Engineering Applications of Artificial Intelligence 70 (2018) 67–80

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Design of optimal high pass and band stop FIR filters using adaptive Cuckoo search algorithm

Shubhendu Kumar Sarangi [a], Rutuparna Panda [b,*], Pradeep Kumar Das [b], Ajith Abraham [c]

[a] Department of Electronics & Instrumentation Engineering, I.T.E.R, Siksha O Anusandhan University, Bhubaneswar, India
[b] Department of Electronics and Tele-communication Engineering, VSS University of Technology, Burla 768018, India
[c] Machine Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, WA-98071-2259, USA

ARTICLE INFO

Keywords:
Optimal FIR filter design
Adaptive cuckoo search

ABSTRACT

This paper presents an efficient design of digital FIR high pass and band stop filters using an adaptive cuckoo search algorithm (ACSA). The important features of ACSA are — (i) the step size is independent and (ii) it is accurately decided from the current fitness value. The step size is decided according to the current fitness value within the iteration process. This increases the convergence speed. The other five global optimizers are also used for optimization. The optimal solutions obtained by the ACSA are compared with the other global optimizers. CEC 2005 benchmark test functions are considered for the comparison. The results are compared in terms of the convergence speed, accuracy, deviation from the desired response, minimum stop-band attenuation and maximum pass-band attenuation. The statistical analysis, i.e. *t*-Test is performed to claim the superiority of the proposed approach. The simulation results presented in this paper reveal the fact that the performance of the ACSA is better than the other algorithms.

1. Introduction

Digital FIR filters are useful for communication and audio signal processing applications (Zhang et al., 2015). They play a vital role to process biomedical images such as Magnetic Resonance Imaging (MRI), Electroencephalogram, Electrocardiogram images (Miller et al., 1988). This has motivated the authors to design optimal FIR digital filters for different engineering applications.

Optimization techniques are used for minimization of error function. As a result, we get a filter response very close to the desired response. Optimization plays an important role in the reduction of cost, time consuming and energy consumption. Optimal use of resources makes the system efficient, accurate and cost efficient (Koziel and Yang, 2011; Yang, 2010; Yang et al., 2011). One can use various optimization methods to design and implement digital FIR filters (Rout, 2014). Here, we use Real Coded Genetic Algorithm (RCGA), Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC) Algorithm, Differential Evolution (DE), Cuckoo Search Algorithm (CSA) and Adaptive Cuckoo Search Algorithm (ACSA) for optimization of digital FIR filters. CEC 2005 benchmark test functions (García et al., 2009) are used to compare the global optimizers. In Real Coded Genetic Algorithm, the chromosomes are described by a vector of floating point numbers. It is faster and simpler than Genetic Algorithm. Note that in RCGA, there is no need to encode and decode the binary value. However, it is unable to determine the global optimum by means of speed of convergence and quality of the solution (Aggarwal et al., 2016). Particle swarm optimization (PSO) was proposed by Eberhart and Kennedy in 1995 (Koziel and Yang, 2011). It is initialized with a population of solution vectors which are random in nature and each solution vector is known as a particle. However, it has insufficient capability to surely achieve the global convergence. Further, the convergence speed is also slow due to the involvement of more parameters.

Cuckoo Search Algorithm (CSA) is a global optimization technique based upon the behavior of cuckoos, which was proposed by Yang and Deb (2009, 2010, 2013). CSA consumes 15%–30% less time as compared to the crossover bacterial foraging, bacterial foraging and Genetic algorithm optimization techniques. CSA is more efficient, faster and its performance is better as compared to the GA (Ababneh and Bataineh, 2008; Ahmad and Antoniou, 2006), PSO (Kennedy and Eberhart, 1995; Luitel and Venayagamoothy, 2010; Ababneh and Bataineh, 2008), adaptive GA (Boudejelaba et al., 2014); and craziness based PSO (Mandal et al., 2012; Kar et al., 2012). The artificial bee colony (ABC) algorithm (Dan, 2016) has been used for frequency sampling based FIR

* Corresponding author.
E-mail addresses: shubhendu1977@gmail.com (S.K. Sarangi), r_ppanda@yahoo.co.in (R. Panda), pdas391@gmail.com (P.K. Das), ajith.abraham@ieee.org (A. Abraham).

https://doi.org/10.1016/j.engappai.2018.01.005
Received 7 September 2017; Received in revised form 10 January 2018; Accepted 11 January 2018
Available online 3 February 2018

filter design, where optimal frequency sampling points were obtained. Use of ABC algorithm for FIR filter design was also reported in Singh and Kaur (2014). Speed based ABC algorithm was presented for optimal 2-D FIR filter design (Dhabal and Venkateswaran, 2017). Multiobjective ABC algorithm (Raju and Kwan, 2017) was used for FIR filter design and results were compared with the differential evolution (DE). These algorithms involve more parameters and are not adaptive in nature. Hence, we need to choose an adaptive algorithm with less number of parameters. An attempt is made in this study.

For a continuous optimization problem like welded beam design and spring design problem, CSA is preferred over other optimization algorithms — GA, adaptive GA, BFO, PSO and COBFO for its remarkable performance (Gandomi et al., 2013, 2012; Zheng and Zhou, 2012).

CSA can be efficiently used to an optimized multilevel thresholding problem in the field of satellite image segmentation (Bhandari et al., 2014). One can efficiently extract features of graphic object by using a Cuckoo search algorithm (Wozniak et al., 2016). Multilevel thresholding of an image can be achieved by optimizing edge magnitude of the image using a Cuckoo search algorithm (Panda et al., 2013a).

The ACSA has been developed by Naik et al. (2015) and Naik and Panda (2016). In ACSA, the step size is adaptively determined during the iteration process of the algorithm. Hence, the performance of ACSA is more efficient than CSA. How fast we can reach the local minima or global minima is very crucial in an optimization algorithm. In this sense, the speed of convergence is faster in ACSA (Naik et al., 2015; Naik and Panda, 2016). In ACSA, search pathway is adaptively decided within the minimum time to find an optimum solution. Further, it requires less parameters. Hence, in this work, ACSA is preferred to find optimal filter coefficients.

In this paper, we focus on integrated, analytical and comparative study of ACSA, CSA, PSO and RCGA for the design of digital FIR high-pass and stop-band filter. For achieving high accuracy, the optimal filter should have a low pass band ripple and optimum stop band attenuation. Section 2 focus on the FIR filter design model. In Section 3, we give emphasis on the implementation of popular optimization algorithms RCGA, PSO, ABC, DE, CSA and ACSA to design digital FIR filters. Comparative analysis about specific implementation problems are also discussed in Section 3. The performance evaluation of ACSA using the standard CEC 2005 benchmark functions is shown in Section 4. The simulation results, obtained by using RCGA, PSO, ABC, DE, CSA and ACSA, are presented in Section 5. Finally, we conclude the complete work in Section 6.

2. FIR filter design formulation

Digital filters are preferred over analog filters for its thermal stability, high accuracy, compact size, flexibility in the reconfiguration, superior performance to cost ratio and perfect reproducibility. They are classified into two types, FIR (Finite Impulse Response) filters and IIR (Infinite Impulse Response) filters (Sarangi et al., 2014). Non-recursive FIR filter is naturally stable, less sensitive to finite word length effects, easy to obtain a linear phase response and convenient to implement (Liu et al., 2010). Whereas, recursive IIR filters provide flat frequency response with lower order. These are faster than non-recursive FIR filters (Antoniou, 2005; Panda et al., 2013b, 2011; Das and Konar, 2006).

Digital filters play a vital role in signal processing and communication systems for reduction of channel width, reduction of random noise, elimination of spectral leakage, mitigating the inter symbol interference, etc. We can use a digital filter bank in both wired and wireless communication systems (Vaidyanathan, 2006). Digital filter characteristics can be changed or adapted by changing the contents of registers. One can expand Canonic Signed Digit (CSD) representation to design digital filters by utilizing masking of frequency response (Yu and Lim, 2002).

Digital FIR filter plays a significant role in biomedical, communication and audio signal processing applications (Lita et al., 2004; Lian and Wei, 2005; Zhang et al., 2015). The quality of the distorted image containing noise can be improved by using digital filters (Tak-Shing et al., 2012). Digital filter can be used for synthesis and analysis of speech signals (Zahorian and Gordy, 1983). It can also be used for discarding random noises in seismic data (Shi-Wei, 2011). We use various window methods for implementation of digital filters. Window function (Hamming, Kaiser, etc.) is selected according to the requirements of the pass-band and stop-band ripple, stop-band attenuation and transition width. We use a window function to approximate the ideal impulse response of infinite length to an actual impulse response of finite length (Parks and McCllean, 1972; Rabiner, 1973; Litwin, 2000).

In this paper, we focus on the non-recursive FIR filter design due to its popularity, inherent stability, linear phase characteristics over a broad range of frequencies and easy implementation. The Z-Transform of the impulse response of the FIR filter is

$$H(Z) = h(0) + h(1)Z^{-1} + h(2)Z^{-2} + \cdots + h(N)Z^{-N}. \tag{1}$$

We can represent $H(z)$ as

$$H(Z) = \sum_{n=0}^{N} h(n)(Z)^{-n} \tag{2}$$

where $h(n)$ is the impulse response of the FIR filter, N is the order of the filter and $N+1$ is the number of coefficients.

The frequency response of the FIR filter can be represented as

$$H(\omega) = \sum_{n=0}^{N} h(n)(e)^{-j\omega n}. \tag{3}$$

The magnitude response of the ideal HP FIR filter is represented as

$$H_1(\omega) = \begin{cases} 0 & \text{for } 0 \le \omega \le \omega_c \\ 1 & \text{for } \omega_c \le \omega \le \pi. \end{cases} \tag{4}$$

The magnitude response of the ideal BS FIR filter is represented as

$$H_2(\omega) = \begin{cases} 0 & \text{for } \omega_1 \le \omega \le \omega_2 \\ 1 & \text{for otherwise.} \end{cases} \tag{5}$$

where ω_c is the cut-off frequency of the HP filter; ω_1 is the lower stop band frequency and ω_2 is the upper stop band frequency.

