An Efficient Craziness Based Particle Swarm Optimization Technique for Optimal IIR Filter Design

S.K. Saha[1], R. Kar[1], D. Mandal[1], and S.P. Ghoshal[2]

[1] Department of Electronics and Communication Engineering
[2] Department of Electrical Engineering,
National Institute of Technology, Durgapur, West Bengal, India
{namus.ahas,rajibkarece,durbadal.bittu,
spghoshalnitdgp}@gmail.com

Abstract. In this paper an improved version of Particle Swarm Optimization (PSO) called Craziness based PSO (CRPSO) is considered as an efficient optimization tool for designing digital Infinite Impulse Response (IIR) filters. Apart from gaining better control on cognitive and social components of conventional PSO, the CRPSO dictates better performance due to incorporation of craziness parameter in the velocity equation of PSO. This modification in the velocity equation not only ensures the faster searching in the multidimensional search space but also the solution produced is very close to the global optimal solution. The effectiveness of this algorithm is justified with a comparative study of some well established algorithms, namely, Real coded Genetic Algorithm (RGA) and conventional Particle Swarm Optimization (PSO) with a superior CRPSO based outcome for the designed 8th order IIR low pass (LP), high pass (HP), band pass (BP) and band stop (BS) filters. Simulation results affirm that the proposed CRPSO algorithm outperforms its counterparts not only in terms of quality output, i.e., sharpness at cut-off, pass band ripple and stop band attenuation but also in convergence speed with assured stability.

1 Introduction

Signal carries information, but this information is getting contaminated with noise which is picked up mostly by electro magnetic means. So, at the receiving end to extract the information signal processing is executed on noise corrupted signal. Depending on nature of signal and point of application signal processing may be analog, digital or mixed in practice. Application of digital signal processing (DSP) has increased many folds as the production of DSP in bulk is easier as the basic operation is confined into mainly addition, multiplication and recalling of previous data. In digital filter design minimum number of discrete components are required that immunes the performance of designed filter from thermal drift.

Digital filters are broadly classified into two main categories namely; finite impulse response (FIR) filter and infinite impulse response (IIR) filter [1-2]. The output of FIR filter depends on present and past values of input, so the name non-recursive is aptly suited to this filter. On the other hand, the output of IIR filter

M.L. Gavrilova et al. (Eds.): Trans. on Comput. Sci. XXI, LNCS 8160, pp. 230–252, 2013.

depends not only on previous inputs, but also on previous outputs with impulse responses continuing forever in time at least theoretically, so the name 'recursive' is aptly suited to this filter; anyway, a large memory is required to store the previous outputs for the recursive IIR filter.

Hence, due to these aspects FIR filter realization is easier with the requirement of less memory space and design complexity. Ensured stability and linear phase response over a wide frequency range are the additional advantages. On the other hand, IIR filter distinctly meets the supplied specifications of sharp transition width, lower pass band ripple and higher stop band attenuation with ensured lower order compared to FIR filter. As a consequence, a properly designed IIR filter can meet the magnitude response close to ideal and more finely as compared to a FIR filter. Due to these challenging features with wide field of applications, performances of IIR filters designed with various optimization algorithms are compared to find out the effectiveness of algorithms and the best optimal IIR filter with assured stability.

In the conventional design approach, IIR filters of various types (Butterworth, Chebyshev and Elliptic etc.) can be implemented with two methods. In the first case, frequency sampling technique is adopted for Least Square Error [3] and Remez Exchange [4] process. In the second method, filter coefficients and minimum order are calculated for a prototype low pass filter in analog domain which is then transformed to digital domain with bilinear transformation. This frequency mapping works well at low frequency, but in high frequency domain this method is liable to frequency warping [5].

IIR filter design is a highly challenging optimization problem. So far, gradient based classical algorithms such as steepest descent and quasi Newton algorithms have been aptly used for the design of IIR filters [6-7]. In general, these algorithms are very fast and efficient to obtain the optimum solution of the objective function for a unimodal problem. But the error surface (typically the mean square error between the desired response and estimated filter output) of IIR filter is multimodal and hence superior evolutionary optimization techniques are required to find out better near global solution.

The shortfalls of classical optimization techniques for handling the multimodal optimization problem are as follows:

- Requirement of continuous and differentiable error fitness function (cost or objective function),
- Usually converges to the local optimum solution or revisits the same sub-optimal solution,
- Incapable to search the large problem space,
- Requirement of the piecewise linear cost approximation (linear programming),

Highly sensitive to starting points when the number of solution variables is increased and as a result the solution space is also increased.

So, it can be concluded that classical search techniques are only suitable for handling differentiable unimodal objective function with constricted search space. But the error surface of IIR filter is usually multimodal and non-differentiable. So the various evolutionary heuristic search algorithms are applied for filter optimization problems, which are as follows: Genetic Algorithm (GA) is developed with the inspiration of the Darwin's "Survival of the Fittest" strategy [8-9]; Simulated Annealing (SA) is designed from the thermodynamic effects [10]; Artificial Immune Systems (AIS) mimics the biological immune systems [11]; Ant Colony Optimization (ACO) simulates the ants' food searching behaviour [12]; Bee Colony Optimization mimics the honey collecting behaviour of the bee swarm [13]; Cats Swarm Optimization(CSO) is based upon the behaviour of cats for tracing and seeking of an object [14]; and PSO and its variants simulate the behaviour of bird flocking or fish schooling [15-21].

Ecology based Predator-prey model as an evolutionary optimization technique is discussed in [22], where each prey is considered as a possible solution in search space which is chased by a predator in predefined region; Searching behaviour of human being is mimicked for the development of Seeker Optimization Algorithm (SOA) [23]; In Bacteria Foraging Optimization (BFO) technique food searching behaviour of E. Coli bacteria is mimicked [24].

Naturally, it is a vast area of research continuously being explored. In this paper, the capability of global searching and near optimum result finding features of GA, PSO and CRPSO are investigated thoroughly for solving 8th order IIR filter design problems. GA is a probabilistic heuristic search optimization technique developed by Holland [25]. The features such as multi-objective, coded variable and natural selection made this technique distinct and suitable for finding the near global solution of filter coefficients.

Particle Swarm Optimization (PSO) is swarm intelligence based algorithm developed by Eberhart *et al.* [26-27]. Several attempts have been taken to design digital filter with basic PSO and its modified versions [15-21], [28-29]. The main attraction of PSO is its simplicity in computation and a few steps are required in the algorithm.

The limitations of the conventional PSO are premature convergence and stagnation problem [30-31]. To overcome these problems an improved version of PSO called CRPSO is suggested by the authors for the design of 8th order digital IIR low pass (LP), high pass (HP), band pass (BP) and band stop (BS) filters.

The paper is organized as follows: Basic structure of IIR filter along with the error fitness function is described in section 2. Different evolutionary algorithms namely, RGA, PSO and CRPSO are discussed in section 3. In section 4, comprehensive and demonstrative sets of data and illustrations are analyzed to make a floor of comparative study of performances among different algorithms. Finally, section 5 concludes the paper.

