Journal of King Saud University – Computer and Information Sciences (2015) 27, 25–39

King Saud University

Journal of King Saud University – Computer and Information Sciences

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

Optimal IIR filter design using Gravitational Search Algorithm with Wavelet Mutation

S.K. Saha [a], R. Kar [a,*], D. Mandal [a], S.P. Ghoshal [b]

[a] *Department of ECE, NIT Durgapur, India*
[b] *Department of EE, NIT Durgapur, India*

Received 6 January 2013; revised 22 October 2013; accepted 13 March 2014
Available online 9 May 2014

KEYWORDS

IIR filter;
GSAWM;
Evolutionary optimization techniques;
Magnitude response;
Pole-zero plot;
Stability

Abstract This paper presents a global heuristic search optimization technique, which is a hybridized version of the Gravitational Search Algorithm (GSA) and Wavelet Mutation (WM) strategy. Thus, the Gravitational Search Algorithm with Wavelet Mutation (GSAWM) was adopted for the design of an 8th-order infinite impulse response (IIR) filter. GSA is based on the interaction of masses situated in a small isolated world guided by the approximation of Newtonian's laws of gravity and motion. Each mass is represented by four parameters, namely, position, active, passive and inertia mass. The position of the heaviest mass gives the near optimal solution. For better exploitation in multidimensional search spaces, the WM strategy is applied to randomly selected particles that enhance the capability of GSA for finding better near optimal solutions. An extensive simulation study of low-pass (LP), high-pass (HP), band-pass (BP) and band-stop (BS) IIR filters unleashes the potential of GSAWM in achieving better cut-off frequency sharpness, smaller pass band and stop band ripples, smaller transition width and higher stop band attenuation with assured stability.

1. Introduction

Digital Signal Processing (DSP) provides greater flexibility, better time and environment stability, higher performance and lower equipment production costs traditional analog techniques. Many microprocessor circuits are being replaced with cost-effective DSP techniques and products. In the creation of digital filters, DSP chips play an important role. A digital filter is simply a discrete time and discrete amplitude convolver. In the z-domain, the linear convolution of the input and filter sequence in the time domain is equivalent to the multiplication of the corresponding z-transforms. The filtered output sequence is obtained by product inverse z-transform. Digital filters are broadly classified into two main categories: finite impulse response (FIR) filters and infinite impulse response (IIR) filters Oppenheim et al., 1999; Proakis and Manolakis, 1996. The output of FIR filters depends on the present and past input values of input, so the name 'non-recursive' is aptly suited for these filter types. However, the output of IIR filters depends not only on

[*] Corresponding author. Tel.: +91 9434788056; fax: +91 343 2547375.
E-mail addresses: rajibkarece@gmail.com (R. Kar), durbadal.bittu@gmail.com (D. Mandal), spghoshalnitdgp@gmail.com (S.P. Ghoshal).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

1319-1578 © 2014 King Saud University. Production and hosting by Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.jksuci.2014.03.002

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 type. A large amount of memory is required to store the previous outputs for recursive IIR filters. Thus, because of these aspects, the creation of a FIR filter is easier given the lower memory space requirements and design complexity. Ensured stability and the linear phase response over a wide frequency range are additional advantages. However, IIR filters distinctly meet the desired specifications of sharp transition width, lower pass band ripple and higher stop band attenuation with lower order compared with FIR filters. As a consequence, a properly designed IIR filter can meet close to the ideal magnitude response more finely than an FIR filter, and IIR filters always ensure stability. Because of these challenging features and a wide range of potential applications, the performances of IIR filters designed with various evolutionary optimization algorithms were compared to determine the comparative effectiveness of the algorithms and the best design of optimal IIR filters.

In the conventional approach, IIR filters of various types (Butterworth, Chebyshev and Elliptic, etc.) can be implemented using two methods. In the first case, a frequency sampling technique is adopted for Least Square Error (Lang, 2000) and Remez Exchange (Jackson and Lemay, 1990) process. In the second method, filter coefficients and minimum order are calculated for a prototype low pass filter in the analog domain, which is then transformed to a digital domain via bilinear transformation. This frequency mapping works well at low frequency, but in high frequency domains, this method is liable to result in frequency warping (Hussain et al., 2011).

IIR filter design is a challenging optimization problem. Thus far, gradient-based classical algorithms, such as steepest descent and quasi-Newton algorithms have been used aptly for the design of IIR filters (Antoniou, 2005; Lu and Antoniou, 2000). In general, these algorithms are very fast and efficient in obtaining 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 an IIR filter is multimodal, and thus, superior evolutionary optimization techniques are required to determine better global solution.

The shortfalls of classical optimization techniques in handling any multimodal optimization problem are as follows: (i) requires a continuous and differentiable error fitness function (cost or objective function), (ii) usually converges to the local optimum solution or revisits the same sub-optimal solution, (iii) incapable of searching a large problem space, (iv) requires a piecewise linear cost approximation (linear programming) and (v) highly sensitive to the starting points when the number of solution variables is increased, and as a result, the solution space is also increased.