We consider Type-1 linear phase FIR filter. This has the odd length and the symmetrical impulse response.

The impulse response is symmetric, if the following condition is satisfied

$$h(n = h(N-n)), \quad 0 \le n \le N. \tag{6}$$

The amplitude response of the FIR filter is given as

$$A(\omega) = h(N/2) + 2\sum_{n=1}^{N/2} h(N/2 - n)\cos(\omega n). \tag{7}$$

We can represent $A(\omega)$ as

$$A(\omega) = h(M) + 2\sum_{n=1}^{N/2} h(M - n)\cos(\omega n). \tag{8}$$

As the desired response possesses a zero phase characteristic, we get

$$\begin{aligned} A(\omega) &= \sum_{n=1}^{N} a(n)\cos(\omega n) \\ a(0) &= h(N/2) \end{aligned} \tag{9}$$

where $a(n) = h(N/2 - n) 1 \le n \le N/2.$

The error function $E(\omega)$ of the HP filter is

$$\begin{aligned} C &= \sum_{n=1}^{N/2} A(\omega) - H_1(\omega) \\ |E(\omega)| &= \int_0^{\pi} C^2 d\omega. \end{aligned} \tag{10}$$

The error function $E(\omega)$ of the BS filter is

$$D = \sum_{n=1}^{N/2} A(\omega) - H_2(\omega)$$
$$|E(\omega)| = \int_0^{\pi} D^2 d\omega. \tag{11}$$

Note that the error function $E(\omega)$ is the fitness function. This function is minimized using the optimization algorithms. The aim of each algorithm is to minimize the fitness function and optimize the filter coefficients.

3. Optimization techniques

This section concentrates on the study and analysis of six popular optimization techniques, i.e., RCGA, PSO, ABC, DE, CSA and ACSA.

3.1. Real Coded Genetic Algorithm (RCGA)

The RCGA is an optimization technique in which we have to initialize the actual chromosome string vector of N_p population. Each string vector containing $(N/2 + 1)$ number of coefficients of the impulse response to achieve positive symmetric linear phase characteristics. Where 'N' represents the order of FIR filters.

Steps for implementing FIR filter using the Real Coded Genetic Algorithm are as follows:

1. Initialize the string vector.
2. Compute the error fitness of each string.
3. Select elite strings for optimizing the error fitness values from the minimum value.
4. The elite string is copying over the non-selected string.
5. Generate the off-springs by utilizing crossover and mutation.
6. Update the genetic cycle.
7. Continue Step 2 to step 6 until reach the stopping criteria (maximum number of iteration).

3.2. Particle Swarm Optimization (PSO)

Particle Swarm Optimization is a population based optimization technique which having the capability to adopt both global and local exploration. It was inspired by the behavior of swarms in bird flocking and fish schooling. Swarm of particles acts as a potential solution for design problem. Each particle is associated with velocity and position vector. Velocity of particle is updated at each iteration depending upon the local best value, present velocity and global best value. The position of each particle is updated by adding its velocity to the present position of the particle.

Steps for implementing FIR filter using the Particle Swarm Optimization are as follows:

1. Set the population size, $N_p = 100$; maximum iterations, $M = 1000$ and the order of the filter is 20. Fixing the values of learning parameters L_1 and L_2 as 2. Filter coefficients having minimum value are -2 and maximum value is 2.
2. Create swarm of the particles having initial position vector, $P^{k=0}$ and initial velocity vector, $V^{k=0}$.
3. Compute the fitness values of each particle
4. Evaluate the initial personal best solution vectors (p_{best}) and the group best solution vector (g_{best}).
5. Update the velocity of particle vectors by using
$$V_i^{k+1} = w * V_i^k + \alpha * L_1 * (g_{\text{best}}^k - P_i^k) + \beta * L_2 * (p_{\text{best}}^k - P_i^k) \tag{12}$$
where L_1 and L_2 represent the learning parameters.
6. Update the position of particle vectors by using
$$P_i^{k+1} = P_i^k + V_i^{k+1}. \tag{13}$$
7. Updating the p_{best} and g_{best} vector on the basis of updated fitness values.
8. Iteration continues from step 4 to step 7 till the stopping criteria (maximum number of iteration = 1000) is reached. The final value of the best is used as the optimum filter coefficient vector for FIR filter design.

3.3. Artificial Bee Colony (ABC)

The ABC algorithm is based on the fundamental principle of self-organization and division of labor. More features of it is found in Karaboga and Basturk (2007).

Steps for implementing FIR filter using the artificial bee colony algorithm are as follows:

1. Simulate behavior of real bees for solving FIR filter coefficient optimization problem.
2. Consider 3 groups of bees: (i) employed bees, (ii) onlookers and (iii) scouts.
3. Choose population size $N = 100$.
4. Initialize first half of the colony consisting of the employed bees. Also initialize the second half of the colony by the onlookers.
5. Set the number of the employed bees = the number of food sources (available round the hive).
6. The employed bee befits a scout, when its food source is finished.

3.4. Differential Evolution (DE)

DE is a stochastic and population-based optimization procedure, which has been suggested by Storn and Price (1997).

Steps for implementing FIR filter using the DE are as follows:

1. Select the size of the population $N = 100$.
2. Choose the parameter vectors
$$x_{i,G} = [x_{1,i,G}, x_{2,i,G}, \ldots, x_{D,i,G}] i = 1, 2, \ldots, N,$$
where G is the generation number.
3. Each of the N parameter vectors undergoes mutation, recombination and selection.
4. For a specified parameter vector $x_{i,G}$, 3 vectors $x_{r1,G}$, $x_{r2,G}$ and $x_{r3,G}$ are randomly selected such that the indices i, $r1$, $r2$ and $r3$ are distinctive.
5. The weighted difference of two of the vectors is added to the third vector,
$$v_{i,G+1} = x_{r1,G} + F(x_{r2,G} - x_{r3,G}). \tag{14}$$
6. Choose mutation factor $F = 0.5$.
7. Choose $CR = 0.1$, where CR is the probability of the elements of the donor vector entering into the trial vector.
8. Continue mutation, recombination and selection, until the stopping criterion is satisfied.

3.5. Cuckoo Search Algorithm (CSA)

Cuckoo Search Algorithm (CSA) is a method of global optimization based upon the behavior of cuckoo birds. It was developed by Yang and Deb in the year 2009 (Yang and Deb, 2009, 2010). Cuckoo lays its eggs in the nest of the other host birds. Interestingly, the host bird takes care the cuckoo's egg, assuming it as its own egg. If a host bird identify the eggs are not its own, then it will abandon its nest and built a new one; otherwise throw those eggs away (Chakraverty and Kumar, 2011; Aggarwal et al., 2016). The objective of the CSA is to minimize the fitness function and maximize the performance.

For simplicity and easy calculation, we have to follow the following rules:

1. Each Cuckoo is permitted to lay one egg at a time and it is arbitrarily placed in the nest of the host bird.

2. The supreme nest having good quality of eggs will go to the next generation.
3. There is a fixed number of available host nest. The host nest identifies the cuckoo eggs with a fixed probability ($0 \leq p \leq 1$).

Assume that there are an N number of Cuckoos present. The search space of the ith cuckoo of n dimension is represented as $X_i = x_i^1, x_i^2, \ldots, x_i^d, \ldots, x_i^N$ with $i = 1, 2, \ldots, N$. Note that d is the dimension of the problem under consideration and $d < N$.

The new search space for ith Cuckoo at time t is computed as

$$X_i(t+1) = X_i(t) + \alpha * Levy(\gamma) \tag{15}$$

where $X_i(t+1)$ represents the new search space, $X_i(t)$ denotes the previous search space, and $Levy(\gamma)$ represents a random walk through Levy flight and α is a constant which depends upon the dimension of the search space (Barthelemy et al., 2008). Generally, we assume α is equal to 1. Note that the Levy distribution is considered as a continuous probability distribution function for a non-negative random variable. It is a stable distribution, assuming $\alpha = 1$. More or less, this is a distinct case of the inverse-gamma distribution. Some of the examples are — the Brownian motion, the length of the path followed by a photon in a turbid medium etc.

The current location is the only key factor which determines the random walk of the subsequent position. The random walk is obtained from the step of the Levy distribution. Levy step size can be represented as

$$Levy(\gamma) = \frac{u}{|Z|^{\frac{1}{(\gamma-1)}}}. \tag{16}$$

We can get u and z values following a normal distribution and generally varies within the range [13] .

$$Step \quad size = \beta * Levy(\gamma). \tag{17}$$

We take the value of β such that the Levy step should not be aggressive.

For the design of the HP and SB FIR filters, fitness function is the error function defined in Eqs. (10) and (11), respectively.

Steps for implementing FIR filter using CSA are as follows:

Step 1: Initialization

A. Set the number of host nests N and probability of finding alien eggs P_a and maximum number of iteration M.
   In this paper, for the design of FIR filter, we initialize $N = 100$, $P_a = 0.25$ and $M = 1000$.
B. For N dimensional problem, N number of host nests are represented as
   $X_i = x_i^1, x_i^2, \ldots, x_i^d, \ldots, x_i^N$
   for $i = 1, 2, \ldots, N$.
C. Take error objective function represented in Eq. (10) or (11) as a fitness function 'F' for the design of HP or BS FIR filter, respectively.