2 IIR Filter Design Formulation

This section discusses the design strategy of IIR filter based on all concerned algorithms. The input-output relation is governed by the following difference equation [2].

$$y(p)+\sum_{k=1}^{n}a_k y(p-k)=\sum_{k=0}^{m}b_k x(p-k) \tag{1}$$

where $x(p)$ and $y(p)$ are the filter's input and output, respectively, and $n(\geq m)$ is the filter's order. With the assumption of coefficient, $a_0 = 1$ the transfer function of the IIR filter is expressed as:

$$H(z)=\frac{\sum_{k=0}^{m}b_k z^{-k}}{1+\sum_{k=1}^{n}a_k z^{-k}} \tag{2}$$

Let $z = e^{j\Omega}$. Then, the frequency response of the IIR filter becomes

$$H(\Omega)=\frac{\sum_{k=0}^{m}b_k e^{-jk\Omega}}{1+\sum_{k=1}^{n}a_k e^{-jk\Omega}} \tag{3}$$

$$H(\Omega)=\frac{Y(\Omega)}{X(\Omega)}=\frac{b_0+b_1e^{-j\Omega}+b_2e^{-j2\Omega}+\ldots+b_me^{-jm\Omega}}{1+a_1e^{-j\Omega}+a_2e^{-j2\Omega}+\ldots+b_ne^{-jn\Omega}} \tag{4}$$

where $\Omega = 2\pi\left(\frac{f}{f_s}\right)$ in [0, π] is the digital frequency; f is the analog frequency and f_s is the sampling frequency. Different fitness functions are used for IIR filter optimization problems [32-34]. The commonly used approach to IIR filter design is to represent the problem as an optimization problem with the mean square error (MSE) as the error fitness function [34] expressed in (5).

$$J(\omega)=\frac{1}{N_s}[(d(p)-y(p))^2] \tag{5}$$

where N_s is the number of samples used for the computation of the error fitness function; $d(p)$ and $y(p)$ are the filter's desired and actual responses, respectively. The difference $e(p)=d(p)-y(p)$ is the error between the desired and the actual filter responses. The design goal is to minimize the MSE $J(\omega)$ with proper adjustment of coefficient vector ω represented as:

$$\omega = [a_0 a_1 ... a_n b_0 b_1 ... b_m]^T . \tag{6}$$

In this paper, a novel error fitness function given in (7) is adopted in order to achieve higher stop band attenuation and to have moderate control on the transition width. Using (7), it is found that the proposed filter design approach results in considerable improvement in stop band attenuation over other optimization techniques.

$$J_1(\omega) = \sum_{\omega} abs\left[abs\left(\left| H_d(\omega) \right| - 1 \right) - \delta_p \right] + \sum_{\omega} \left[abs\left(\left| H_d(w) \right| - \delta_s \right) \right] \tag{7}$$

For the first term of (7), $\omega \in$ pass band including a portion of the transition band and for the second term of (7), $\omega \in$ stop band including the rest portion of the transition band. The portions of the transition band chosen depend on pass band edge and stop band edge frequencies.

The error fitness function given in (7) represents the generalized fitness function to be minimized using the evolutionary algorithms RGA, conventional PSO and the proposed CRPSO individually. Each algorithm tries to minimize this error fitness J_1 and thus optimizes the filter performance. Unlike other error fitness functions as given in [32-34] which consider only the maximum errors, J_1 involves summation of all absolute errors for the whole frequency band, and hence, minimization of J_1 yields much higher stop band attenuation and lesser pass band ripples.

3 Evolutionary Algorithms Employed

3.1 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 built upon the Darwin's "Survival of the Fittest" strategy [25]. Each encoded chromosome that constitutes the population is a solution to the filter designing optimization problem. These solutions may be good or bad, but are tested rigorously through the genetic operations such as crossover and mutation to evolve a global optimal or near global optimal solution of the problem at hand. Chromosomes are constructed over some particular 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 objective function of the corresponding optimization problem. Each chromosome has a probability of selection and has to take part in the genetic operation based upon the Roulette's wheel strategy. In the genetic operations, crossover and mutation bring the variation in alleles of gene in the chromosome population along with the alleviation of trapping to local optimal solution.

Steps of RGA as implemented for the optimization of coefficient vector ω are as follows [35-36]:

Step 1: Initialize the real coded chromosome strings (ω) of n_p = 120 population, each consisting of equal number of numerator and denominator filter coefficients b_k and a_k , respectively; total coefficients = (n+1)*2 for nth order filter to be designed; minimum and maximum values of filter coefficients, hmin = -2, hmax = 2; number of samples=128; $\delta_p = 0.001$, $\delta_s = 0.0001$; maximum iteration cycles= 400, n=8.

Step 2: Decoding of the strings and evaluation of error fitness $J_1(\omega)$ according to (7).

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 offspring.

Step 6: Genetic cycle updating.

Step 7: The iteration stops when maximum number of cycles is reached. The grand minimum error and its corresponding chromosome string or the desired solution having (n+1)*2 number of coefficients are finally obtained.

3.2 Particle Swarm Optimization (PSO)

PSO is flexible, robust, population based stochastic search algorithm with attractive features of simplicity in implementation and ability to quickly converge to a reasonably good solution. Additionally, it has the capability to handle larger search space and non-differential objective function, unlike traditional optimization methods. Eberhart *et al.* [26-27] developed PSO algorithm to simulate random movements of bird flocking or fish schooling.

The algorithm starts with the random initialization of a swarm of individuals, which are known as particles within the multidimensional problem search space, in which each particle tries to move toward the optimum solution, where next movement is influenced by the previously acquired knowledge of particle best and global best positions once achieved by individual and the entire swarm, respectively. The features incorporated within this simulation are velocity matching of individuals with the nearest neighbour, elimination of ancillary variables and inclusion of multidimensional search and acceleration by distance. Instead of the presence of direct recombination operators, acceleration and position modification supplement the recombination process in PSO. Due to the aforementioned advantages and simplicity, PSO has been applied to different fields of practical optimization problems.

To some extent, IIR filter design with PSO is already reported in [15-21], [28-29]. A brief idea about the algorithm for a D-dimensional search space with n_p particles that constitutes the flock is presented here. Each i^{th} particle is described by a position vector as $S_i = (s_{i1}, s_{i2}, ..., s_{iD})^T$ and velocity is expressed by $V_i = (v_{i1}, v_{i2}, ..., v_{iD})^T$.

The best position that the i^{th} particle has reached previously $pbest_i = (p_{i1}, p_{i2}, ..., p_{iD})^T$, and group best is expressed as $gbest = (p_{g1}, p_{g2}, ..., p_{gD})^T$.

The maximum and minimum velocities are $V_{\max}$, $V_{\min}$, respectively.

$$V_{\max} = (v_{\max 1}, v_{\max 2}, ..., v_{\max D})^T \text{ and } V_{\min} = (v_{\min 1}, v_{\min 2}, ..., v_{\min D})^T .$$

The positive constants C_1, C_2 are related with accelerations and $rand_1, rand_2$ lie in the range [0, 1]. The inertia weight w is a constant chosen carefully to obtain fast convergence to optimum result. k denotes the iteration number.

