.85dB of RGA and 23.24 dB of PM. Table V shows the maximum stop band attenuation (dB), maximum pass band ripple (normalized), maximum stop band ripple (normalized) and transition width for all the aforementioned optimization algorithms. This table makes the complete comparison between all the methods in respect of all the specifications required for the design of FIR high pass filter.

TABLE III
OPTIMIZED COEFFICIENTS OF FIR HIGH PASS FILTER OF ORDER 20

h(N)	RGA	Conventional PSO	IPSO
h(1)=h(21)	0.017500201326591	0.016684532700955	0.014051882288068
h(2)=h(20)	-0.034664703361583	-0.037026662035285	-0.034948107822049
h(3)=h(19)	0.039675441442365	0.036919859322288	0.041566245350805
h(4)=h(18)	-0.011039969232810	-0.013338708349390	-0.018209649407699
h(5)=h(17)	-0.029877373263831	-0.030173799057269	-0.024453045297124
h(6)=h(16)	0.060407393938230	0.060619921506844	0.056569079961636
h(7)=h(15)	-0.045357174244925	-0.041732050291725	-0.048853500210415
h(8)=h(14)	-0.036529016628391	-0.033569883346433	-0.032039242781614
h(9)=h(13)	0.151333684873435	0.153153102550027	0.149742172523051
h(10)=h(12)	-0.256137892428781	-0.255126724552754	-0.260703666822173
h(11)	0.300174715164156	0.300174715164156	0.305283906116923

TABLE IV
COMPARISON SUMMARY OF STOP BAND ATTENUATION OF FIR HIGH PASS FILTER OF ORDER 20

Filter Type	Maximum Stop-band Attenuation (dB)			
	PM	RGA	Conventional PSO	IPSO
High Pass	23.24	27.85	28.52	29.59

TABLE V
SUMMERY OF IPSO RESULTS WITH OTHER ALGORITHMS FOR FIR HIGH PASS FILTER OF ORDER 20

Algorithm	High Pass filter				
	Maximum stop band attenuation (dB)	Maximum pass band ripple (normalized)	Maximum stop band ripple (normalized)	Transition width	Execution time per 100 cycles
PM	23.24	0.069	0.06888	0.0845	-
RGA	27.85	0.137	0.04049	0.0896	3.6733
PSO	28.52	0.132	0.03752	0.0898	2.6125
IPSO	29.59	0.147	0.03316	0.0924	2.9872

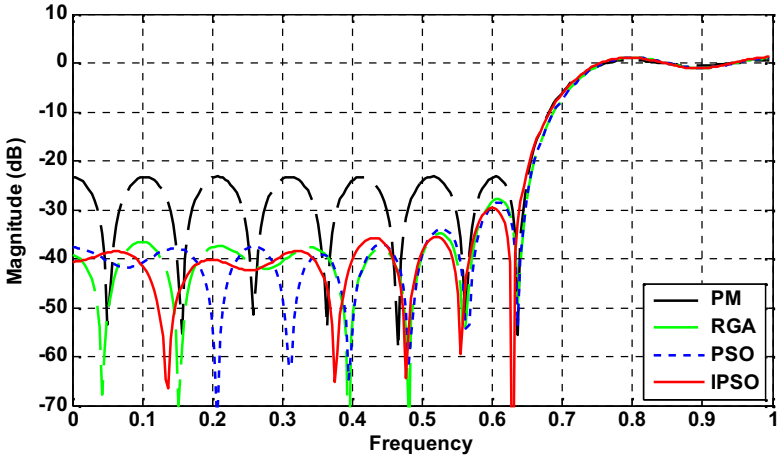

Fig. 1 Magnitude (dB) Plot of the FIR High Pass Filter of Order 20

From the table it can be easily brought to the notice that the required stop band ripple (normalized) is the minimum in the case of IPSO. It is 0.03316 in the case of IPSO as compared to 0.03752 in the case of PSO, 0.04049 in RGA and 0.06888 as in the case of PM. The table also shows that the target of achieving the required transition width is also achieved by IPSO. IPSO is also time economical, as one can see that the time required is also less as compared to RGA. Thus IPSO proves itself to be efficient in optimizing each and every specification required for the design of FIR high pass filter.

Fig. 1 shows the complete comparison of the magnitude responses obtained by PM, RGA, conventional PSO and IPSO. From this figure one can get completely convinced of the benefits of using IPSO. The curve drawn in red shows the magnitude response given by IPSO. This shows the much better magnitude response achieved by IPSO in terms of stop band attenuation, pass band ripple and the transition width than all other optimization techniques.