Step 2: Iterative Algorithm

A. Select a Cuckoo 'i' randomly by using levy flight and compute its fitness $F(t)$.
B. Select a new nest 'k' randomly from 'N' nests and compute its fitness function F_k.
C. Compare the fitness values. If $F_k < F_i$ then replace the egg 'k' with egg 'i'.
D. The worst nest are eliminated depending upon the probability $p_a = 0.25$ and new one are built.
E. Arrange the solution in ascending order and identify the current best.
F. Update the value of 't' i.e., $t = t + 1$.
G. Repeat steps B–F until the stopping criteria ($t = t_{\max}$) is satisfied and find out the best nest which is used as the optimum filter coefficients for the FIR filter design.

3.6. Adaptive Cuckoo Search Algorithm (ACSA)

CSA is a heuristic search optimization algorithm in which the search space is updated by a Levy step which obtained from a Levy distribution. There is no control over the step size during the process of iteration while achieving global optimization by using a standard Cuckoo search algorithm. The search space of CSA is associated with a random walk nature for which CSA may consume more time. In Adaptive Cuckoo Search Algorithm (ACSA), search space becomes independent of the Levy distribution.

In ACSA, the step size not only depends upon the current generation but also depends upon the individual nest's fitness value in the search space. Step size of ith nest $S_i(t+1)$ can be represented as

$$S_i(t+1) = \left(\frac{1}{t}\right)^G \tag{18}$$

where

$G = \left|\frac{BF(t) - Fi(t)}{BF(t) - WF(t)}\right|$

t = Generation
$F_i(t)$ = Fitness value of nest 'i'
$BF(t)$ = Best fitness value
$WF(t)$ = Worst fitness value.

The step size is inversely proportional to the generation of Cuckoo search raised to Gth power. Thus, the step size becomes minimum when we get the global optimum. Eq. (18) designates that the step size can be made changed and is decided from the current fitness value.

The new search space for the ith nest in the generation is computed as

$$X_i(t+1) = X_i(t) + randn * S_i(t+1). \tag{19}$$

There is no need to define any initial parameter while using an ACSA algorithm for the optimization purpose. As a result, ACSA requires less parameters for which it becomes faster than CSA.

Steps for implementing FIR filter using ACSA are as follows:

Step 1: Initialization

A. Set the number of host nests N and probability of finding alien eggs P_a and maximum number of iteration M.
   In this paper, for the design of FIR filter we initialize $N = 100$, $P_a = 0.25$ and $M = 1000$.
B. For N dimensional problem, N number of host nests represented as
   $X_i = x_i^1, x_i^2, \ldots, x_i^d, \ldots, x_i^N$
   for $i = 1, 2, \ldots, N$.
C. Take error objective function represented in Eq. (10) or (11) as a fitness function 'F' for the design of HP or BS FIR filter, respectively.

Step 2: Iteration

A. Find the best fitness value $BF(t)$ and the worst fitness value $WF(t)$.
B. Compute the size of each step $S_i(t+1)$ applying Eq. (18).
C. Compute the new position of Cuckoo nests (new search space) applying Eq. (19).
D. Compute the fitness function of host nest $F_i(X_i)$ where $i = 1, 2, \ldots N$.
E. Then select randomly a nest k among N.
F. If $F_k < F_i$ then replace ith nest by the kth nest.
   Stop.
G. Eliminate the worst nest depending upon the probability $P_a = 0.25$ and new one are built.

Table 1
Unimodal CEC 2005 test functions used with $d = 30$.

Function	Description	Range of 'X'	f_bias	Global optimal
Shifted sphere function	$F_1(X) = \sum_{i=1}^{d} Z_i^2 + f\_bias_1$	$[-100, 100]^d$	$F_1(X^*) = f\_bias_1 = -450$	$X^* = 0$
Shifted Schwefel's problem 1.2	$F_2(X) = \sum_{i=1}^{d} \left(\sum_{j=1}^{i} Z_j\right)^2 + f\_bias_2$	$[-100, 100]^d$	$F_2(X^*) = f\_bias_2 = -450$	$X^* = 0$
Shifted rotated high conditioned elliptic function	$F_3(X) = \sum_{i=1}^{d} (10^6)^{\frac{i-1}{d-1}} Y_i^2 + f\_bias_3$	$[-100, 100]^d$	$F_3(X^*) = f\_bias_3 = -450$	$X^* = 0$
Shifted Schwefel's problem 1.2 with noise in fitness	$F_4(X) = \left(\sum_{i=1}^{d} \left(\sum_{j=1}^{i} Z_j\right)^2\right) * (1 + 0.4\,\lvert N(0,1)\rvert) + f\_bias_4$	$[-100, 100]^d$	$F_4(X^*) = f\_bias_4 = -450$	$X^* = 0$
Schwefel's problem 2.6 with global optimum on bounds	$F_5(X) = \max\left(\lvert A_i X - B_i\rvert\right) + f\_bias_5$	$[-100, 100]^d$	$F_5(X^*) = f\_bias_5 = -310$	$X^* = 0$

Where
$Z = X - O$
$X = [X_1, X_2, \dots, X_d]$
$O = [O_1, O_2, \dots, O_d]$
$Y = (X - O) * M$
'd' represent dimensions. 'O' is the shifted global optimum.
M is the orthogonal matrix.

Table 2
Basic multimodal CEC 2005 test functions used with $d = 30$.

Function	Description	Range of 'X'	f_bias	Global optimal
Shifted Rosenbrock's Function	$F_6(X) = \sum_{i=1}^{d-1} \left(100\left(Z_i^2 - Z_{i+1}\right)^2 + \left(Z_i - 1\right)^2\right) + f\_bias_6$	$[-100, 100]^d$	$F_6(X^*) = f\_bias_6 = 390$	$X^* = 0$
Shifted rotated Griewank's Function without bound	$F_7(X) = \sum_{i=1}^{d} \frac{Z_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{Y_i}{\sqrt{i}}\right) + 1 + f\_bias_7$	$[0, 600]^d$	$F_7(X^*) = f\_bias_7 = -180$	$X^* = 0$
Shifted rotated Ackley's with global optimum on bounds	$F_8(X) = -20\exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} P_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d} \cos\left(2\pi P_i\right)\right) + 20 + e + f\_bias_8$	$[-32, 32]^d$	$F_8(X^*) = f\_bias_8 = -140$	$X^* = 0$
Shifted Rastrigin's Function	$F_9(X) = \sum_{i=1}^{d} \left(\left(Z_i\right)^2 - 10\cos\left(2\pi Z_i\right) + 10\right) + f\_bias_9$	$[-5, 5]^d$	$F_9(X^*) = f\_bias_9 = -330$	$X^* = 0$
Shifted Rotated Rastrigin's Function	$F_{10}(X) = \sum_{i=1}^{d} \left(\left(R_i\right)^2 - 10\cos\left(2\pi R_i\right) + 10\right) + f\_bias_{10}$	$[-5, 5]^d$	$F_{10}(X^*) = f\_bias_{10} = -330$	$X^* = 0$
Shifted Rotated Weierstrass' Function	$F_{11}(X) = \sum_{i=1}^{d} \left(\sum_{k=0}^{20} 0.5^k \cos\left(2\pi 3^k \left(U_i + 0.5\right)\right)\right) - d\sum_{k=0}^{20} 0.5^k \cos\left(2\pi 3^k .0.5\right) + f\_bias_{11}$	$[-0.5, 0.5]^d$	$F_{11}(X^*) = f\_bias_{11} = 90$	$X^* = 0$
Schwefel's problem 2.13	$F_{12}(X) = \sum_{i=1}^{d} \left(A_i - B_i(X)\right)^2 + f\_bias_{12}$	$[-\pi, \pi]^d$	$F_{12}(X^*) = f\_bias_{12} = -460$	$X^* = 0$

$Y = (X - O) * M$
$M = M'(1 + 0.3\,\lvert N(0,1)\rvert)$
M is the linear transformation matrix, condition number = 3
$P = (X - O) * Q$
Q is the linear transformation matrix, condition number = 100
$R = (X - O) * S$
S is the linear transformation matrix, condition number = 2
$U = (X - O) * V$
V is the linear transformation matrix, condition number = 5
$A_i = \sum_{i=1}^{d} \left(a_{ij} \sin\left(\alpha_j\right) + b_{ij} \cos\left(\alpha_j\right)\right)$
$B_i(X) = \sum_{i=1}^{d} \left(a_{ij} \sin\left(X_j\right) + b_{ij} \cos\left(X_j\right)\right)$.

Table 3
Expanded functions used with $d = 30$.

Function	Description	Range of 'X'	f_bias	Global optimal
Shift Expanded Griewank's plus Rosenbrock's Function	$F_{13}(X) = F_7\left(F_6\left(Z_1, Z_2\right)\right) + F_7\left(F_6\left(Z_2, Z_3\right)\right) + \cdots + F_7\left(F_6\left(Z_{d-1}, Z_d\right)\right) + F_7\left(F_6\left(Z_d, Z_1\right)\right) + f\_bias_{13}$	$[-5, 5]^d$	$F_{13}(X^*) = f\_bias_{13} = -130$	$X^* = 0$
Shift Expanded Scaffer's Function	$F_{14}(X) = EF\left(Z_1, Z_2, \dots .Z_d\right) = F\left(Z_1, Z_2\right) + F\left(Z_2, Z_3\right) + \cdots + F\left(Z_{d-1}, Z_d\right) + F\left(Z_d, Z_1\right) + f\_bias_{14}$	$[-100, 100]^d$	$F_{14}(X^*) = f\_bias_{14} = -300$	$X^* = 0$

Where $F_7(X)$ is the Griewank's Function
$F_7(X) = \frac{Z_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{Y_i}{\sqrt{i}}\right) + 1$
$F_6(X)$ is Rosenbrock's Function
$F_6(X) = \sum_{i=1}^{d-1} \left(100\left(Z_i^2 - Z_{i+1}\right)^2 + \left(Z_i - 1\right)^2\right)$
$F(W, X) = 0.5 + \frac{\left(\sin 2\left(\sqrt{W^2 + X^2}\right) - 0.5\right)}{\left(1 + 0.001\left(X^2 + W^2\right)\right)^2}$.