The basic steps of the PSO algorithm are as follows [19-21]:

Step1: Initialize the real coded particles (ω) of n_p = 25 population, each consisting of equal number of numerator and denominator filter coefficients b_k and a_k, respectively; total coefficients D = (n+1)*2 for equal number numerator and denominator coefficients with nth order filter to be designed; minimum and maximum values of filter coefficients, hmin = -2, hmax = 2; number of samples=128; $\delta_p = 0.001$, $\delta_s = 0.0001$; maximum iteration cycles= 100 ; n= 8.

Step 2: Compute the error fitness value for the current position S_i of each particle

Step 3: Each particle can remember its best position $(pbest)$ which is known as cognitive information and that would be updated with each iteration.

Step 4: Each particle can also remember the best position the swarm has ever attained $(gbest)$ and is called social information and would be updated in each iteration.

Step 5: Velocity and position of each particle are modified according to (8) and (9), respectively [26].

$$V_i^{(k+1)} = w * V_i^{(k)} + C_1 * rand_1 * \{pbest_i^{(k)} - S_i^{(k)}\} + C_2 * rand_2 * \{gbest_i^{(k)} - S_i^{(k)}\} \quad (8)$$

where
$$\begin{aligned} V_i &= V_{\max} \; for V_i > V_{\max} \\ &= V_{\min} \; for \; V_i < V_{\min} \end{aligned}$$

$$S_i^{(k+1)} = S_i^{(k)} + V_i^{(k+1)} \quad (9)$$

Step 6: The iteration stops when maximum number of cycles is reached. The grand minimum error fitness and its corresponding particle or the desired solution having (n+1)*2 number of coefficients are finally obtained.

3.3 Craziness Based Particle Swarm Optimization (CRPSO) Technique

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 [37]:

$$V_i^{(k+1)} = r_2 * sign(r_3) * V_i^k$$
$$+(1-r_2)*C_1*r_1*\left\{pbest_i^{(k)} - S_i^{(k)}\right\} + (1-r_2)*C_2*(1-r_1)*\left\{gbest^{(k)} - S_i^{(k)}\right\} \quad (10)$$

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:

$$\begin{aligned} sign(r_3) &= -1 \quad \text{where} \quad r_3 \le 0.05 \\ &= 1 \quad \text{where} \quad r_3 > 0.05 \end{aligned} \quad (11)$$

The two random parameters $rand_1$ and $rand_2$ of (8) 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 changed according to (9), 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 modelled 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,

$$V_i^{(k+1)} = V_i^{(k+1)} + P(r_4) * sign(r_4) * v_i^{craziness} \quad (12)$$

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 $[v_i^{\min}, v_i^{\max}]$; and $p(r_4)$ and $sign(r_4)$ are defined, respectively, as:

$$P(r_4) = 1 \quad \text{when} \quad r_4 \le P_{cr} \\ = 0 \quad \text{when} \quad r_4 > P_{cr} \tag{13}$$

$$sign(r_4) = -1 \quad \text{when} \quad r_4 \ge 0.5 \\ = 1 \quad \text{when} \quad r_4 < 0.5 \tag{14}$$

where P_{cr} is a predefined probability of craziness.

The steps of CRPSO algorithm are as follows:

Step 1: Population is initialized for a swarm of n_p vectors, in which each vector represents a solution of filter coefficient values.

Step 2: Computation of initial cost values of the total population, n_P.

Step 3: Computation of population based minimum cost value, i.e., the group best solution vector (*gbest*) and computation of the personal best solution vectors (*pbest*).

Step 4: Updating the velocities as per (10) and (12); updating the particle vectors as per (9) and checking against the limits of the filter coefficients; finally, computation of the updated cost values of the particle vectors and population based minimum cost 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 cost values are reached; finally, *gbest* is the vector of optimal IIR filter coefficients.

The justifications of choosing the value of different CRPSO parameters are as follows:

Reversal of the direction of bird's velocity should rarely occur; to achieve this, $r_3 \le 0.05$ (a very low value) is chosen when $sign(r_3)$ will be -1 to reverse the direction. If P_{cr} is chosen less or, equal to 0.3, the random number r_4 will have more probability to become more than P_{cr}, in that case, craziness factor $P(r_4)$ will be zero in most cases, which is actually desirable, otherwise heavy unnecessary oscillations will occur in the convergence curve near the end of the maximum iteration cycles as referred to (9). $v^{craziness}$ is chosen very small (=0.0001) as shown in Table 2. $r_4 \ge 0.5$ or, <0.5 is chosen to introduce equal probability of direction reversal of $v^{craziness}$ as referred to (12).

The design objective in this paper is to obtain the optimal combination of the IIR LP, HP, BP and BS filter coefficients, so as to acquire the maximum stop band attenuation with the least transition width. Here lies the author's contribution that this design objective has been attained by the proposed CRPSO technique. The values of the parameters used for RGA, PSO and CRPSO techniques are given in Table 2.

4 Simulation Results and Discussions

Extensive simulation study has been performed for comparison of optimization performances of three algorithms namely, RGA, PSO, and CRPSO, respectively, for the 8th order IIR LP, HP, BP and BS filter optimization problems. The design specifications followed for all algorithms are given in Table 1.

The values of the control parameters of RGA, PSO, and CRPSO are given in Table 2. Each algorithm is run for several times to get the best solution and the best results are reported in this paper. All optimization programs are run in MATLAB 7.5 version on core (TM) 2 duo processor, 3.00 GHz with 2 GB RAM.

Three aspects of the algorithms are investigated in this work namely, their accuracy, speed of convergence and stability. Figures 1, 4, 7 and 10 show the comparative gain plots in dB for the designed 8th order IIR LP, HP, BP and BS filters obtained for different algorithms. Normalized gain plots are shown in Figures 2, 5, 8 and 11 for the comparative study of 8th order IIR LP, HP, BP and BS filters. The best optimized numerator coefficients (b_k) and denominator coefficients (a_k) obtained after completion of predefined iteration cycles are reported in Tables 3, 6, 9 and 12. The values of statistical parameters for stop band attenuation in dB for 8th order IIR LP, HP, BP and BS filters designed using RGA, PSO, and CRPSO, respectively, are presented in Tables 4, 7, 10 and 13. Tables 5, 8, 11 and 14 show the maximum pass band ripple (normalized), maximum, minimum, average stop band ripple (normalized), and the transition widths for 8^{th} order IIR LP, HP, BP and BS filters designed using RGA, PSO and CRPSO, respectively. From the above tables and figures it can be explored that the proposed 8th order IIR filter designed with CRPSO attains the highest stop band attenuation in all cases with comparatively good figures for the rest of the parameters, such as stop band and pass band ripples, transition width etc. Figures 5, 8, 11 and 14 show the pole-zero plots for all 8th order IIR filters concerned with this paper for CRPSO based technique. These figures demonstrate the existence of poles within the unit circle which ensures the bounded input bounded output (BIBO) stability condition for the designed IIR filters.