Thus, it can be concluded that classical search techniques are only suitable for handling differentiable unimodal objective functions with constricted search space. Accordingly, the various evolutionary heuristic search algorithms applied to filter optimization problems in recent times include the following. Genetic Algorithms (GAs) are inspired by Darwin's "Survival of the Fittest" strategy (Karaboga and Cetinkaya, 2004; Tsai et al., 2006; Yu and Xinjie, 2007). Simulated Annealing (SA) was designed based on thermodynamic effects (Chen et al., 2001); Artificial Immune Systems (AIS) mimic biological immune systems (Kalinli and Karaboga, 2005). Ant Colony Optimization (ACO) simulates ants' food searching behavior (Karaboga et al., 2004). Bee Colony Optimization mimics the honey collecting behavior of a bee swarm (Karaboga and Cetinkaya, 2011). Cat Swarm Optimization (CSO) is based upon the behavior of cats when tracking and seeking an object (Panda et al., 2011). In addition, Particle Swarm Optimization (PSO) simulates the behavior of bird flocking or fish schooling (Pan and Chang, 2011; Das and Konar, 2007; Fang et al., 2009; Gao et al., 2008; Sun et al., 2010; Chen and Luk, 2010; Luitel and Venayagamoorthy, 2008a,b; Mandal et al., 2011, 2012; Wang et al., 2011). In Quantum-behaved PSO (QPSO), the quantum behavior of particles in a potential well is applied to a conventional PSO algorithm (Fang et al., 2009; Sun et al., 2010). To eliminate premature convergence and stagnation, chaotic perturbation is applied to the particles in the Chaos PSO (CPSO) technique Gao et al., 2008. Differential Evolution PSO (DEPSO) reflects the hybridization of DE and PSO in offspring that are created by parental mutation (Luitel and Venayagamoorthy, 2008a). A Particle Swarm Optimization with a quantum infusion technique is adopted in Luitel and Venayagamoorthy (2008b). In Craziness-based PSO (CRPSO), the sudden direction changing behavior of a particle in a swarm is mimicked in the conventional velocity equation of PSO with the incorporation of a 'craziness factor' (Mandal et al., 2011, 2012).

In this paper, the comparative capability of the global search and near optimum result finding features of RGA, PSO, GSA and GSAWM are individually investigated thoroughly in the solving of 8th-order IIR filter design problems. GA is a probabilistic heuristic search optimization technique developed by Holland (1975). The features, such as multi-objectivity, coded variables and natural selection, make this technique distinct and suitable for finding the near global solution of filter coefficients.

Particle Swarm Optimization (PSO) is a swarm intelligence-based algorithm developed by Kennedy and Eberhart (1995) and Eberhart and Shi (1998). Several attempts have been made to design digital filters with basic PSO and its modified versions (Pan and Chang, 2011; Das and Konar, 2007; Fang et al., 2009; Gao et al., 2008; Sun et al., 2010; Chen and Luk, 2010; Luitel and Venayagamoorthy, 2008a,b; Mandal et al., 2011, 2012; Wang et al., 2011). The main attraction of PSO is its simplicity in computation, and a few steps are required in the implementation of the algorithm.

The limitations of the conventional PSO are premature convergence and stagnation problems (Ling et al., 2008; Biswal et al., 2009). To overcome these problems, a hybridized version of the Gravitation Search Algorithm (GSA), called the Gravitation Search Algorithm with Wavelet Mutation (GSAWM), is suggested by the authors for the design of 8th-order LP, HP, BP and BS IIR filters.

Wavelets are mathematical transient functions that are characterized by translation and dilation factors. According to the mutation strategy, every string has an unequal mutation probability. Thus, randomly selected strings, dependent on the mutation probability of mutation, and their elements undergo the mutation process. Mutation introduces variation in the string elements that aids in finding better near optimal solutions. In wavelet mutation, iteration-dependent variable mutation is addressed, i.e., in the exploration (early search stage) larger wavelet mutation function values and in the exploitation (fine tuning or the local search) stage steadily decreasing wavelet mutation function values are utilized for efficient searching in the multidimensional search space (Ling et al., 2008).

The paper is organized as follows: Section 2 describes the IIR filter design problem. Evolutionary algorithms, namely, RGA, PSO, GSA and GSAWM, as well as their comparative results, are discussed in Section 3. Section 4 discusses the simulation results obtained for the designed IIR filters employing different algorithms. Finally, Section 5 presents the paper's conclusions.

2. IIR filter design formulation

This section discusses the IIR filter design strategy. The input–output relation is governed by the following difference equation (Proakis and Manolakis, 1996):

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

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

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

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

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

$$\text{or } H(\Omega) = \frac{Y(\Omega)}{X(\Omega)} = \frac{b_0 + b_1 e^{-j\Omega} + b_2 e^{-j2\Omega} + \ldots + b_m e^{-jm\Omega}}{1 + a_1 e^{-j\Omega} + a_2 e^{-j2\Omega} + \ldots + b_n e^{-jn\Omega}} \tag{4}$$

where $\Omega = 2\pi\left(\frac{f}{f_s}\right)$ in $[0, \pi]$ 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 (Karaboga and Cetinkaya, 2004; Luitel and Venayagamoorthy, 2008a,b). The commonly used approach for IIR filter design is to represent the problem as an optimization problem with the mean square error (MSE) as the error fitness function (Karaboga and Cetinkaya, 2004; Luitel and Venayagamoorthy, 2008a,b), as expressed in (5).

$$J_1(\omega) = \frac{1}{N_s}\left[(d(p) - y(p))^2\right] \tag{5}$$

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 $J_1(\omega)$ with proper adjustment of coefficient vector ω represented as follows:

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

In this paper, a novel error fitness function given in (7) is adopted to achieve higher stop band attenuation and better control of the transition width. Using (7), the filter design approach results in considerable improvement in stop band attenuation over other optimization techniques.

$$J_2(\omega) = \Sigma_\Omega abs[abs(|H(\Omega)| - 1) - \delta_p] + \Sigma_\Omega[abs(|H(\Omega)| - \delta_s)] \tag{7}$$

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

The error fitness function given in (7) represents the generalized fitness function to be minimized when individually employing the evolutionary algorithms RGA, conventional PSO, GSA and the proposed GSAWM. Each algorithm attempts to minimize this error fitness $J_2(\omega)$ and thus optimizes the filter performance. Unlike other error fitness functions, such as given in Karaboga and Cetinkaya (2004) and Luitel and Venayagamoorthy, 2008a,b, that consider only the maximum errors, $J_2(\omega)$ involves the summation of all absolute errors for the whole frequency band, and minimization of $J_2(\omega)$ yields much higher stop band attenuation and lower pass band and stop band ripples.