H. Update the value of 't' i.e., $t = t + 1$.
I. Repeat steps A–H until the stopping criteria ($t > t_{\max}$) is satisfied and find out the best nest which is used as the optimum filter coefficients for the design of FIR filter.

4. Performance evaluation of the ACSA

In this section, the superiority of ACSA over CSA has been claimed. In this simulation experiment, we consider 14 numbers of different benchmark test functions from CEC 2005 benchmark test functions (García et al., 2009). Five different CEC 2005 unimodal test functions are displayed in Table 1. In this case, the convergence rate is imperative in order to validate an optimization algorithm. Another 7 different CEC 2005 multimodal test functions are given in Table 2. Note that it is very challenging to optimize these test functions. Two numbers of expanded benchmark test functions (Griewank's Function and Rosenbrock's Function) are displayed in Table 3.

Table 4 displays the control parameters used for the performance measure of both the ACSA and CSA algorithm. From Table 4, it is observed that the ACSA has less parameters than CSA. For this simulation experiment, 80,000 function evaluations, for different functions discussed in Tables 1 and 2, are considered. However, 8000 function evaluations, for different functions displayed in Table 3, are considered. It is noteworthy to mention here that the procedure is run for 50 times each. From extensive simulation experiments only, we decide the

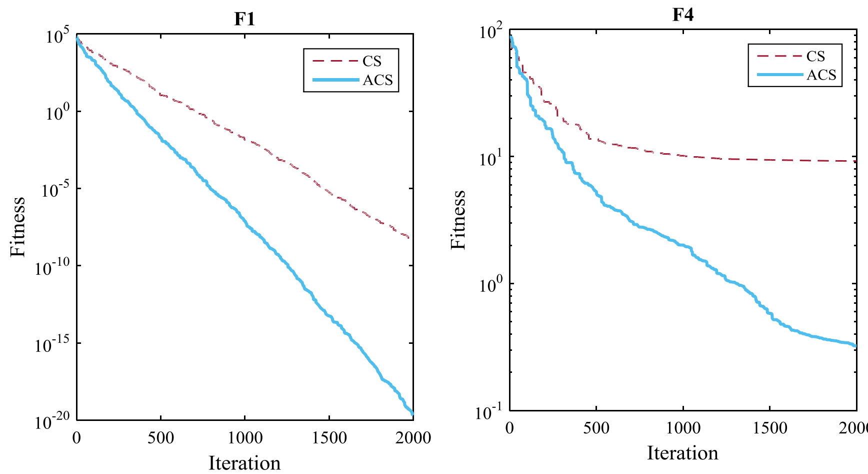

Fig. 1. Iteration vs Fitness values of the two unimodal CEC 2005 benchmark functions.

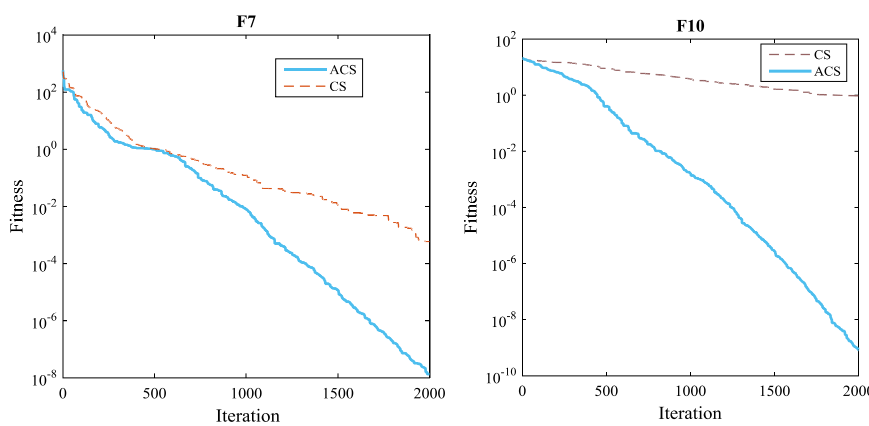

Fig. 2. Iteration vs Fitness values of the two multimodal CEC 2005 benchmark functions.

Table 4
Parameters considered for CEC 2005 test functions.

Algorithm	Parameters
ACSA	$N = 25$, $p_a = 0.25$, $t_{MAX} = 2000$
CSA	$N = 25$, $p_a = 0.25$, $\alpha = 1$, $\lambda = 1.5$, $t_{MAX} = 2000$

number of function evaluations in order to get the best results. The control parameters shown in Table 4 are also selected based on extensive simulation results.

For a comparison, here one can see the statistical analysis. For a better analysis, different statistical parameters like the best, mean and the standard deviation are computed. Further, the average time required to run 80,000/8000 function evaluations for 14 different CEC 2005 test functions (shown in Tables 1–3) is also computed. In this paper, the term 'Best' indicates the minimum value (out of fifty independent run). Here, the term 'Mean' denotes the average value, the term 'Std.' infers the standard deviation. Further, the word 'Atime' signifies the average time to evaluate either 80,000 (for f_1–f_{12})/8000 (for f_{13}–f_{14}) function evaluation out of 50 independent run. Note that, for the CEC 2005 test functions displayed in Tables 1–3, the lower values of these statistical parameters are desirable. The performance of the ACSA using the CEC 2005 unimodal benchmark test functions is presented in Table 5 and displayed in Fig. 1. The performance of the ACSA using the CEC 2005 multimodal benchmark test functions is presented in Table 6 and shown in Fig. 2. The performance of the ACSA using the CEC 2005 extended benchmark test functions is presented in Table 7. From these extensive simulation results, it is seen that the proposed ACSA has better convergence rate than the CSA. Therefore ACSA can be used for solving the filter optimization problem efficiently. The order of the FIR digital filter is usually high. This warrants us to think for an algorithm which is faster than others.

From the Tables 5–7 and Figs. 1, 2, it is observed that ACSA has better convergence than the CSA. Hence, it is claimed that the ACSA is useful for solving filter design problems professionally with a quicker speed.

It is observed that the performance of ACSA is much better than other three algorithms with respect to speed. This has encouraged the authors to investigate its use in the digital FIR filter design. In the following section, the filter design results are presented using the ACSA.

5. Results and discussions

The Performance of the scheme depends upon the tuning of control parameters. There is no specific procedure available for perfect tuning of parameters. Researchers execute a large number of simulations by varying the value of the parameter within a particular range to get the best control parameter. When a control parameter is properly tuned, performance of the algorithm is maximized, efficiency is enhanced and accuracy is optimized. In this paper, the selected control parameters used for optimization algorithms for the design of FIR filters are given in Table 8. V_1 is the minimum particle velocity and V_2 is the maximum particle velocity.

The Flowchart for the design of FIR filter using ACSA is shown in Fig. 3.

Table 5
Performance of the CEC 2005 test functions discussed in Table 1.

Function	Algorithm	Best	Mean	Std.	Atime
F_1	ACSA	3.0370E−23	3.8093E−21	8.7317E−21	4.6486E+00
	CSA	2.3299E−11	5.1380E−10	8.2079E−10	5.8815E+00
F_2	ACSA	1.9097E−13	1.0426E−11	1.6660E−11	4.8729E+00
	CSA	1.4653E−05	4.8509E−05	2.4076E−05	6.1455E+00
F_3	ACS	2.1647E+00	7.3498E+00	3.9094E+00	2.0559E+01
	CS	5.0589E+00	1.4478E+01	7.4441E+00	2.2052E+01
F_4	ACSA	2.4995E−01	1.6332E+00	1.3854E+00	7.9577E+00
	CSA	1.6294E+00	4.6322E+00	2.2598E+00	9.9586E+00
F_5	ACSA	8.4449E+00	2.2977E+01	1.6785E+01	5.6737E+00
	CSA	1.3383E+01	2.7541E+01	1.7172E+01	8.0461E+00

Table 6
Performance of CEC 2005 multimodal test functions presented in Table 2.

Function	Algorithm	Best	Mean	Std.	Atime
F_6	ACSA	5.2250E−03	1.6055E−02	5.4150E−03	5.6440E+00
	CSA	1.5105E−02	3.3915E−02	2.1755E−02	6.8046E+00
F_7	ACSA	−9.6235E+03	−8.2098E+03	5.6733E+02	5.4038E+00
	CSA	−9.4003E+03	−8.9277E+03	2.8188E+02	6.5412E+00
F_8	ACSA	3.7818E+01	6.0532E+01	1.8498E+01	5.2736E+00
	CSA	3.1763E+01	5.0220E+01	9.9724E+00	6.4269E+00
F_9	ACSA	2.5672E−11	8.7137E−08	2.4694E−07	5.1234E+00
	CSA	1.0482E−04	5.3191E−01	6.9265E−01	6.3730E+00
F_{10}	ACSA	0.0000E+00	4.6842E−04	1.8050E−03	5.7260E+00
	CSA	4.6405E−10	2.5976E−04	1.2350E−03	6.9299E+00
F_{11}	ACSA	3.0022E−20	6.5550E−03	3.6005E−02	1.6963E+01
	CSA	8.5587E−08	2.6638E−01	3.0923E−01	1.9023E+01
F_{12}	ACSA	1.1211E−21	3.4794E−04	1.9000E−03	1.6086E+01
	CSA	2.7364E−09	4.9932E−01	2.0335E+00	1.8988E+01

Table 7
Performance evaluation of CEC 2005 extended benchmark test functions displayed in Table 3.