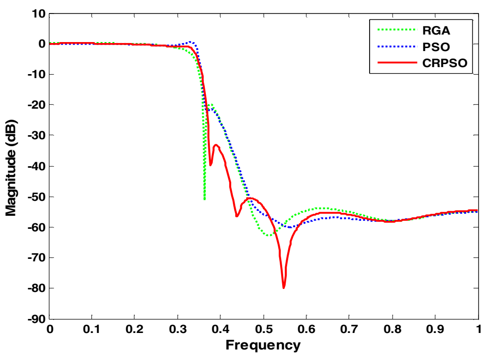

Fig. 1. Gain plots in dB for 8th order IIR LP filter using RGA, PSO and CRPSO

Table 1. Design Specifications of IIR LP, HP, BP and BS Filters

Filter Type	Pass band ripple (δ_p)	Stop band ripple (δ_s)	Pass band normalized edge frequency (ω_p)	Stop band normalized edge frequency (ω_s)
LP [19]	0.001	0.0001	0.35	0.40
HP	0.1	0.01	0.35	0.30
BP	0.1	0.01	0.35 and 0.65	0.3 and 0.7
BS	0.1	0.01	0.25 and 0.75	0.3 and 0.7

Table 2. Control Parameters of RGA, PSO and CRPSO

Parameters	RGA	PSO	CRPSO
Population size	120	25	25
Iteration cycles	400	100	100
Crossover rate	1	-	-
Crossover	Two Point Crossover	-	-
Mutation rate	0.01	-	-
Mutation	Gaussian Mutation	-	-
Selection	Roulette	-	-
Selection probability	1/3	-	-
C1, C2	-	2.05, 2.05	2.05, 2.05
$v_i^{\min}$, $v_i^{\max}$	-	0.01, 1.0	0.01, 1.0
wmax, wmin	-	1.0, 0.4	-
p_{cr}	-	-	0.3
$v^{craziness}$	-	-	0.0001

Table 3. Optimized Coefficients and Performance Comparison of Concerned Algorithms for 8th Order IIR LP Filter

Algorithms	Num. Coeff. (b_k)	Den. Coeff. (a_k)	Max. Stop Band Attenuation (dB)
RGA	0.0167 0.0059 0.0434 0.0234 0.0451 0.0302 0.0277 0.0120 0.0092	0.9996 -3.5213 7.1631 -9.4231 8.7904 -5.7905 2.6429 -0.7583 0.1066	20.000
PSO	0.0165 0.0060 0.0423 0.0237 0.0454 0.0286 0.0275 0.0122 0.0073	0.9996 -3.5201 7.1638 -9.4233 8.7894 -5.7906 2.6430 -0.7593 0.1072	21.5683
CRPSO	0.0169 0.0054 0.0424 0.0228 0.0456 0.0285 0.0275 0.0115 0.0092	0.9990 -3.5209 7.1621 -9.4221 8.7896 -5.7908 2.6431 0.7587 0.1072	33.1170

Table 4. Statistical Results for Stop Band Attenuation (dB) for 8th Order IIR LP Filter

Algorithm	Maximum	Mean	Variance	Standard Deviation
RGA	20.0000	42.9281	263.0129	16.2177
PSO	21.5683	44.5499	264.6049	16.2667
CRPSO	33.1170	48.3590	80.5940	8.9774

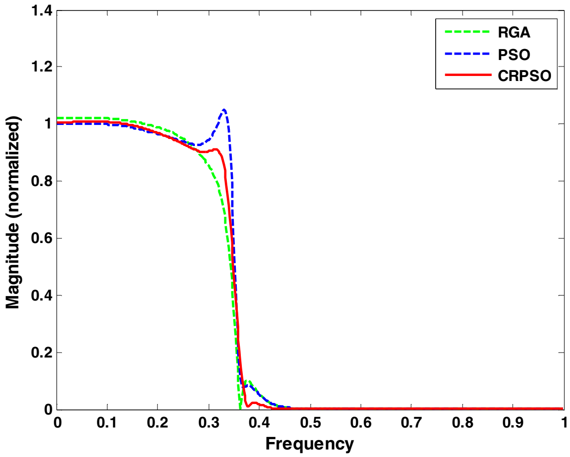

Fig. 2. Normalized gain plots for 8th order IIR LP filter using RGA, PSO and CRPSO

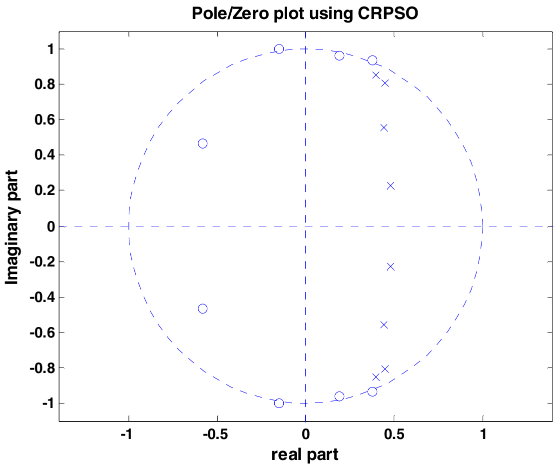

Fig. 3. Pole-zero plot of 8th order IIR LP filter using CRPSO

Table 5. Qualitatively Analyzed Results for 8th Order IIR LP Filter

Algorithm	Maximum Pass band ripple (normalized)	Stop band ripple (normalized)			Transition Width
		Maximum	Minimum	Average	
RGA	0.0214	0.1000	7.3286×10^{-4}	5.0366×10^{-2}	0.0341
PSO	0.0500	0.0835	1.0000×10^{-3}	4.2250×10^{-2}	0.0216
CRPSO	0.0086	0.0221	1.0000×10^{-4}	1.1100×10^{-2}	0.0370

Table 6. Optimized Coefficients and Performance Comparison of Concerned Algorithms for 8th order IIR HP filter

Algorithms	Num. Coeff. (b_k)	Den. Coeff. (a_k)	Max. Stop Band Attenuation (dB)
RGA	0.1250 -0.7092 1.9588 -3.3672 3.9090 -3.1264 1.6821 -0.5585 0.0881	0.9999 -2.1875 3.8221 -3.6220 2.9095 -1.3332 0.5678 -0.0861 0.0285	46.2199
PSO	0.1252 -0.7091 1.9587 -3.3671 3.9091 -3.1263 1.6821 -0.5584 0.0881	1.0001 -2.1874 3.8222 -3.6220 2.9096 -1.3333 0.5678 -0.0861 0.0285	47.7018
CRPSO	0.1252 -0.7091 1.9587 -3.3672 3.9090 -3.1263 1.6820 -0.5584 0.0883	1.0000 -2.1874 3.8223 -3.6220 2.9094 -1.3334 0.5679 -0.0861 0.0284	49.9710

Table 7. Statistical Results for Stop Band Attenuation (dB) for 8th Order IIR HP Filter

Algorithm	Maximum	Mean	Variance	Standard Deviation
RGA	46.2199	49.8589	13.2467	2.6391
PSO	47.7018	50.7807	9.4796	3.0789
CRPSO	49.9710	53.1421	3.8708	1.9674