3. Employed evolutionary optimization algorithms

Evolutionary optimization algorithms stand upon the platform of heuristic search methods, which are characterized by features such as being stochastic, adaptive and learning to produce intelligent optimization schemes. Such schemes have the potential to adapt to their ever-changing dynamic environment through previously acquired knowledge. A few such efficient algorithms are discussed in the context of designing IIR filters as well as comparison of performances as part of handling the optimization problem of IIR filter design.

3.1. Real coded genetic algorithm (RGA)

The Standard Genetic Algorithm (also known as real coded GA) is mainly a probabilistic search technique, based on the principles of natural selection and evolution as adapted from Darwin's "Survival of the Fittest" strategy (Holland, 1975). Each encoded chromosome that constitutes the population is a solution to the filter design optimization problem.

The steps of RGA as implemented for the optimization of the coefficient vector ω are as follows (Mondal et al., 2010, 2011, 2012):

Step 1: Initialize the real coded chromosome strings (ω) of the n_p population, each consisting of an equal number of numerator and denominator filter coefficients b_k and a_k, respectively; the total coefficients = $(n+1)*2$ for the nth order filter to be designed; the minimum and maximum values of the coefficients are -2 and $+2$, respectively; the number of samples = 128; pass band ripple $\delta_p = 0.01$, and stop band ripple $\delta_s = 0.001$.

Step 2: Decoding of the strings and evaluation of error fitness values $J_2(\omega)$ according to (7).

Step 3: Selection of elite strings to increase error fitness values from the minimum value.

Step 4: Copying of 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 the maximum number of cycles is reached. The grand minimum error fitness and its corresponding chromosome string or the desired solution having (n + 1)*2 number of filter coefficients are finally obtained.

3.2. Particle Swarm Optimization (PSO)

PSO is a flexible, robust, population-based stochastic search algorithm with the attractive features of simplicity in implementation and ability to quickly converge to a reasonably good solution. In addition, it has the capability to handle a larger search space and non-differential objective functions, unlike traditional optimization methods. Kennedy and Eberhart (1995), Eberhart and Shi (1998) developed PSO algorithms to simulate the 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 attempts to move toward the optimum solution and where next movement is influenced by the previously acquired knowledge of particle best and global best positions, once achieved, of the individuals and the entire swarm, respectively.

To some extent, IIR filter design and other designs with PSO are already reported in Pan and Chang (2011), Das and Konar (2007), Fang et al. (2009), Gao et al. (2008), Sun et al. (2010), Chen and Luk (2010), Luitel and Venayagamoorthy (2008a,b) and Mandal et al. (2011, 2012).

The basic steps of the PSO algorithm are as follows (Mandal et al., 2011, 2012):

Step 1: Initialize the real coded particles (ω) of the n_p population; each consists of an equal number of numerator and denominator filter coefficients b_k and a_k, respectively; the total coefficients D = (n + 1)*2 for nth order filter to be designed; the minimum and maximum values of the coefficients are −2 and +2, respectively; the number of samples = 128; pass band ripple $\delta_p = 0.01$, and stop band ripple $\delta_s = 0.001$, with a maximum velocity $V_{\max} = 1.0$ and minimum velocity $V_{\min} = 0.01$.
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 that is updated with each iteration.
Step 4: Each particle can also remember the best position the swarm has ever attained (*gbest*), which is known as social information and updated with each iteration.
Step 5: The velocity and position of each particle are modified according to (8) and (10), respectively Kennedy and Eberhart, 1995.

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

$$\text{where } V_i = V_{\max} for\ V_i > V_{\max} \\ = V_{\min} for\ V_i < V_{\min} \quad (9)$$

$$S_i^{(k+1)} = S_i^{(k)} + V_i^{(k+1)} \quad (10)$$

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

3.3. Gravitation Search Algorithm (GSA)

In GSA Rashedi et al., 2009, 2011; Bahrololoum et al., 2012, agents/solution vectors are considered as objects, and their performances are measured by their masses. All these objects attract each other via gravity forces, and these forces produce a global movement of all objects toward the objects with heavier masses. Thus, masses cooperate using a direct form of communication through gravitational forces. The heavier masses (which correspond to better solutions) move more slowly than lighter ones. This guarantees the exploitation step of the algorithm.

Three types of masses are defined in theoretical physics:

(a) *Active gravitational mass* (M_a) is a measure of the strength of the gravitational field due to a particular object. The gravitational field of an object with a small active gravitational mass is weaker than that of an object with a more active gravitational mass.
(b) *Passive gravitational mass* (M_p) is a measure of the strength of an object's interaction with the gravitational field. Within the same gravitational field, an object with a smaller passive gravitational mass experiences a smaller force than an object with a larger passive gravitational mass.
(c) *Inertial mass* (M_i) is a measure of an object's resistance to changing its state of motion when a force is applied. An object with a large inertial mass changes its motion more slowly, and an object with small inertial mass changes it rapidly.

In GSA, each mass (agent) has four specifications: position, inertial mass, active gravitational mass, and passive gravitational mass. The position of the mass corresponds to the solution of the problem, and its gravitational and inertial masses are determined using a fitness function. In other words, each mass presents a solution, and the algorithm is navigated by properly adjusting the gravitational and inertial masses. By lapse of iteration cycles, it is expected that masses be attracted by the heaviest mass. This heaviest mass will present an optimum solution in the search space.