Function	Algorithm	Best	Mean	Std.	Atime
F_{13}	ACS	9.4810E−01	9.4810E−01	1.2179E−16	1.6694E+00
	CS	9.4810E−01	9.4810E−01	7.7462E−08	1.7728E+00
F_{14}	ACS	4.2537E−04	6.4923E−04	1.0509E−04	5.1025E−01
	CS	5.7781E−04	6.8379E−04	7.1882E−05	5.8777E−01

Table 8
Parameters used.

Parameters	Symbol	RCGA	PSO	ABC	DE	CSA	ACSA
Population size	N	100	100	100	100	100	100
Maximum iteration cycle	M	1000	1000	1000	1000	1000	1000
Tolerance	T	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
Limits of filter coefficients		−1, 1	−1, 1			−1, 1	−1, 1
Probability of discovering cuckoo eggs	P_a	–	–			0.25	0.25
Order of FIR filter	N_o	–	20			20	20
Constant parameter	L_1, L_2, V_1, V_2 α, β, γ	–	$L_1 = L_2 = 2.2$, $V_1 = 0.01, V_2 = 1$			$\alpha = 1$, $\beta = 10$, $\gamma = 1.5$	–

The performance of the non-recursive Finite Impulse Response filters optimizing by the RCGA, PSO, ABC, DE, CSA and ACSA is estimated by means of filter coefficients, pass band ripple, minimum stop-band attenuation, maximum pass-band attenuation, stop-band attenuation, normalized pass-band ripple, accuracy, computational complexity, convergence rate and efficiency. We treat the set of solutions as optimal solutions having minimum fitness value. The controlling parameters are mentioned in Table 8. The design specifications of filters are kept same for both the design problems.

5.1. Non-recursive Finite Impulse Response high pass filter

This section discusses the design of non-recursive FIR HP filter. The design specifications of high pass FIR filter are as follows — the order of HPF is 20 and its cut-off frequency (w_c) is 0.43π. The error function $E(\omega)$, described in Eq. (10), is the fitness function of optimization algorithms. The filter coefficient varies between −1 and 1. The filter coefficients are optimized by employing four optimization algorithms (RCGA, PSO, CSA, ACSA) and the corresponding values are presented in Table 9.

Fig. 4 represents the graphical comparison of magnitude response in dB. The response clearly indicates that ACSA has better stop band attenuation capability as compared to other optimization techniques (RCGA, PSO, CSA). The performances of ACSA, CSA, RCGA and PSO can be visualized accurately by enlarging the stop band response, which is shown in Fig. 5.

Table 9 represents a qualitative analysis of the FIR high pass filter. From Table 9, we clearly observe that the minimum stop-band attenuation of ACSA is −32.4003, whereas that of CSA is −27.9470. The

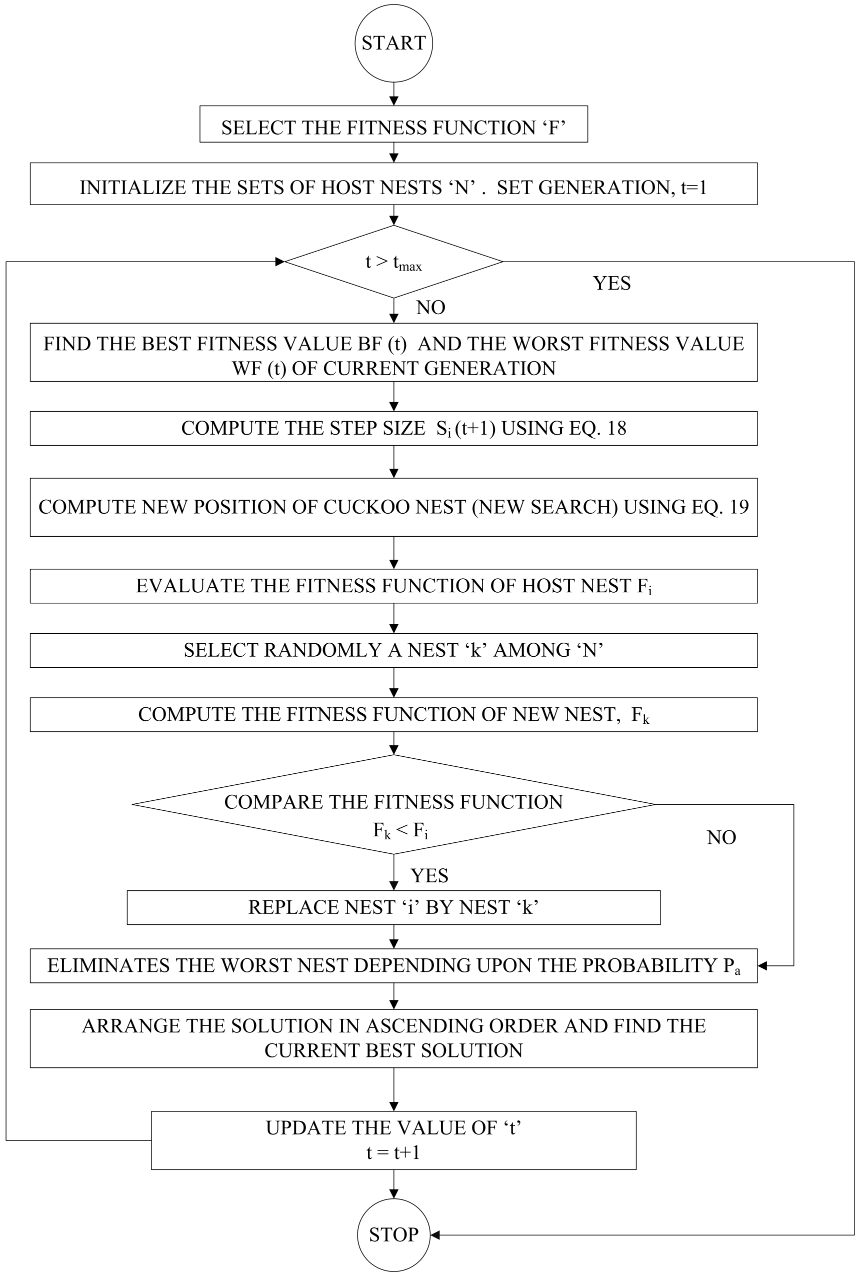

Fig. 3. Flowchart for the design of FIR filter using ACSA.

Table 9
Qualitative analysis for Finite Impulse Response high pass filter of order 20.

Algorithm	Minimum stop-band attenuation (dB)	Maximum pass-band Attenuation (dB)	Algorithm execution time (second) per 100 cycles
RCGA	−18.1732	0.9402	3.7246
PSO	−22.0897	0.9183	2.6175
ABC	−24.2986	0.8364	1.7243
DE	−26.7285	0.6924	1.1273
CSA	−27.9470	0.4211	0.8215
ACSA	−32.4003	0.2749	0.6358

maximum pass band attenuation of ACSA (0.2749) is the smallest among the four optimization techniques. The execution time of ACSA is smaller than CSA as the search space of ACSA become independent upon Levy distribution and there is no need for defining initial parameters. As the ACSA requires the smallest execution time, it is faster than other three optimization techniques.

Table 10 represents a statistical analysis of stop-band attenuation of the FIR HP filter of order 20. Among the four optimization techniques,

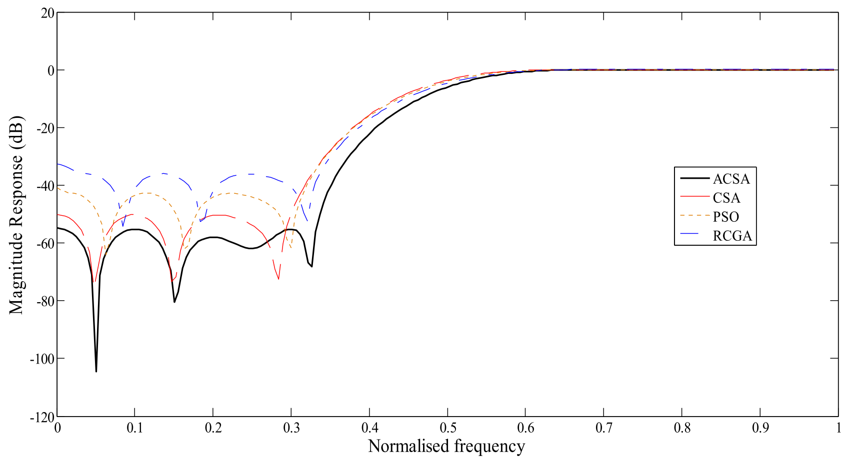

Fig. 4. Magnitude response (dB) for 20th order FIR high pass filter.

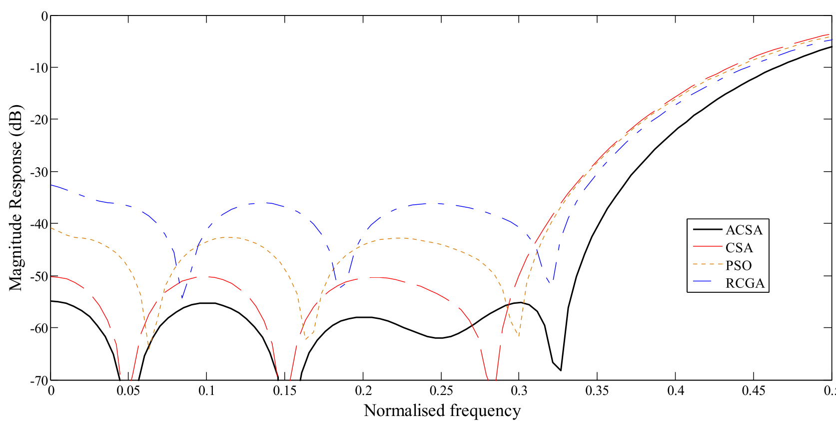

Fig. 5. Enlarged stop-band response (dB) for 20th order FIR high pass filter.