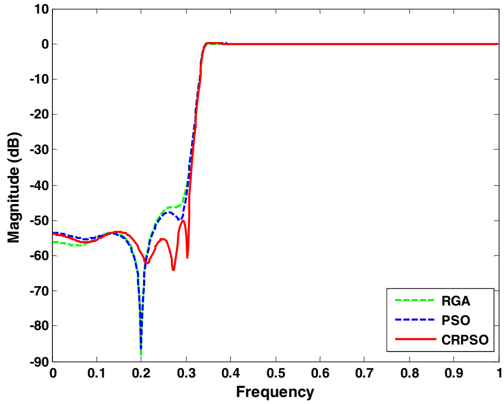

Fig. 4. Gain plots in dB for 8th order IIR HP filter using RGA, PSO and CRPSO

Table 8. Qualitatively Analyzed Results for 8th Order IIR HP Filter

Algorithm	Maximum Pass Band Ripple (normalized)	Stop Band Ripple (normalized)			Transition Width
		Maximum	Minimum	Average	
RGA	0.0146	0.48863×10^{-2}	0.39587×10^{-4}	0.24629×10^{-2}	0.0598
PSO	0.0186	0.41201×10^{-2}	0.47667×10^{-4}	0.20839×10^{-2}	0.0500
CRPSO	0.0356	0.31726×10^{-2}	6.2291×10^{-4}	0.18978×10^{-2}	0.0349

Table 9. Optimized Coefficients and Performance Comparison of Concerned Algorithms for 8th order IIR BP filter

Algorithms	Num. Coeff. (b_k)	Den. Coeff. (a_k)	Max. Stop Band Attenuation (dB)
RGA	0.1369 -0.0069 -0.0200 -0.0043 0.1897 0.0069 -0.0338 -0.0056 0.1253	0.9971 -0.0075 1.5866 -0.0094 1.7020 0.0000 0.8246 -0.0025 0.2247	18.2445
PSO	0.1274 0.0071 -0.0209 0.008 0.1857 0.0001 -0.0292 -0.0052 0.1299	0.9927 -0.002 1.5940 0.0029 1.6978 -0.0002 0.8079 -0.0034 0.2058	20.1389
CRPSO	0.1082 -0.0078 -0.0233 0.0018 0.1561 -0.0033 -0.0273 -0.0015 0.1037	1.0001 -0.0062 1.6899 0.0028 1.7556 -0.0023 0.8516 -0.0078 0.2038	22.7295

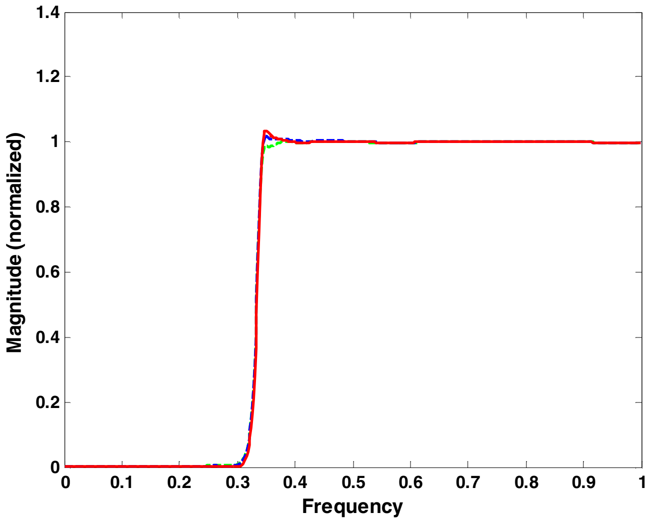

Fig. 5. Normalized gain plots for 8th order IIR HP filter using RGA, PSO and CRPSO

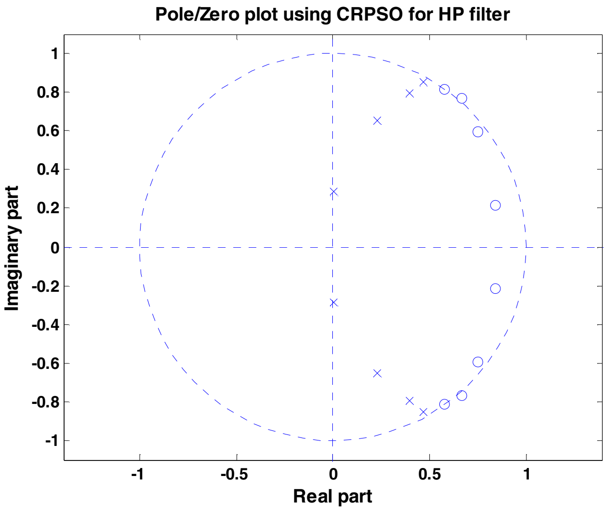

Fig. 6. Pole-zero plot of 8th order IIR HP filter using CRPSO

Table 10. Statistical Results for Stop Band Attenuation (dB) for 8th Order IIR BP Filter

Algorithm	Maximum	Mean	Variance	Standard Deviation
RGA	18.2445	20.3032	4.2382	2.0587
PSO	20.1389	21.4826	1.8054	1.3437
CRPSO	22.7295	24.4450	1.1011	1.0493

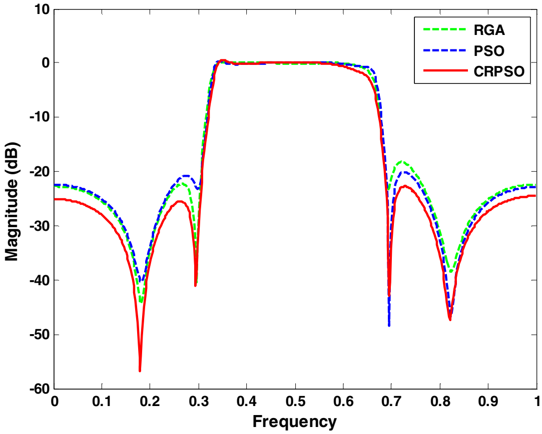

Fig. 7. Gain plots in dB for 8th order IIR BP filter using RGA, PSO and CRPSO

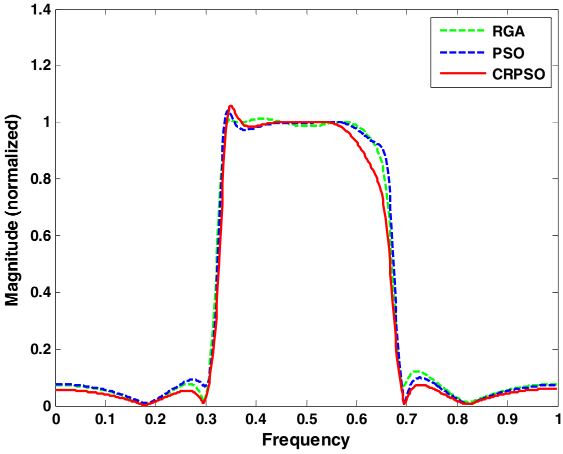

Fig. 8. Normalized gain plots for 8th order IIR BP filter using RGA, PSO and CRPSO