The GSA could be considered as an isolated system of masses. It is similar to a small artificial world of masses obeying the Newtonian laws of gravitation and motion. More precisely, masses obey the following two laws.

i. *Law of gravitation:* each particle attracts every other particle, and the gravitational force between two particles is directly proportional to the product of their masses and inversely proportional to the square of the distance (R) between them. R is used as R^{rPower} ($rPower$ = 1) because R offered better results than R^2 in all the experimental cases with standard benchmark functions (Rashedi et al., 2009). This difference is a deviation of GSA from normal Newton's law of gravitation.

ii. *Law of motion:* the current velocity of any mass is equal to the sum of the fraction of its previous velocity and the variation in the velocity. The variation in the velocity or acceleration of any mass is equal to the force acted on the system divided by the mass of inertia.

Currently, let us consider a system with N agents (masses). The position of the i^{th} agent is defined by

$$X_i = (x_i^1, ...x_i^d, ...x_i^n) \quad \text{for} i = 1,2,...,N \tag{11}$$

where x_i^d presents the position of ith agent in the dth dimension.

At a specific iteration cycle t, the force acting on ith mass from jth mass is defined as in the following equation

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t)\times M_{aj}(t)}{(R_{ij}(t)+\varepsilon)^{rPower}}\left(X_j^d(t)-X_i^d(t)\right) \tag{12}$$

where $M_{aj}(t)$ is the active gravitational mass related to the jth agent at iteration cycle t; $M_{pi}(t)$ is the passive gravitational mass related to the j^{th} agent at iteration cycle t; $G(t)$ is gravitational constant at iteration cycle t; ε is a small constant, and $R_{ij}\ (t)$ is the Euclidian distance between the two agents i and j given by (13).

$$R_{ij}(t) = \left\|X_i(t), X_j(t)\right\|_{rNorm}; rNorm\text{is usually 2} \tag{13}$$

To give a stochastic characteristic to the algorithm, it is expected that the total force that acts on ith agent in dth dimension be a randomly weighted sum of dth components of the forces exerted from other agents (j) given by (14).

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} rand_j F_{ij}^d \tag{14}$$

where $rand_j$ is a random number in the interval [0, 1], corresponding to the jth agent.

Thus, by the law of motion, the acceleration of the ith agent at iteration cycle t, and in dth dimension, $a_i^d(t)$ is given by (15).

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \tag{15}$$

where $M_{ii}\ (t)$ is the inertial mass of the ith agent.

Furthermore, the next velocity of an agent is considered as a fraction of its current velocity added to its acceleration. Therefore, its position and its velocity can be calculated by employing (16) and (17), respectively.

$$v_i^d\ (t+1) = rand_i \times v_i^d\ (t+1) + a_i^d \tag{16}$$

$$x_i^d\ (t+1) = x_i^d\ (t) + v_i^d\ (t+1) \tag{17}$$

In (16), $rand_i$ is a uniform random variable in [0, 1]. This random number is utilized to obtain a randomized characteristic to the search. The gravitational constant (G) is initialized at the beginning and will be reduced with the iteration cycle to control the search accuracy. In other words, G as a function of the initial value (G_0) and iteration cycle (t) is expressed as in (18). α is a constant with a set value of 20.

$$G = G_0 \exp\left(-\alpha^*\left(\frac{t}{\text{maxcycles}}\right)\right) \tag{18}$$

Gravitational and inertia masses are simply calculated by the error fitness evaluation as defined by (7). A heavier mass indicates a more efficient agent. This means that better agents have higher attractions and walk more slowly. Assuming the equality of the gravitational and the inertia masses, the values of masses are calculated using the error fitness values. Gravitational and inertial masses are updated by the following equations:

$$M_{ai} = M_{pi} = M_{ii} \quad \text{for} i = 1,2,.........N \tag{19}$$

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \tag{20}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{1}^{N} m_i(t)} \tag{21}$$

where $fit_i(t)$ represents the error fitness value of the ith agent at iteration cycle (t), and $worst(t)$ and $best(t)$ are defined in (22) and (23), respectively, for the minimization problem as considered in this work.

$$best(t) = \min_{j\in\{1,......N\}} fit_j(t) \tag{22}$$

$$worst(t) = \max_{j\in\{1,......N\}} fit_j(t) \tag{23}$$

One way to achieve a good compromise between exploration and exploitation is to reduce the number of agents with a lapse of time in (14). Thus, it is supposed that a set of agents with bigger masses apply their forces to the other. However, this policy should be adopted carefully because it may reduce the exploration power and increase the exploitation capability.

To avoid local optimum trapping, the algorithm must explore at the beginning. After lapses of iterations, exploration must fade out, and exploitation must occur. In the improvement of the performance of GSA by controlling exploration and exploitation, only the *Kbest* agents will attract the others. *Kbest* is a function of iteration cycle with the initial value K_0, and it decreases with the iteration cycle. In such a way, all agents apply the force at the beginning, and as iteration cycle progresses, *Kbest* is decreased linearly. In the end, there will be only one agent applying force to the others. Therefore, (14) could be modified as in (24).

$$F_i^d(t) = \sum_{j\in Kbest,\ j\neq i} rand_j F_{ij}^d(t) \tag{24}$$

In (24), *Kbest* is the set of first K agents with the minimum error fitness values and the greatest masses.

3.4. Gravitation Search Algorithm with Wavelet Mutation (GSAWM)

3.4.1. Basic wavelet theory: a concept

Certain seismic signals can be modeled by combining translations and dilations of an oscillatory function with a finite duration called a "wavelet", as shown in Fig. 1 and represented by Ling et al. (2008, 2007) and Daubechies (1990).