A. Comparative effectiveness and convergence profiles of RGA, conventional PSO and IPSO

The algorithms can be compared in terms of the convergence speeds also. Fig. 2 shows the plot of minimum error fitness values against the number of iteration cycles when RGA is employed. The total number of iteration cycle taken for this plot is 800. Fig. 3 shows the plot of minimum error values against the number of iteration cycles when PSO is employed. The total number of iteration cycles taken for this plot is 350. This figure shows the good convergence achieved by conventional PSO than RGA and the lesser value of error fitness is also attained.

Fig. 4 shows the plot of minimum error values against the number of iteration cycles when the proposed IPSO is employed. In this case the maximum number of iteration cycles taken is 200. Near global convergence is nicely achieved by this new optimization technique. IPSO achieves good convergence and lesser value of minimum error in far lesser number of iteration cycles. IPSO saves computational time as well as cost and hence it is a more efficient and smart way of optimization for the design of FIR high pass filter. Thus it can be very well said that the IPSO outperforms the traditional method of RGA and conventional PSO in each and every requirement needed for the design of FIR high pass filter. The convergence profiles have been shown for the filter order of 20.

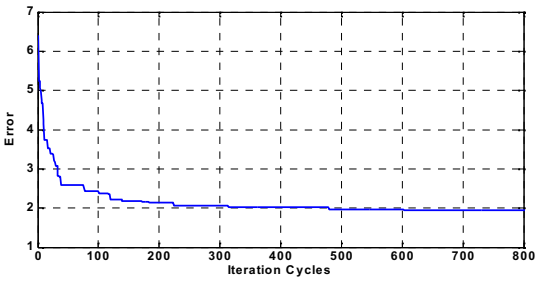

Fig. 2 Convergence Profile for RGA in case of FIR High Pass Filter of Order 20

From the figures drawn for this filter, it is seen that the IPSO algorithm is significantly faster than the conventional PSO algorithm for finding the optimum filter. The IPSO converges to a much lower fitness in lesser number of iterations. Further, PSO yields suboptimal higher values of error fitness but IPSO yields near optimal (least) error fitness values. With a view to the above fact, it may finally be inferred that the performance of IPSO technique is better as compared to RGA and conventional PSO in designing the optimal FIR High Pass filter. All optimization programs are run in MATLAB 7.5 version on core (TM) 2 duo processor, 3.00 GHz with 2 GB RAM.

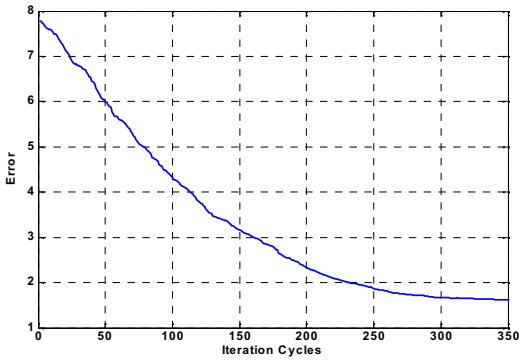

Fig. 3 Convergence Profile for conventional PSO in case of FIR High Pass Filter of Order 20

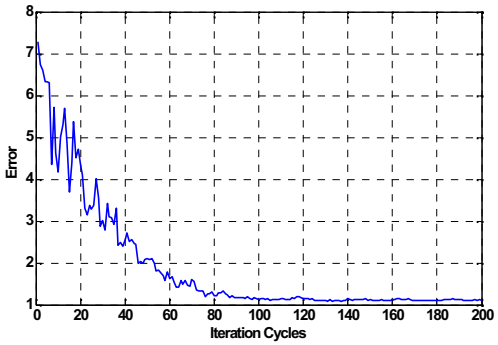

Fig. 4 Convergence Profile for IPSO in case of FIR High Pass Filter of Order 20

IV. Conclusion

This paper presents a novel and accurate method for designing linear phase digital FIR high pass filter by using evolutionary optimization based on IPSO. For the sake of comparison, Park McClellan and other evolutionary techniques as RGA, conventional PSO are applied individually. Extensive simulation results justify that the proposed algorithm IPSO outperforms Park McClellan, RGA and conventional PSO in the accuracy of the magnitude response of the filter as well as in the convergence speed and is thus adequate for use in other related design problems.