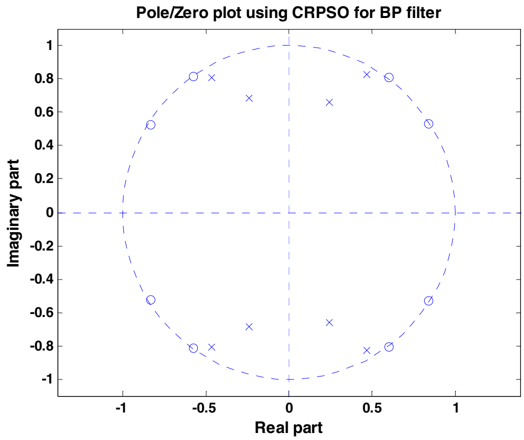

Fig. 9. Pole-zero plot of 8th order IIR BP filter using CRPSO

Table 11. Qualitatively Analyzed Results for 8th Order IIR BP Filter

Algorithm	Maximum Pass Band Ripple (normalized)	Stop band ripple (normalized)			Transition Width
		Maximum	Minimum	Average	
RGA	0.0134	12.24×10^{-2}	12.0000×10^{-3}	6.7200×10^{-2}	0.0311
PSO	0.0399	9.84×10^{-2}	3.7771×10^{-3}	5.1089×10^{-2}	0.0277
CRPSO	0.0578	7.30×10^{-2}	1.4313×10^{-3}	3.7200×10^{-2}	0.0409

Table 12. Optimized Coefficients and Performance Comparison of Concerned Algorithms for 8th order IIR BS filter

Algorithms	Num. Coeff.(b_k)	Den. Coeff. (a_k)	Max. Stop Band Attenuation (dB)
RGA	0.2269 -0.0189 0.5039 0.0170 0.6409 -0.0136 0.4866 0.0093 0.2189	1.0190 -0.0067 0.0968 0.0109 0.8671 0.0180 -0.0322 0.0177 0.1182	17.4734
PSO	0.2142 -0.0058 0.4833 -0.0008 0.6503 0.0097 0.4976 0.0041 0.2091	1.0073 -0.0069 0.0980 -0.0077 0.8902 -0.0073 -0.0198 -0.0048 0.1089	21.9740
CRPSO	0.2144 -0.0083 0.4817 -0.0055 0.6589 0.0001 0.4841 0.0050 0.2162	0.9959 -0.0061 0.0894 0.0040 0.8909 0.0038 -0.0273 -0.0003 0.1095	23.8659

Table 13. Statistical Results for Stop Band Attenuation (dB) for 8th Order IIR BS Filter

Algorithm	Maximum	Mean	Variance	Standard Deviation
RGA	17.4734	21.0867	13.0559	3.6133
PSO	21.9740	24.1658	4.8038	2.1918
CRPSO	23.8659	24.8641	0.5169	0.7190

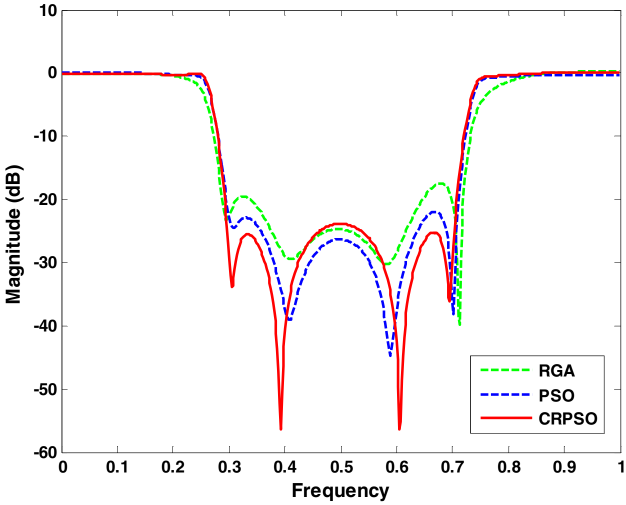

Fig. 10. Gain plots in dB for 8th order IIR BS filter using RGA, PSO and CRPSO

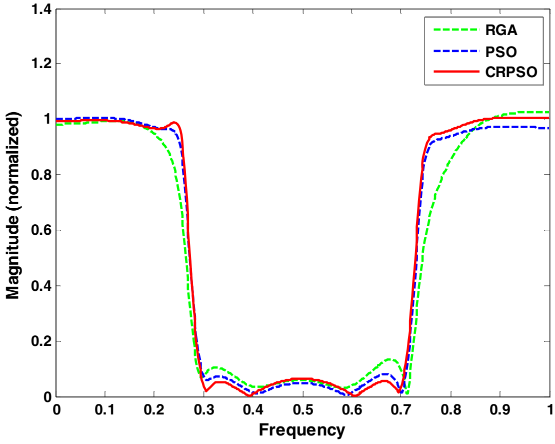

Fig. 11. Normalized gain plots for 8th order IIR BS filter using RGA, PSO and CRPSO

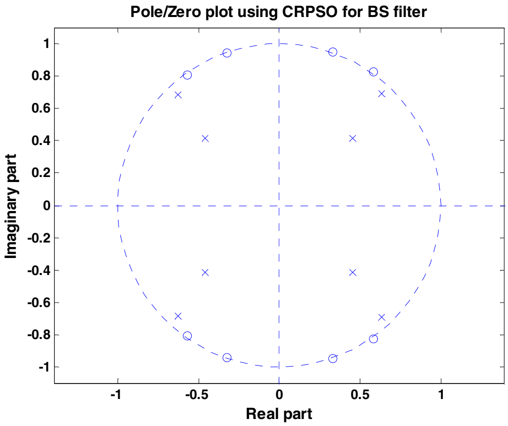

Fig. 12. Pole-zero plot of 8th order IIR BS filter using CRPSO

Table 14. Qualitatively Analyzed Results for 8th Order IIR BS Filter

Algorithm	Maximum Pass Band Ripple (normalized)	Stop Band Ripple (normalized)			Transition Width
		Maximum	Minimum	Average	
RGA	0.0268	13.38×10^{-2}	30.6000×10^{-3}	8.2200×10^{-2}	0.0535
PSO	0.0303	7.97×10^{-2}	5.8373×10^{-3}	4.2769×10^{-2}	0.0377
CRPSO	0.0344	6.41×10^{-2}	1.4978×10^{-3}	3.2799×10^{-2}	0.0410

Table 15. Comparison of Performance Criteria among algorithms published in relevant literatures

Reference	Proposed Algorithm	Filter Type	Order	Stop Band Attenuation (dB)	Max. Pass Band Ripple	Max. Stop Band Ripple	Transition Width
Luitel *et al.* [32]	DE-PSO	LP	9th	25	0.257	0.259	NR*
Luitel *et al.* [33]	PSO-QI	LP	9th	27	0.808	0.793	NR*
Karaboga *et al.* [34]	GA	LP	9th	14	NR*	NR*	NR*
Gao *et al.* [38]	DC	LP	6th	29	NR*	NR*	NR*
		HP	6th	42	NR*	NR*	NR*
Xue *et al.* [39]	GA	LP	7th	15	NR*	NR*	NR*
Present paper	CRPSO	LP	8th	33.1170	0.0086	0.00221	0.0370
		HP	8th	49.9710	0.0356	0.31726 e-2	0.0349
		BP	8th	22.7295	0.0578	7.30 e-2	0.0409
		BS [21]	8th	23.8659	0.0344	6.41 e-2	0.0410