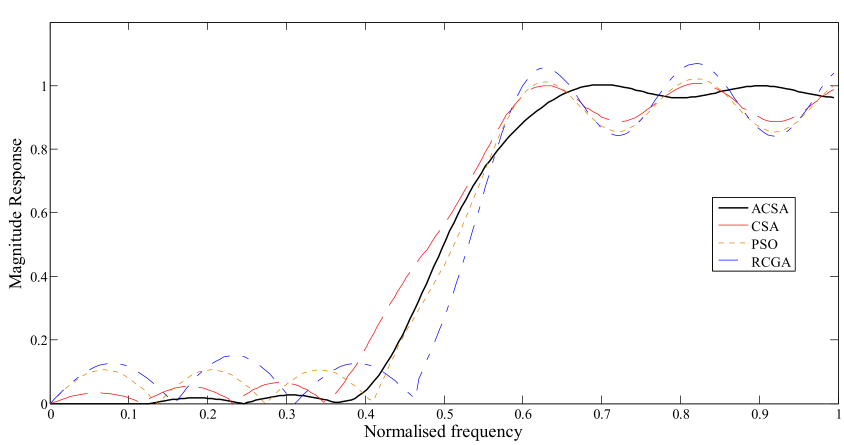

Fig. 6. Normalized magnitude response of 20th order FIR high pass filter.

ACSA has the best mean value and the minimum variance (−62.9236). Fig. 6 displays the normalized magnitude response of FIR filters optimized by using RCGA, PSO, CSA and ACSA. ACSA provides smallest variance and least overshoot. Hence, the frequency response obtained by utilizing ACSA is closer to the desired response.

Fig. 7 shows the plot for the enlarged portion of the normalized magnitude response of the pass-band of FIR HP of order 20. The maximum normalized pass-band ripple of the ACSA is 1.0030, which represents the least overshoot among the four optimization techniques. Table 11 summarizes about the comparative analysis of the performance parameters by means of maximum, mean, variance and standard deviation of the pass-band ripple (normalized).

Table 12 indicates that the ACSA has minimum normalized stop-band ripple (maximum stop-band ripple (normalized) of 0.0325,

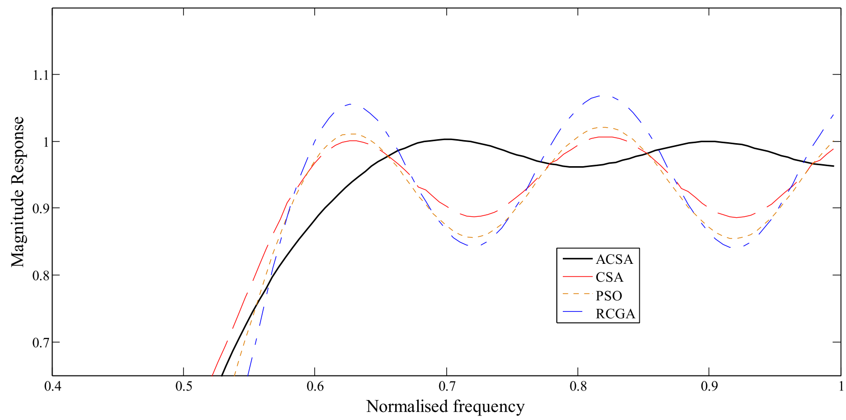

Fig. 7. Expanded pass-band response of 20th order FIR high pass filter.

Table 10
Statistical analysis of stop band attenuation for Finite Impulse Response high pass filter.

Algorithm	Stop-band attenuation (dB)		
	Mean	Variance	Standard deviation
RCGA	−23.8692	−54.8736	−27.4368
PSO	−24.2713	−55.9358	−27.9679
ABC	−25.3421	−55.9864	−27.9824
DE	−26.8127	−55.9878	−27.9892
CSA	−28.1072	−56.4144	−28.2072
ACSA	−32.4971	−62.9236	−31.4618

Table 11
Qualitative analysis of pass-band ripple for Finite Impulse Response high pass filter of order 20.

Algorithm	Normalized pass-band ripple			
	Maximum	Mean	Variance	Standard deviation
RCGA	1.0680	0.9987	0.0083	0.0911
PSO	1.0210	0.9732	0.0079	0.0889
ABC	1.0124	0.9748	0.0064	0.0776
DE	1.0101	0.9702	0.0060	0.0762
CSA	1.0055	0.9691	0.0050	0.0707
ACSA	1.0030	0.9617	0.0029	0.0539

average stop-band ripple (normalized) of 0.0237). Which clearly signifies that the performance of ACSA is superior. ACSA has a slightly larger transition width as compared to CSA, PSO, and RCGA.

5.2. Non-recursive Finite Impulse Response band stop filter

The order of the non-recursive band-stop filter is 20 and its cut-off frequencies, w_{c1} is 0.38π and w_{c2} is 0.73π . The error function $E(\omega)$, described in Eq. (11), is acting as a fitness function of optimization algorithms. The lower limit of the filter coefficient is -1 and its upper limit is $+1$. The filter coefficients are optimized by employing six optimization algorithms.

Fig. 8 shows the evaluation of magnitude response (dB). The response clearly indicates that ACSA has better stop band attenuation capability as compared to CSA. The performance of CSA and ACSA can be visualized accurately by enlarging the stop band response, which is shown in Fig. 9.

The qualitative analysis of FIR band-stop filter are shown in Table 13. Which displays the performance comparison between the ACSA and CSA. The minimum stop-band attenuation of ACSA is better than CSA. The maximum pass band attenuation of ACSA (0.1607) is smaller than CSA (0.3242). The execution time of ACSA is smaller than CSA.

Table 14 presents a statistical analysis of stop-band attenuation of the FIR BS filter of order 20 and gives a comparative analysis of performance by means of statistical parameters. Among the four optimization techniques, the ACSA has minimum variance (−56.3108) and minimum standard deviation (−28.1554). The mean value of ACSA is −28.0481. Fig. 10 displays the normalized magnitude response of FIR filters optimized by using RCGA, PSO, CSA and ACSA. The ACSA provides the smallest variance and the least overshoot. Hence, the frequency response of the ACSA is closer to the desired response.

Table 12
Quantitative analysis of stop-band ripple for Finite Impulse Response high pass filter of order 20.

Algorithm	Stop-band ripple		Transition Width
	Maximum (Normalized)	Average (Normalized)	
RCGA	0.1510	0.0640	0.1105
PSO	0.1152	0.0611	0.1579
ABC	0.1002	0.0568	0.1724
DE	0.0983	0.0500	0.1882
CSA	0.0635	0.0393	0.2157
ACSA	0.0325	0.0237	0.2281

Table 13
Qualitative analysis for Finite Impulse Response band stop filter of order 20.

Algorithm	Minimum stop-band attenuation (dB)	Maximum pass-band Attenuation (dB)	Algorithm execution time (second) per 100 cycles
RCGA	−18.3564	0.8961	3.5173
PSO	−18.1738	0.8587	2.6419
ABC	−20.2543	0.6455	1.8644
DE	−22.3678	0.5378	1.2653
CSA	−27.8359	0.3242	0.5037
ACSA	−29.8517	0.1607	0.4339

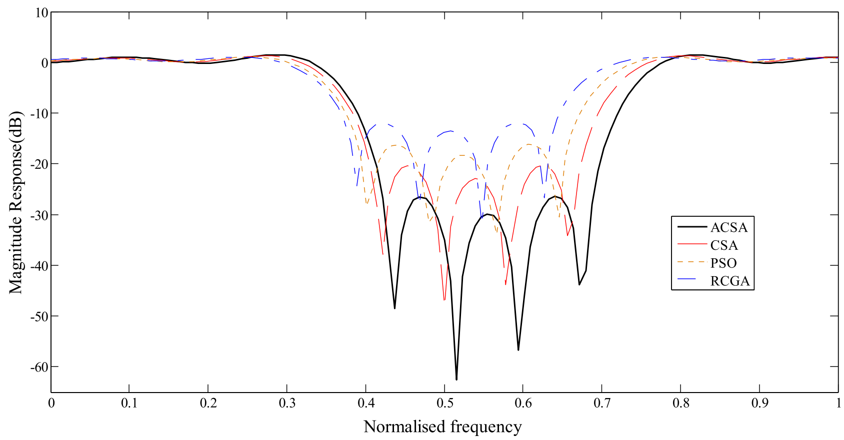

Fig. 8. Magnitude response (dB) of 20th order FIR band stop filter.

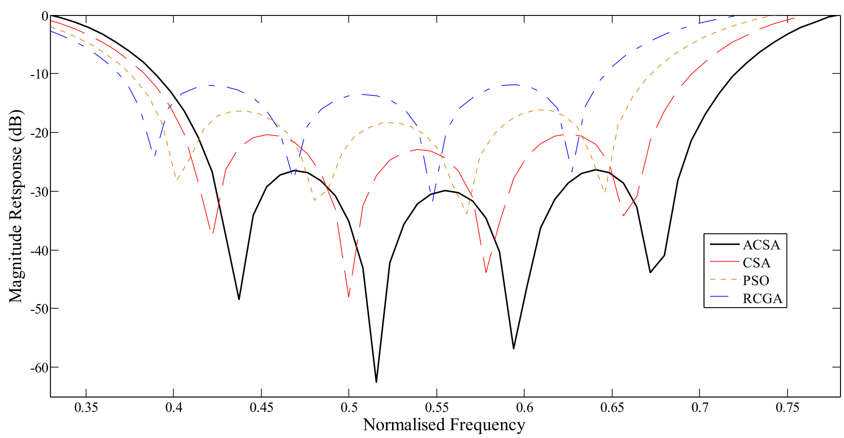

Fig. 9. Expanded stop-band response (dB) of 20th order FIR band stop filter.