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right);\ x\in\Re,\ a,b\in\Re,\ a>0 \tag{25}$$

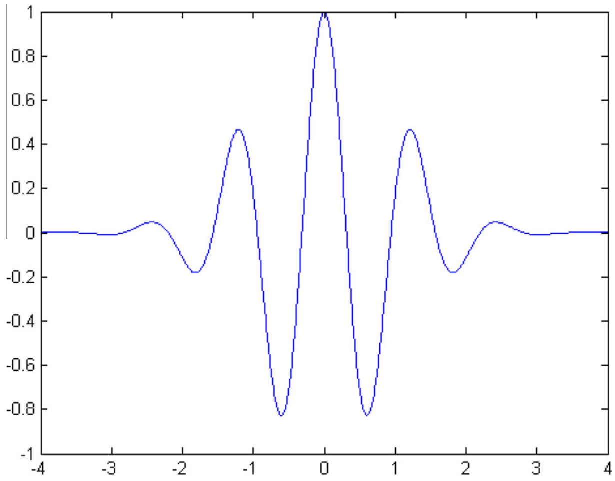

Figure 1 Morlet wavelet.

In (25), a is the dilation (scale) parameter, and b is the translation (shift) parameter. a controls the spread of the wavelet, and b determines its control position. A set of basis function $\psi_{a,b}(x)$ is derived from scaling and shifting the mother wavelet. The basis function of the transform is called the daughter wavelet. The mother wavelet has to satisfy the following admissibility condition:

$$C_\psi = 2\pi \int_{-\infty}^{+\infty} \frac{|\overline{\psi}(\omega)|^2}{\omega} d\omega < \infty \tag{26}$$

In addition, 99% of the energy of $\psi(x)$ is confined to a finite domain [−2.5, 2.5] and is bounded.

3.4.2. Association of wavelet based mutation with GSA (GSAWM)

It is proposed that every element of the particle of the population will mutate. Among the population, a randomly selected ith particle and its jth element (within the limits $[S_{j,\min}, S_{j,\max}]$) at the kth iteration ($S_{i,j}^{(k)}$) of vector $S_i^{(k)}$ will undergo mutation, provided any random number generated within [0, 1] is greater than the probability of mutation p_m. p_m is assigned a low value of 0.15 to ensure mutation will occur in most of the iteration cycles (k), as given by (27).

$$S_{i,j}^{(k)} = \begin{cases} S_{i,j}^{(k)} + \sigma \times (S_{j,\max} - S_{i,j}^{(k)}), & \text{if } \sigma > 0 \\ S_{i,j}^{(k)} + \sigma \times (S_{i,j}^{(k)} - S_{j,\min}), & \text{if } \sigma \leqslant 0 \end{cases} \tag{27}$$

$$\sigma = \frac{1}{\sqrt{a}} e^{\frac{-\left(\frac{x}{a}\right)^2}{2}} \cos\left(5\left(\frac{x}{a}\right)\right) \tag{28}$$

where a is the dilation (scale) parameter, and translation (shift) $b = 0$.

Eq. (27) represents the mutation strategy applied to GSA-based solutions. This strategy is the only way GSAWM differs from GSA. Otherwise, all other steps of GSA and GSAWM are the same. Eq. (28) represents the mother wavelet.

Different dilated Morlet Wavelets are shown in Fig. 2. From this figure, it is clear that as the dilation parameter a increases, the amplitude of $\psi_{a,0}(x)$ or σ (wavelet mutation parameters) will be scaled down. To enhance the searching performance in the fine tuning stage, this property will be utilized in mutation operation. As over 99% of the total energy of the mother wavelet function is contained in the interval [−2.5, 2.5], x can be randomly generated from$[-2.5 \times a, 2.5 \times a]$ (Ling et al., 2008, 2007). The value of the dilation parameter a is set to vary with the value of k/K to meet the fine tuning needs, where k is the current iteration number, and K is the maximum number of iterations. To perform a local search when k is large, the value of a should increase as k/K increases to reduce the significance of the mutation. Thus, a monotonic increasing function governing a and k/K may be written as given in the following equation (Ling et al., 2008, 2007; Daubechies, 1990).

$$a = e^{-\ln(g_1)\times\left(1-\frac{k}{K}\right)^{\xi_{\omega m}}+\ln(g_1)} \tag{29}$$

where $\xi_{\omega m}$ is the shape parameter of the monotonic increasing function, and g_1 is the upper limit of the parameter a. The value of a is thus between 1 and g_1. The variation of a against iteration cycle (k/t) with ξ as parameter is shown in Fig. 2. The magnitude of mutation operator σ decreases as a increases toward g_1 with the increase in iteration cycles (as referred to Fig. 2), thus resulting in appreciable mutation during the early search or exploration stages and fine tuning (i.e., lesser mutation) during the local search or exploitation stage near the end of the maximum iteration cycles. A perfect balance between the exploration of new regions and the exploitation of the already sampled regions in the search space is expected in GSAWM. This balance, which critically affects the performance of the GSAWM, is governed by the right choices of the control parameters, e.g., the swarm size (n_p), the probability of mutation (p_m), and the shape parameter of WM ($\xi_{\omega m}$). Changing the parameter $\xi_{\omega m}$ will change the characteristics of the monotonic increasing function of WM. The dilation parameter a will take a value to perform fine tuning faster as $\xi_{\omega m}$ increases (as referred to Fig. 2). In general, if the optimization problem is smooth and symmetric, it is easier to find the solution, and the fine tuning can be accomplished in early iteration cycles.

Thus, a larger value of $\xi_{\omega m}$ can be used to increase the step size (σ) for the early mutation. Rigorous sensitivity analysis with respect to the dependence of a on (k/K), $\xi_{\omega m}$ and g_1 is performed to determine the individual best values of $\xi_{\omega m}$ and g_1 (Refer to Fig. 2). The individual best values of probability of mutation, p_m, $\xi_{\omega m}$ and g_1 are 0.15, 2.0 and 10,000, respectively.