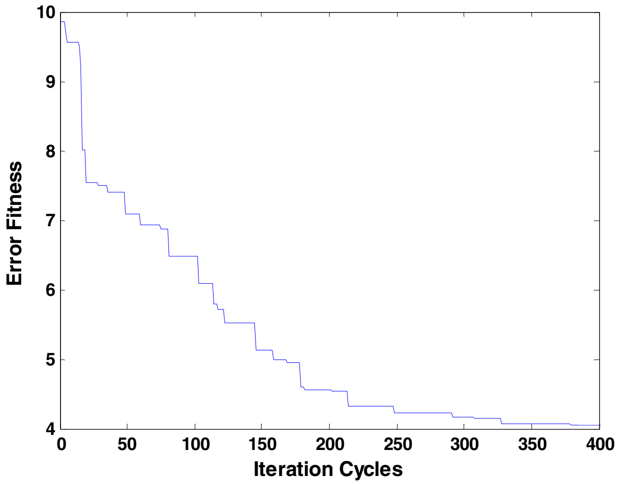

Fig. 13. Convergence profiles for RGA for 8th order IIR BS filter

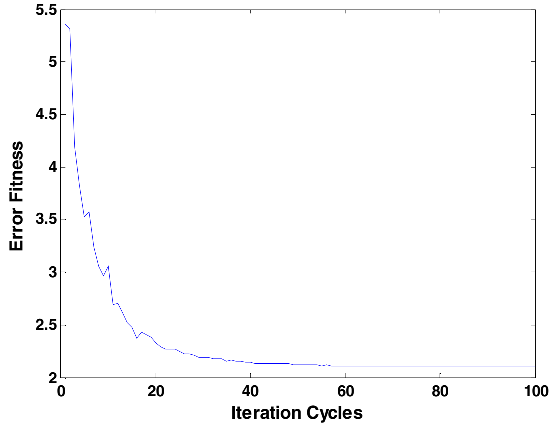

Fig. 14. Convergence profiles for PSO for 8th order IIR BS

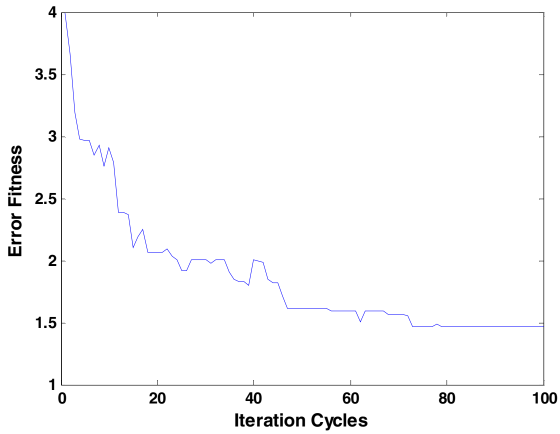

Fig. 15. Convergence profiles for CRPSO for 8th order IIR BS

Comparative study of results in terms of order, maximum attenuation, transition width, pass band and stop band ripples of IIR filters designed with different approaches adopted in different published literatures are reported in Table 15. Luitel *et al.* [32] proposed DE-PSO algorithm for the design of 9th order LP filter and maximum stop band attenuation, pass band and stop band ripples of approximately 25 dB, 0.257 and 0.259, respectively. Again, in [33], Luitel *et al.* proposed PSO-QI algorithm for the design of 9th order LP filter with the values of maximum stop band attenuation, pass band and stop band ripples of 27 dB, 0.808 and 0.793, respectively. Karaboga *et al.* proposed GA for the design of 9th order LP filter and the maximum value of stop band attenuation as 14 dB was reported in [34]. Gao *et al.* proposed differential cultural algorithm for the design of 6th order LP and HP filters in [38]. Maximum stop band attenuations of 29 dB and 42 dB for LP and HP filters, respectively, were reported there. Xue *et al.* also proposed GA for the design of 7th order LP filter and maximum stop band attenuation of 15 dB was reported in [39]. In the present paper CRPSO is proposed for the design of 8th order LP, HP, BP and BS filters. With this optimization technique, values of maximum stop band attenuation are 33.1170 dB, 49.9710 dB, 22.7295 dB and 23.8659 dB; maximum pass band ripple are 0.0086, 0.0356, 0.0578 and 0.0344; maximum stop band ripple are 0.221×10^{2}, 0.31726×10^{-2}, 7.30×10^{-2} and 6.41×10^{-2} and transition widths are 0.0370, 0.0349, 0.0409 and 0.0410 are obtained for LP, HP, BP and BS filters, respectively. So, CRPSO yields consistently higher stop band attenuation, lower stop band ripples with moderate control on the transition width and pass band ripples.

4.1 Comparison of Effectiveness and Convergence Profiles of RGA, PSO and CRPSO

Figures 13-15 depict the convergences of error fitness values obtained by RGA, PSO, and CRPSO for the 8th order IIR BS filter. Similar plots can also be obtained for the rest of the filters, which are not shown here.

As shown in Figures 13-15, RGA, PSO and CRPSO take 379, 85 and 79 iteration cycles to reach the error value of 4.043, 2.105 and 1.461, respectively, from which it can be concluded the CRPSO based approach finds the near sub-optimal solution of filter coefficients most fleetly among others with ensured grand minimum error value. With consideration of above facts and Figures 13-15, it can be easily inferred that the proposed CRPSO based optimization technique not only obtains the lowest error fitness value but also fast enough to achieve that. With a view to the above fact, it may finally be concluded that the performance of the CRPSO is the best among the three mentioned algorithms. All optimization programs are 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 stochastic optimization algorithm, CRPSO, is applied to the optimal design of 8th order low pass, high pass, band pass and band stop IIR digital filters. The proposed filter design algorithm, CRPSO, is based upon the PSO in which pitfalls of conventional PSO have been judiciously managed with the perspective of closely mimicking the behaviour of fish in a school. The optimal filters thus obtained meet the stability criterion and show the best attenuation characteristics with reasonably good transition widths. The CRPSO algorithm converges very fast to the best quality optimal solution and reaches the lowest minimum error fitness value in the shortest number of iteration cycles. Statistically analysed results obtained for the CRPSO also justify the potential of the proposed algorithm for the realization of digital IIR filters.