Table 14
Statistical analysis of stop band attenuation for Finite Impulse Response band stop filter of order 20.

Algorithm	Stop-band attenuation (dB)		
	Mean	Variance	Standard deviation
RCGA	−19.7583	−50.9782	−25.4891
PSO	−20.2179	−51.1738	−25.5869
ABC	−22.2763	−51.6882	−25.6224
DE	−24.3317	−51.8246	−25.6648
CSA	−26.6836	−52.9104	−26.4552
ACSA	−28.0481	−56.3108	−28.1554

Table 15 displays a comparative analysis of the performance by comparing the value of maximum (Max.), mean (Average), variance (Var.) and standard deviation (Std. dev.) of the pass-band ripple (normalized). Fig. 10 displays the plot for the enlarged portion of the normalized magnitude response of the pass-band of the FIR BS filter of order 20. The maximum normalized pass-band ripple of ACSA is 1.0035, which represents the least overshoot among the four optimization techniques. The result indicates that there is a significant improvement of maximum overshoot, mean, variance and standard deviation while comparing the ACSA with the CSA.

Table 16 displays a comparative analysis of the performance by comparing the values of maximum, mean, variance and standard deviation of the pass-band ripple (normalized). One can clearly observe that the ACSA has a minimum standard deviation and minimum variance, which signifies the superiority of ACSA. From Table 17, we can conclude that the maximum stop-band ripple (normalized) and the average stop-band ripple (normalized) of the ACSA is smallest among the four optimization techniques. Stop-band width of ACSA is also the smallest, which indicates excellent performance of ACSA.

Table 15
Qualitative analysis of pass-band ripple for Finite Impulse Response band stop filter of order 20.

Algorithm	Normalized passband ripple			
	Max.	Average	Var.	Std. dev.
RCGA	1.1390	0.9897	0.0059	0.0768
PSO	1.1160	0.9872	0.0054	0.0735
ABC	1.1088	0.9863	0.0048	0.0724
DE	1.1004	0.9854	0.0042	0.0712
CSA	1.0385	0.9847	0.0036	0.0600
ACSA	1.0035	0.9832	0.0021	0.0458

Table 16
Quantitative analysis of pass-band ripple for Finite Impulse Response band stop filter of order 20.

Algorithm	Normalized pass-band ripple			
	Maximum	Mean	Variance	Standard deviation
RCGA	1.1390	0.9897	0.0059	0.0768
PSO	1.1160	0.9872	0.0054	0.0735
ABC	1.1025	0.9866	0.0044	0.0721
DE	1.1001	0.9858	0.0041	0.0702
CSA	1.0385	0.9847	0.0036	0.0600
ACSA	1.0035	0.9832	0.0021	0.0458

Fig. 11 displays the plot for the enlarged portion of the normalized magnitude response of the pass-band of the FIR BS filter of order 20. The maximum normalized pass-band ripple of ACSA is 1.0035, which

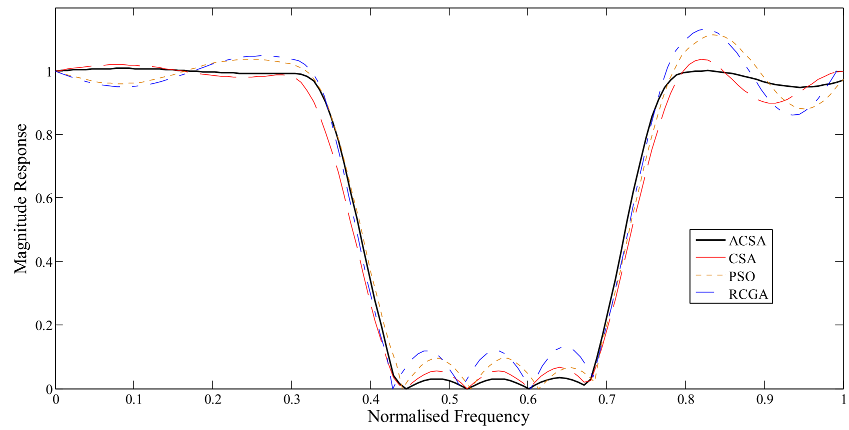

Fig. 10. Normalized magnitude response of 20th order FIR band stop filter.

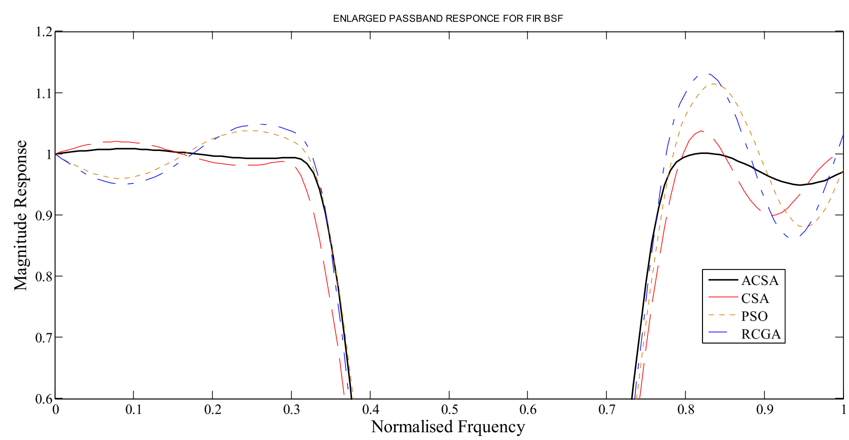

Fig. 11. Enlarged pass-band response for 20th order FIR band stop filter.

Table 17
Quantitative analysis of stop-band ripple for Finite Impulse Response band stop filter of order 20.

Algorithm	Stop-band ripple		Stop-band width
	Maximum (Normalized)	Average (Normalized)	
RCGA	0.1307	0.1028	0.2539
PSO	0.1092	0.0975	0.2441
ABC	0.1026	0.0936	0.2422
DE	0.1022	0.0872	0.2400
CSA	0.0662	0.0463	0.2310
ACSA	0.0330	0.0396	0.2266

represents the least overshoot among the four optimization techniques. The result indicates that there is a significant improvement of maximum overshoot, mean, variance and standard deviation while comparing the ACSA with the CSA.

5.3. *t*-Test

It is interesting to note that *t*-Test determines the means of two groups, which are statistically different from each other or not. We can calculate the results of *t*-Test by using Eq. (20).

$$t = \frac{m_o - m_a}{\sqrt{\left(\sigma_o^2/n_o\right) + \left(\sigma_a^2/n_a\right)}}. \tag{20}$$

Standard error of difference (St_{error}) is represented as

$$St_{error} = S_p\left(\frac{1}{n_o} + \frac{1}{n_a}\right)^{0.5} \tag{21}$$

where S_p is pooled standard deviation.

$$S_p = \left(\frac{\left(n_o - 1\right)\sigma_o^2 + \left(n_a - 1\right)\sigma_a^2}{n_o + n_a - 2}\right)^{0.5}$$

where, m_a is the mean value of the ACSA and m_o is the mean value of other optimization techniques (CSA, PSO or RCGA) . Here, σ_a and n_a are the standard deviation of ACSA and the number of samples computed in ACSA, respectively. Note that σ_o represents the standard deviation of other optimization techniques (CSA, PSO or RCGA), whereas n_o is the number of samples computed in the optimization technique.

Here we perform *t*-Test by individually comparing CSA, DE, ABC, PSO or RCGA with ACSA. The high positive *t* value indicates the superiority of the ACSA over other optimization techniques (CSA, DE, ABC, PSO or RCGA). Table 18 clearly indicates the superiority of the ACSA over CSA, DE, ABC, PSO, RCGA for the design of FIR high-pass filter as all the *t* values of pass-band ripple and stop-band ripple for

Table 18
t-Test result for Finite Impulse Response high pass filter of order 20.

Algorithm	Pass-band ripple		Stop-band ripple	
	t	Standard error of difference	t	Standard error of difference
RCGA	1.9438	0.019	4.2416	0.010
PSO	0.6150	0.019	4.2236	0.009
ABC	0.5667	0.019	4.0011	0.009
DE	0.5002	0.019	3.8676	0.009
CSA	0.4162	0.018	1.9100	0.008

Table 19
t-Test result for Finite Impulse Response band stop filter of order 20.

Algorithm	Pass-band ripple		Stop-band ripple	
	t	Standard error of difference	t	Standard error of difference
RCGA	0.4048	0.016	5.0224	0.013
PSO	0.2581	0.015	4.8293	0.012
ABC	0.2011	0.015	4.4332	0.012
DE	0.1884	0.015	3.3883	0.012
CSA	0.0994	0.015	0.5943	0.011

CSA, DE, ABC, PSO and RCGA are positive t values. Standard error of difference and t value of the CSA is smaller than DE, ABC, PSO and RCGA indicates CSA is better than DE, ABC, PSO and RCGA. From Table 19, we confirm that ACSA has better performance in the design of FIR band-stop filter as compared to CSA, DE, ABC, PSO and RCGA. The t value and the standard error of difference of CSA is minimum, which signifies its superiority over DE, ABC, PSO and RCGA.

6. Conclusion

In this paper, a comparative and analytical study of FIR filter design using six popular optimization techniques (RCGA, PSO, ABC, DE, CSA and ACSA) is discussed. The performance of FIR HP and BS filters optimized by above four optimization techniques is studied. The aim of the filter design is to find the optimal filter coefficients having minimum relative error with respect to the ideal filter response. The simulation result clearly signifies that ACSA performs better by means of magnitude response with high stop-band attenuation, optimum pass-band and stop-band ripples; and smallest execution time.