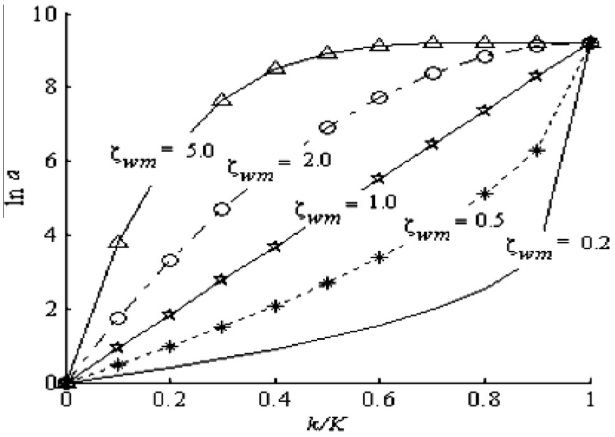

Figure 2 Effect of shape parameter ξ_{wm} to a with respect to $\frac{k}{K}$ with $g_1 = 10{,}000$, $p_m = 0.15$.

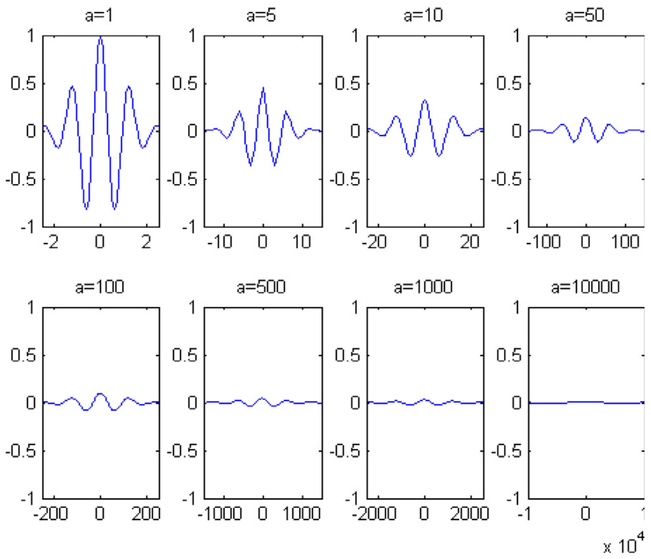

Figure 3 Morlet wavelet dilated by higher and higher values of parameter *a*.

3.4.3. Implementation of the GSAWM for the IIR filter design problem

The steps of the GSAWM, as implemented for the solution of IIR filter design carried out in this work, are shown below:

Step 1. Initialization: Population (swarm size) of agent vectors, $n_p = 25$; maximum iteration cycles = 500; for equal number of numerator and denominator coefficients b_k and a_k, respectively; total coefficients = $(n + 1)*2$ for nth order filter to be designed; minimum and maximum values of the coefficients are -2 and $+2$, respectively; number of samples = 128; $\delta_p = 0.01$, $\delta_s = 0.001$; $\alpha = 20$; $G_0 = 1000$; $rNorm = 2$; $rPower = 1$; initial velocities = zeros $(n_p, \ (n + 1)*2)$; $\xi_{\omega m} = 2.0$; $g_1 = 10{,}000$; $p_m = 0.15$ (see Fig. 3).

Step 2. Generate the initial agent vectors n_p, each having $(n + 1)*2$ number of filter coefficients randomly generated within limits.

Step 3. Compute the error fitness values of the total population, n_p, as defined by (7).

Step 4. Compute the population-based best solution (h_{gbest}) vector.Step 5. Update $G(t)$, $best(t)$, $worst(t)$ and $M_i(t)$ for $i = 1, 2, \ldots, n_p$; t is the current iteration cycle.Step 6. Calculate the total forces in different directions.Step 7. Calculate the accelerations and velocities of agents.Step 8. Update the agents' positions.

Step 9. Compute the wavelet parameters '*a*' as per (29); compute $x = 2.5*a$ if $rand(1) \geqslant 0.5$; otherwise, $x = -2.5*a$; Compute σ as per (28); update the agents' positions as per the new mutation formula (27), and check against the limits of the filter coefficients.Step 10. Repeat Steps 3–9 until the stopping criterion (either maximum iteration cycles or near global optimal solution or agent, h_{gbest}) is met.

Table 1 Design specifications of LP, HP, BP and BS IIR filters.

Type of filter	Pass band ripple (δ_p)	Stop band ripple (δ_s)	Pass band normalized edge frequencies	Stop band normalized edge frequencies
LP	0.01	0.001	0.45	0.50
HP	0.01	0.001	0.35	0.30
BP	0.01	0.001	0.35 and 0.65	0.3 and 0.7
BS	0.01	0.001	0.25 and 0.75	0.3 and 0.7

Table 2 Control parameters of RGA, PSO, GSA and GSAWM.

Parameters	RGA	PSO	GSA	GSAWM
Population size	120	25	25	25
Iteration cycles	500	500	500	500
Crossover rate	1.0	–	–	–
Crossover	Single point crossover	–	–	–
Mutation rate	0.01	–	–	–
Mutation	Gaussian mutation	–	–	–
Selection	Roulette	–	–	–
Selection probability	1/3	–	–	–
C_1	–	2.05	–	–
C_2	–	2.05	–	–
$V_{\min}$	–	0.01	–	–
$V_{\max}$	–	1.0	–	–
Maximum inertia weight ($w_{\max}$)	–	1.0	–	–
Minimum inertia weight ($w_{\min}$)	–	0.4	–	–
α	–	–	20	20
G_0	–	–	1000	1000
$rNorm$	–	–	2	2
$rPower$	–	–	1	1
$\xi_{\omega m}$, p_m, g_1	–	–	–	2.0, 0.15, 10,000