References

1. Oppenheim, A.V., Buck, R.W.: Discrete-Time Signal Processing. Prentice-Hall, Englewood Cliffs (1999)
2. Proakis, J.G., Manolakis, D.G.: Digital Signal Processing. Prentice-Hall, Englewood Cliffs (1996)
3. Lang, M.C.: Least-squares design of IIR filters with prescribed magnitude and phase responses and pole radius constraint. IEEE Trans. on Signal Processing 48(11), 3109–3121 (2000)
4. Jackson, L.B., Lemay, G.J.: A simple Remez exchange algorithm to design IIR filters with zeros on the unit circle. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, USA, vol. 3, pp. 1333–1336 (1990)
5. Hussain, Z.M., Sadik, A.Z., O'Shea, P.: Digital Signal Processing-An Introduction with MATLAB Applications. Springer (2011)
6. Antoniou, A.: Digital Signal Processing: Signals, Systems and Filters. McGraw Hill (2005)
7. Lu, W.S., Antoniou, A.: Design of digital filters and filter banks by optimization: a state of the art review. In: Proc. European Signal Processing Conf., Finland, vol. 1, pp. 351–354 (2000)
8. Tsai, J.T., Chou, J.H., Liu, T.K.: Optimal design of digital IIR filters by using hybrid Taguchi genetic algorithm. IEEE Trans. on Industrial Electronics 53(3), 867–879 (2006)
9. Yu, Y., Xinjie, Y.: Cooperative co-evolutionary genetic algorithm for digital IIR filter design. IEEE Trans. on Industrial Electronics 54(3), 1311–1318 (2007)
10. Chen, S., Istepanian, R., Luk, B.L.: Digital IIR filter design using adaptive simulated annealing. Digital Signal Processing 11(3), 241–251 (2001)
11. Kalinli, A., Karaboga, N.: Artificial immune algorithm for IIR filter design. Engineering Applications of Artificial Intelligence 18(8), 919–929 (2005)
12. Karaboga, N., Kalinli, A., Karaboga, D.: Designing digital IIR filters using ant colony optimization algorithm. Engineering Applications of Artificial Intelligence 17(3), 301–309 (2004)
13. Karaboga, N.: A new design method based on artificial bee colony algorithm for digital IIR filters. Journal of Franklin Institute 346(4), 328–348 (2009)
14. Panda, G., Pradhan, P.M., Majhi, B.: IIR system identification using cats swarm optimization. Expert Systems with Applications 38(10), 12671–12683 (2011)
15. Chen, S., Luk, B.L.: Digital IIR filter design using particle swarm optimization. Int. J. Modelling, Identification and Control 9(4), 327–335 (2010)
16. Das, S., Konar, A.: A swarm intelligence approach to the synthesis of two-dimensional IIR filters. Engineering Applications of Artificial Intelligence 20(8), 1086–1096 (2007)
17. Fang, W., Sun, J., Xu, W.: A new mutated quantum-behaved particle swarm optimization for digital IIR filter design. EURASIP Journal on Advances in Signal Processing, 1–7 (2009)
18. Pan, S.T., Chang, C.Y.: Particle swarm optimization on D-stable IIR filter design. In: IEEE World Congress on Intelligent Control Automation, Taipei, pp. 621–626 (2011)
19. Saha, S.K., Kar, R., Mandal, D., Ghoshal, S.P.: IIR filter design with craziness based particle swarm optimization technique. World Academy of Science, Engineering and Technology 60, 1628–1635 (2011)
20. Saha, S.K., Chaudhuri, A., Kar, R., Mandal, D., Ghoshal, S.P.: Optimization of IIR high pass filter using craziness based particle swarm optimization technique. In: Proc. IEEE Symposium on Humanities, Science and Engineering Research, pp. 401–406 (2012)
21. Saha, S.K., Rakshit, J., Kar, R., Mandal, D., Ghoshal, S.P.: Optimal digital stable IIR band stop filter optimization using craziness based particle swarm optimization technique. In: Proc. IEEE World Congress on Information and Communication Technology, pp. 774–779 (2012)

22. Singh, B., Dhillon, J.S., Brar, Y.S.: Predator prey optimization method for the design of IIR filter. WSEAS Transactions on Signal Processing 9(2), 51–62 (2013)
23. Saha, S.K., Kar, R., Mandal, D., Ghoshal, S.P.: Seeker optimization algorithm: application to the design of linear phase finite impulse response filter. IET Signal Process 6(8), 763–771 (2012)
24. Saha, S.K., Kar, R., Mandal, D., Ghoshal, S.P.: Bacteria foraging optimization algorithm for optimal FIR filter design. International Journal of Bio-Inspired Computation 5(1), 52–66 (2013)
25. Holland, J.H.: Adaptation in Natural and Artificial Systems. Univ. Michigan Press, Ann Arbor (1975)
26. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proc. IEEE Int. Conf. on Neural Network, vol. 4, pp. 1942–1948 (1995)
27. Eberhart, R.C., Shi, Y.: Comparison between genetic algorithm and particle swarm optimization. In: Porto, V.W., Saravanan, N., Waagen, D., Eiben, A.E. (eds.) EP 1998. LNCS, vol. 1447, pp. 611–616. Springer, Heidelberg (1998)
28. Sun, J., Fang, W., Xu, W.: A quantum-behaved particle swarm optimization with diversity-guided mutation for the design of two-dimensional IIR digital filters. IEEE Trans. on Circuits and Systems-II 57(2), 141–145 (2010)
29. Gao, Y., Li, Y., Qian, H.: The design of IIR digital filter based on chaos particle swarm optimization algorithm. In: IEEE Int. Conf. on Genetic and Evolutionary Computing, Hubei, pp. 303–306 (2008)
30. Ling, S.H., Iu, H.H.C., Leung, F.H.F., Chan, K.Y.: Improved hybrid particle swarm optimized wavelet neural network for modelling the development of fluid dispensing for electronic packaging. IEEE Trans. Ind. Electron. 55(9), 3447–3460 (2008)
31. Biswal, B., Dash, P.K., Panigrahi, B.K.: Power quality disturbance classification using fuzzy c-means algorithm and adaptive particle swarm optimization. IEEE Trans. Ind. Electron. 56(1), 212–220 (2009)
32. Luitel, B., Venayagamoorthy, G.K.: Differential evolution particle swarm optimization for digital filter design. In: IEEE Congress on Evolutionary Computation, pp. 3954–3961 (2008)
33. Luitel, B., Venayagamoorthy, G.K.: Particle swarm optimization with quantum infusion for the design of digital filters. In: IEEE Swarm Intelligence Symposium, USA, pp. 1–8 (2008)
34. Karaboga, N., Cetinkaya, B.: Design of minimum phase digital IIR filters by using genetic algorithm. In: Proc. IEEE 6th Nordic Signal Processing Symposium, Finland, pp. 29–32 (2004)
35. Saha, S.K., Kar, R., Mandal, D., Ghoshal, S.P.: Optimal IIR filter design using novel particle swarm optimization technique. International Journal of Circuits, Systems and Signal Processing, Springer 6(2), 151–162 (2012)
36. Saha, S.K., Kar, R., Mandal, D., Ghoshal, S.P.: Gravitational search algorithm: application to the optimal digital IIR filter design. Journal of King Saud University–Engineering Sciences (2013), doi: `http://dx.doi.org/10.1016/j.jksues.2012.12.003`
37. Mondal, S., Ghoshal, S.P., Kar, R., Mandal, D.: Craziness based Particle Swarm Optimization Algorithm for FIR Band Stop Filter Design. Journal of Swarm and Evolutionary Optimization 7, 58–64 (2012)
38. Gao, H., Diao, M.: Differential cultural algorithm for digital filters design. In: IEEE 2nd Int. Conf. on Computer Modeling and Simulation, pp. 459–463 (2010)
39. Xue, L., Rongchun, Z., Qing, W.: Optimizing the design of IIR filter via genetic algorithm. In: IEEE Int. Conf. on Neural Networks and Signal Processing, China, pp. 476–479 (2003)