The magnitude response clearly indicates that ACSA has better stop band attenuation capability as compared to other three optimization techniques. The search space of ACSA becomes independent upon Levy distribution. Further, there is no need to define initial parameters. Hence, ACSA executes faster than CSA. From the t-Test analysis, it is observed that ACSA may be treated as an efficient tool to be used for optimal filter design. Finally, it is concluded that ACSA is more efficient, accurate, faster and a better global optimizer than RCGA, PSO, ABC, DE and CSA.

References

Ababneh, J.I., Bataineh, M.H., 2008. Linear phase FIR filter design using particle swarm optimization and genetic algorithms. Digit Signal Process. 18 (4), 657–668.
Aggarwal, A., Rawat, T.K., Upadhyay, D.K., 2016. Design of optimal digital FIR filter using evolutionary and swarm optimization techniques. AEU 70, 373–385.
Ahmad, S.U., Antoniou, A., 2006. A genetic algorithm approach for fractional delay FIR filters. In: IEEE International Symposium on Circuits and Systems, ISCAS 2006, pp. 2517–2520.
Antoniou, A., 2005. Digital Signal Processing: Signal Systems and Filters. McGraw Hill, NY, USA.
Barthelemy, P., Bertolotti, J., Wiersma, D.S., 2008. A Levy flight for light. Nature 453, 495–498.
Bhandari, A.K., Sing, V.K., Kumar, A., Sing, G.K., 2014. Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy. Expert Syst. Appl. 41, 3538–3560.
Boudejelaba, K., Ros, F., Chikouche, D., 2014. Potential of particle swarm optimization and genetic algorithms for FIR filter design. Circuits Syst. Signal Process. 33 (10), 3195–3222.
Chakraverty, S., Kumar, A., 2011. Design optimization for reliable embedded system using Cuckoo search. In: International Conference on Electronics Computer Technology, Vol. 1, pp. 264–268.
Dan, J., 2016. The application of artificial bee colony (ABC) algorithm in FIR filter design. In: 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery. ICNC-FSKD, IEEE, pp. 663–667.
Das, S., Konar, A., 2006. Two dimensional IIR filter design with modern search heuristics: a comparative study. Int. J. Comput. Intell. Appl. 6 (3), 329–355.
Dhabal, S., Venkateswaran, P., 2017. A novel accelerated artificial bee colony algorithm for optimal design of two dimensional FIR filter. Multidimens. Syst. Signal Process. 28 (2), 471–493.
Gandomi, A.H., Yang, X.S., Alavi, A.H., 2013. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng. Comput. 29 (1), 17–35.
Gandomi, A.H., Yang, X.S., Talatahari, S., Deb, S., 2012. Coupled eagle strategy and differential evolution for unconstrained and constrained global optimization. Comput. Math. Appl. 63 (1), 191–200.
García, S., Molina, D., Lozano, M., Herrera, F., 2009. A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC'2005 special session on real parameter optimization. J. Heuristics 15 (6), 617–644.
Kar, R., Mandal, D., Mandal, S., Ghosal, S.P., 2012. Craziness based particle swarm optimization algorithm for FIR band stop filter design. Swarm Evol. Comput. 58–64.
Karaboga, D., Basturk, B., 2007. A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm. J. Global Optim. 39 (3), 459–471.
Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Network, Vol. 4, pp. 1942–1948.
Koziel, S., Yang, X.S., 2011. Coputational Optimization, Methods and Algorithms, Vol. 356. Springer, Germany.
Lian, Yong, Wei, Ying, 2005. A computationally efficient nonuniform FIR digital filter bank for hearing aids. IEEE Trans. Circuits Syst. I: Regul. Pap. 52, 2754–2762.
Lita, L., Visan, D.A., Cioc, I.B., Banica, L., 2004. Low voltage FIR filter for receivers used in communication systems. In: Proceedings of the 27th International Spring Seminar on Electronics Technology: Meeting the Challenges of Electronics Technology Progress, 2001. IEEE, pp. 434–438.
Litwin, L., 2000. FIR and IIR digital filters. IEEE Potentials 0278-6648, 28–31.
Liu, G., Li, Y.X., He, G., 2010. Design of digital FIR filters using differential evolution algorithm based on reserved gene. In. IEEE Congress on Evolutionary Computation, pp. 1–7.
Luitel, B., Venayagamoothry, G.K., 2010. Particle swarm optimization with quantum infusion for system identification. Eng. Appl. Artif. Intell. 23 (5), 635–649.
Mandal, S., Ghosal, S.P., Kar, R., Mandal, D., 2012. Design of optimal linear phase craziness based FIR high pass filter using craziness based particle swarm optimization. J. King South Univ. 24, 83–92.
Miller, T.R., Hagge, R.G., Wallis, J.W., Sampathkumaran, K.S., 1988. Interactive digital filtering of gated cardiac studies during cinedisplay. IEEE Trans. Med. Imaging 7 (3), 188–192.
Naik, M., Nath, M.R., Wunnava, A., Sahany, S., 2015. A new adaptive Cuckoo search algorithm. In: IEEE 2nd International Conference on Recent Trend in Information Systems.
Naik, M.K., Panda, R., 2016. A novel adaptive Cuckoo search algorithm for intrinsic discriminant analysis based phase reorganisation. Appl. Soft Comput. 661–675.
Panda, R., Agrawal, S., Bhuyan, S., 2013a. Edge magnitude based multilevel thresholding using Cuckoo search technique. Expert Syst. Appl. 40, 7617–7628.
Panda, R., Naik, M.K., Mishra, N., 2013b. Design of two-dimensional recursive filters using bacteria foraging optimization. In: Proceedings of IEEE International Conference on Computational Intelligence. IEEE Computational Intelligence Society, Singapore, pp. 188–193.
Panda, G., Pradhan, P.M., Majhi, B., 2011. IIR system identification using cat swarm optimization. Expert Syst. Appl. 38 (10), 12671–12683.
Parks, T.W., McCllean, J.H., 1972. Chebyshev approximation for non recursive digital filters with linear phase. IEEE Trans. Circuit Theory CT-19, 189–194.
Rabiner, L.R., 1973. Approximate design relationships for low-pass FIR digital filters. IEEE Trans. Audio Electroacoust. AU-21, 456–460.
Raju, R., Kwan, H.K., 2017. FIR filter design using Multiobjective Artificial Bee Colony algorithm. In: 2017 IEEE 30th Canadian Conference Electrical and Computer Engineering, CCECE, pp. 1-4.
Rout, T.K., 2014. Digital Signal Processing, first ed. Oxford University Press.
Sarangi, S.K., Panda, R., Dash, M., 2014. Design of 1-D and 2-D recursive filters using crossover bacterial foraging and Cuckoo search techniques. Eng. Appl. Artif. Intell. 34, 109–121.
Shi-Wei, W., 2011. A new denoting method in seismic data processing based on Wiener filter. In: Electron Signal Process. In: LNEE, vol. 97, pp. 191–194.
Singh, K., Kaur, G., 2014. Design of low pass FIR filter using Artificial Bee Colony optimization technique and its comparison with particle swarm optimization. Int. J. Adv. Res. Comput. Sci. Soft. Eng. 4 (9), 379–387.
Storn, R., Price, K., 1997. Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces. J. Global Optim. 11, 341–359.

Tak-Shing, W., Bouman, C.A., Pollak, i., 2012. Image enhancement using hypothesis selection filter: theory and application of JPEG coding. IEEE Trans. Image Process. 22 (3), 898–913.

Vaidyanathan, P.P., 2006. Filter banks in digital communications. IEEE Circuits Syst. Mag. 1 (2), 4–25.

Wozniak, M., Poiap, D., Napoli, C., Tramontana, E., 2016. Graphic object feature extraction system based on Cuckoo search algorithm. Expert Syst. Appl. 66, 20–31.

Yang, X.S., 2010. Engineering Optimization: An Introduction with Metaheuristic Applications. Wiley, New York.

Yang, X.S., Deb, S., (2009) Cuckoo search via Levy flights. In: Proceeding of the World Congress on Nature & Biologically Inspired Computing, pp. 210–214.

Yang, X.S., Deb, S., 2010. Engineering optimization by Cuckoo search. Int. J. Math. Model. Number. Optim. 1 (4), 330–343.

Yang, X.S., Deb, S., 2013. Cuckoo search: recent advances and applications. Neural Comput. Appl. 24 (1), 169–174.

Yang, X.S., Deb, S., Fong, S., 2011. Accelerated particle swarm optimization and support vector machine for business optimization and applications. In: Networked Digital Technologies. In: Common Comput. Inf. Sci., vol. 136, pp. 53–66.

Yu, Y.J., Lim, Y.C., (2002) Genetic algorithm approach for the optimization of multiplierless sub-filters generated by the frequency response masking technique. In: Proceedings of the IEEE International Conference on Electronics, Circuits, and Systems, Vol. 3, pp. 1163–1166.

Zahorian, A.S., Gordy, P., 1983. Finite impulse response (FIR) filters for speech analysis and synthesis. IEEE ICASSP Acount Speech Signal Process. 8, 808–811.

Zhang, Y., Chen, C.-H., He, T., Temes, G.C., 2015. A continuous-time delta-sigma modulator for biomedical ultrasound beamformer using digital ELD compensation and FIR feedback. IEEE Trans. Circuits Syst. I: Regul. Pap. 62, 1689–1698.

Zheng, H.Q., Zhou, Y., 2012. A novel Cuckoo search algorithm based on Gauss distribution. J. Comput. Inf. Syst. 8, 4193–4200.