References

[1] L. Litwin, "FIR and IIR digital filters," *IEEE Potentials*, pp. 28–31, 2000.
[2] T.W. Parks, C.S. Burrus, *Digital Filter Design*, Wiley, New York, 1987.
[3] T.W. Parks, J.H. McClellan, "Chebyshev approximation for non recursive digital filters with linear phase,' *IEEE Trans. Circuits Theory*, CT-19, 1972, pp. 189–194.
[4] J.H. McClellan, T.W. Parks, L.R. Rabiner, "A computer program for designing optimum FIR linear phase digital filters," *IEEE Trans. Audio Electro acoust*., AU-21, pp. 506–526, 1973.
[5] L.R. Rabiner, "Approximate design relationships for High-pass FIR digital filters," *IEEE Trans. Audio Electro acoust*., AU-21, 1973, pp. 456–460.
[6] O. Herrmann, W. Schussler, Design of non-recursive digital filters with linear phase, *Electron. Lett.*, 6, 1970, pp.329–330.
[7] N.E. Mastorakis, I.F. Gonos, M.N.S Swamy, "Design of Two Dimensional Recursive Filters Using Genetic Algorithms," *IEEE*

Transaction on Circuits and Systems I - *Fundamental Theory and Applications*, 50, 2003, pp. 634–639.
[8] S.U. Ahmad, A. Antoniou, "A genetic algorithm approach for fractional delay FIR filters", *IEEE International Symposium on Circuits and Systems*, ISCAS 2006, pp.2517-2520.
[9] Hung-Ching Lu, Shian-Tang Tzeng, "Design of arbitrary FIR log filters by genetic algorithm approach," *Signal Processing*, 2000, 80, pp. 497-505.
[10] S. Chen, IIR Model Identification Using Batch-Recursive Adaptive Simulated Annealing Algorithm, *6th Annual Chinese Automation and Computer Science Conference*, 2000, pp.151–155.
[11] Karaboga D, Horrocks DH, Karaboga N, Kalinli A.: 'Designing digital FIR filters using Tabu search algorithm,' *IEEE International Symposium on Circuits and Systems*, 1997, ISCAS '97, vol.4, pp.2236-2239
[12] Karaboga N.: 'A new design method based on artificial bee colony algorithm for digital IIR filters', *Journal of the Franklin Institute*, 2009, 346, (4), pp.328–348
[13] B. Luitel, G.K. Venayagamoorthy, Differential Evolution Particle Swarm Optimization for Digital Filter Design, *IEEE Congress on Evolutionary Computation* (CEC 2008), PP. 3954-3961, 2008.
[14] J.I. Ababneh, M. H. Bataineh, Linear phase FIR filter design using particle swarm optimization and genetic algorithms, *Digital Signal Processing*, 18, 657–668, 2008.
[15] J. Kennedy, R. Eberhart, Particle Swarm Optimization, in *Proc. IEEE int. Conf. On Neural Network*, 1995.
[16] R. Eberhart, Y. Shi, Comparison between Genetic Algorithms and Particle Swarm Optimization, *Proc. 7th Ann. Conf. on Evolutionary Computation*, San Diego, 2000.
[17] Mahamed G.H. Omran, Andries P. Engelbrecht, Ayed Salman, " Differential Evolution Based Particle Swarm Optimization", *IEEE Swarm Intelligence Symposium*, SIS 2007, pp. 112-119, 2007.
[18] Shu Jun, Li Jian, "A Hybrid of Differential Evolution and Particle Swarm Optimization for Global Optimization", *Third International Symposium on Intelligent Information Technology Application*, 2009. IITA 2009, pp. 138-141.
[19] Luitel B, Venayagamoorthy GK.: 'Differential evolution particle swarm optimization for digital filter design,' *IEEE World Congress on Computational Intelligence* (*IEEE Congress on Evolutionary Computation*), CEC 2008, pp.3954-3961.
[20] Shanhe Jiang , Qishen Wang , Julang Jiang, "Particle Swarm Optimization Algorithm Based on Velocity Differential Mutation", *Control and Decision Conference*, 2009. pp. 1860-1865.
[21] J. Sun., B.Feng and W.B.Xu, "Particle Swarm Optimization with Particles Having Quantum Behavior", in *Proc. Congress on Evolutionary Computation*, 2004, pp. 325-331.