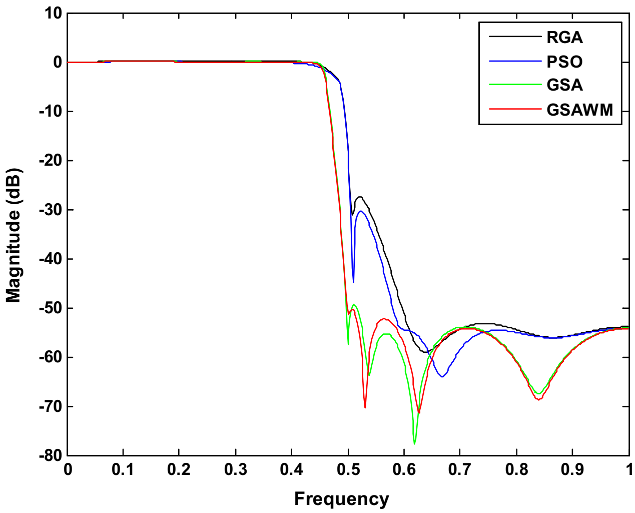

Figure 4 Gain plots in dB for the 8th-order IIR LP filters designed using RGA, PSO, GSA and GSAWM.

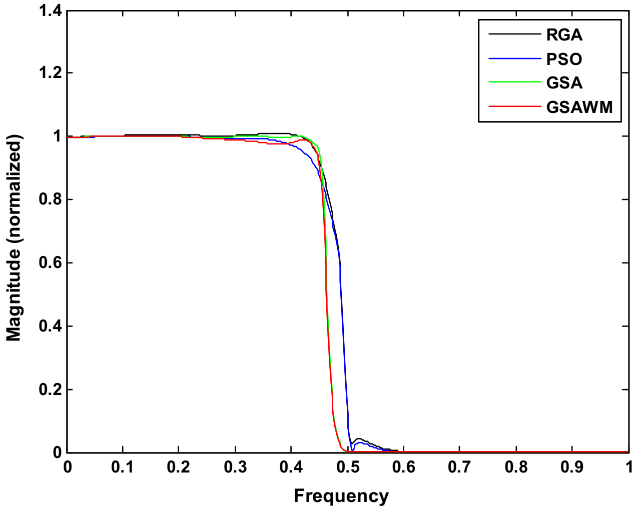

Figure 5 Normalized gain plots for the 8th-order IIR LP filters designed using RGA, PSO, GSA and GSAWM.

Finally, h_{gbest} is the vector of optimal filter coefficients of number $(n + 1)*2$. Extensive simulation studies have been individually performed to compare the optimization performances of the four algorithms RGA, PSO, GSA and GSAWM, respectively, for 8th-order LP, HP, BP and BS IIR filter optimization problems. The design specifications followed for all algorithms are given in Table 1.

4. Results and discussion

The values of the control parameters of RGA, PSO, GSA and GSAWM are given in Table 2. Each algorithm was run 30 times to obtain the best solution, and the best results are reported in this paper. All optimization programs were run in MATLAB version 7.5 on a computer with a 3.00 GHz core (TM) 2 duo processor with 2 GB of RAM.

Three aspects of the algorithms were investigated in this work, namely, their accuracy, speed of convergence and stability. Fig. 4 shows the comparative gain plots in dB for the designed 8th-order IIR LP filters obtained using different algorithms. Fig. 5 represents the comparative normalized gain plots for 8th-order IIR LP filters. The best optimized numerator coefficients (b_k) and denominator coefficients (a_k) obtained are reported in Table 3. Maximum stop band attenuations of 27.5145 dB, 30.3635 dB, 49.3552 dB and 51.9880 dB (the highest) were obtained for the RGA, PSO GSA and GSAWM algorithms, respectively. The gain plots and Tables 4 and 5 also explore that the proposed 8th-order IIR filter design employing GSAWM, which achieved the highest stop band attenuation and the lowest stop band ripple, variance and stan-

Table 4 Statistical data for stop band attenuation (dB) for the 8th-order IIR LP filters.

Algorithm	Maximum	Mean	Variance	Standard deviation
RGA	27.5145	40.3870	165.7013	12.8725
PSO	30.3635	46.5478	130.9666	11.4441
GSA	49.3552	53.1923	5.1670	2.2731
GSAWM	51.9880	53.5667	1.2434	1.1151

Table 3 Optimized coefficients and performance comparison of concerned algorithms for the 8th-order IIR LP filters.

Algorithms	Numerator coefficients (b_k)			Denominator coefficients (a_k)			Maximum stop band attenuation (dB)
RGA	0.0415	0.1234	0.2676	0.9994	−1.1555	2.7421	27.5145
	0.3806	0.4206	0.3484	−2.3022	2.4552	−1.4037	
	0.2164	0.0925	0.0233	0.7776	−0.2480	0.0524	
PSO	0.0413	0.1241	0.2668	1.0001	−1.1546	2.7413	30.3635
	0.3791	0.4202	0.3478	−2.3016	2.4547	−1.4044	
	0.2165	0.0936	0.0235	0.7781	−0.2483	0.0519	
GSA	0.0298	0.0778	0.1680	1.0001	−1.6888	3.3754	49.3552
	0.2378	0.2717	0.2340	−3.4260	3.2805	−2.0249	
	0.1596	0.0739	0.0261	1.0290	−0.3239	0.0594	
GSAWM	0.0300	0.0781	0.1675	1.0005	−1.6891	3.3756	51.9880
	0.2375	0.2713	0.2338	−3.4265	3.2803	−2.0252	
	0.1600	0.0742	0.0262	1.0293	−0.3235	0.0598	

Table 5 Qualitatively analyzed data for the 8th-order IIR LP filters.