[22] Li Xuyuan ,Hualong Xu ,Zhaogang Cheng, "One Improved Discrete Particle Swarm Optimization based on Quantum Evolution Concept", *International Conference on Intelligent Computation Technology and Automation* (ICICTA 2008), vol 1, pp. 96-100.
[23] Jin Yanxia, Zhan Hanchang, "An improved quantum particle swarm optimization algorithm", *2nd International Conference on Information Science and Engineering*, 2010 , pp. 985-988.
[24] Wei Fang, Jun Sun, Wenbo Xu, and Jing Liu, "FIR Digital Filters Design Based on Quantum-behaved Particle Swarm Optimization", *First International Conference on Innovative Computing, Information and Control*, 2006,vol 1, pp. 615-619.
[25] Bipul Luitel, Ganesh Kumar Venayagamoorthy, "Particle Swarm Optimization with Quantum Infusion for the Design of Digital Filters", *Swarm Intelligence Symposium*, SIS-2008, 2008. 10.1109/SIS.2008, pp. 1-8.
[26] M. Najjarzadeh, A.Ayatollahi, "FIR Digital Filters Design: Particle Swarm Optimization Utilizing LMS and Minimax Strategies," *Signal Processing and Information Technology*, ISSPIT 2008, pp. 129-132.
[27] Mounira Tarhouni, Kaouther Laabidi, Salah Zidi, Moufida Ksouri-Lahmari, "A nonlinear MIMO system identification based on improved multi-kernel least squares support vector machines (improved multi-kernel LS-SVM)," *8th international multi-conference on systems, signals and devices*, 2011, pp 1-6.
[28] Sabine Dieter, Christoph Fischer, Wolfgang Menzel, "Design of a Folded Reflectarray Antenna Using Particle Swarm Optimization", *2010 European Microwave Conference* (EuMC), 2010, pp.731 – 734.
[29] K.A. Papadopoulos, C.A. Papagianni, C.A. Papas, D.I. Kaklamani, I.S. Venieris, "Beam Array Optimization for Smart Antenna Systems using Stochastic Algorithms", *3rd European Conference on Antennas and Propagation*, 2009, pp. 1345 – 1349.
[30] Adel A. A., El-Gammal, Adel A. El-Samahy, "A Modified Design of PID Controller For DC Motor Drives Using Particle Swarm Optimization PSO", *International Conference on Power Engineering, Energy and Electrical Drives*, 2009. POWERENG '09, pp. 419-424.
[31] Tista Banerjee, Sumana Choudhuri, Jitendranath Bera, Abhisek Maity, "Off-line Optimization of PI and PID Controller for a Vector Controlled Induction Motor Drive using PSO", *International Conference on Electrical and Computer Engineering* (ICECE), 2010, pp. 74 – 77.
[32] S. H. Ling, H. H. C. Iu, F. H. F. Leung, and K. Y. Chan, "Improved hybrid particle swarm optimized wavelet neural network for modeling the development of fluid dispensing for electronic packaging," *IEEE Trans. Ind. Electron*., vol. 55, no. 9, pp. 3447–3460, Sep. 2008.
[33] B. Biswal, P. K. Dash, and B. K. Panigrahi, "Power quality disturbance classification using fuzzy C-means algorithm and adaptive particle swarm optimization," *IEEE Trans. Ind. Electron.*, vol. 56, no. 1, pp. 212–220, Jan. 2009.
[34] D. Mandal, S. P. Ghoshal, and A. K. Bhattacharjee, "Application of Evolutionary Optimization Techniques for Finding the Optimal set of Concentric Circular Antenna Array," *Expert Systems with Applications*, vol. 38, pp. 2942-2950, 2010.
[35] D. Mandal, S. P. Ghoshal, and A. K. Bhattacharjee, "Comparative Optimal Designs of Non-uniformly Excited Concentric Circular Antenna Array Using Evolutionary Optimization Techniques," *IEEE Second International Conference on Emerging Trends in Engineering and Technology*, ICETET'09, 2009, pp. 619-624.
[36] Archana Sarangi, Rabi Kumar Mahapatra, Siba Prasada Panigrahi, DEPSO and PSO-QI in digital filter design, *Expert Systems with Application*s, vol. 38, 2011, 10966–10973.
[37] D.Mandal, S.P.Ghoshal and A.K.Bhattacharjee, "Swarm Intelligence based Optimal Design of Concentric Circular Antenna Array," *Journal of Electrical Engineering*, vol.10, no.3,pp 30-39,2010