Algorithm	Maximum pass band ripple (normalized)	Stop band ripple (normalized)			Transition width
		Maximum	Minimum	Average	
RGA	0.0095	4.2100×10^{-2}	15.7130×10^{-4}	2.1836×10^{-2}	0.0297
PSO	0.0021	3.0300×10^{-2}	6.2811×10^{-4}	1.5464×10^{-2}	0.0338
GSA	0.0028	0.3406×10^{-2}	1.2959×10^{-4}	0.1768×10^{-2}	0.0400
GSAWM	0.0246	2.5154×10^{-3}	3.0415×10^{-4}	1.4098×10^{-3}	0.0423

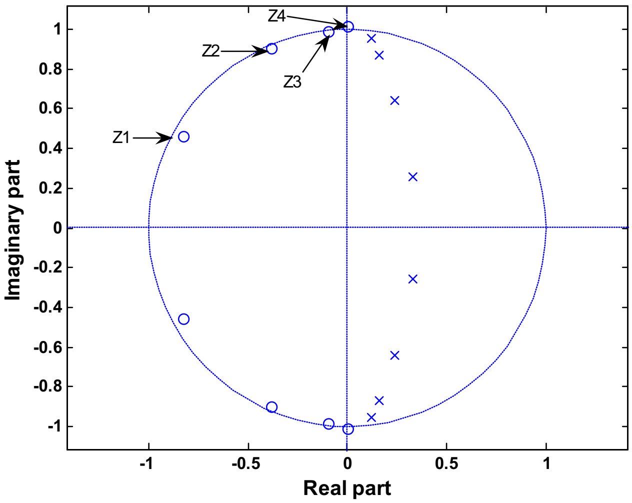

Figure 6 Pole-zero plot of the 8th-order IIR LP filter designed using GSAWM.

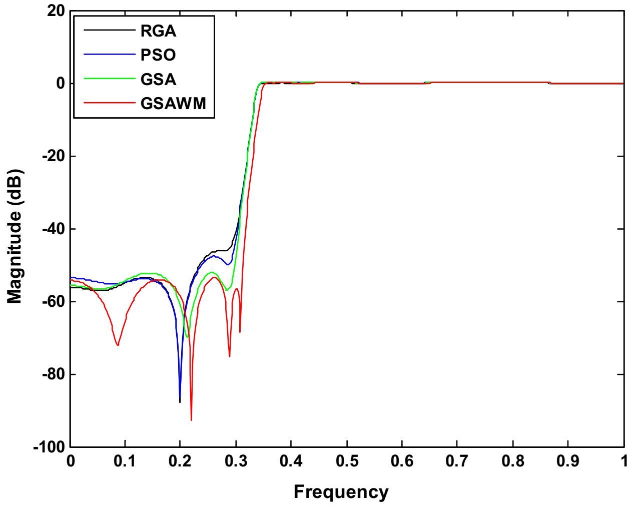

Figure 7 Gain plots in dB for the 8th-order IIR HP filters designed using RGA, PSO, GSA and GSAWM.

Table 6 Radii of zeroes for the 8th order IIR LP filter.

Algorithm	Zeroes			
	*Z*1	*Z*2	*Z*3	*Z*4
GSAWM	0.945084	0.981959	0.993659	1.013003

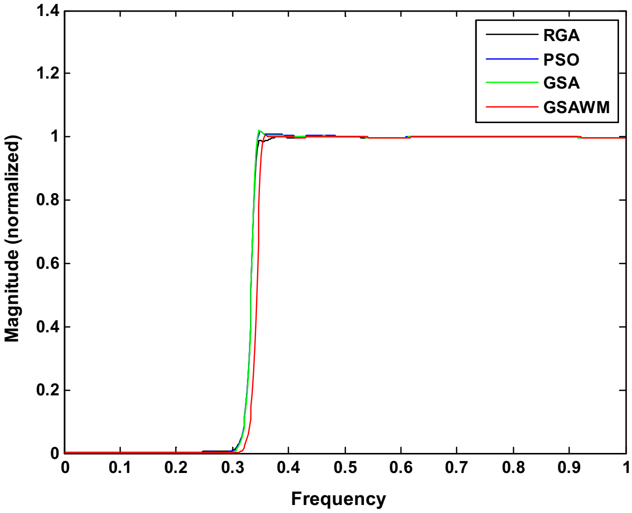

Figure 8 Normalized gain plots for the 8th-order IIR HP filter designed using RGA, PSO, GSA and GSAWM.

dard deviation with an appreciably small transition width and pass band ripple compared with the other algorithms.

Fig. 6 shows the pole-zero plot for a 8th-order IIR LP filter designed with the GSAWM algorithm. This figure demonstrates the existence of poles within the unit circle, which ensures the bounded input bounded output (BIBO) stability condition. The radii of zeroes are also presented in Table 6.

Fig. 7 shows the comparative gain plots in dB for 8th-order IIR HP filters following the individual application of RGA, PSO, GSA and GSAWM optimization techniques, respectively. Fig. 8 represents the comparative normalized gain plots for the 8th-order IIR HP filters. The best optimized numerator coefficients (b_k) and denominator coefficients (a_k) obtained are reported in Table 7. Maximum stop band attenuations of 46.2199 dB, 47.7018 dB, 52.1714 dB and 53.5630 dB (the highest) were obtained for RGA, PSO, GSA and GSAWM algorithms, respectively. Gain plots and Tables 8 and 9 also prove that the proposed optimization technique, GSAWM, achieves the highest stop band attenuation and lowest stop band and pass band ripples, along with the smallest variance and standard deviation, compared with the other algorithms.

Table 7 Optimized coefficients and performance comparison of concerned algorithms for the 8th-order IIR HP filters.

Algorithms	Numerator coefficients (b_k)			Denominator coefficients (a_k)			Maximum stop band attenuation (dB)
RGA	0.1250	−0.7092	1.9588	0.9999	−2.1875	3.8221	46.2199
	−3.3672	3.9090	−3.1264	−3.6220	2.9095	1.3332	
	1.6821	−0.5585	0.0881	0.5678	−0.0861	0.0285	
PSO	0.1252	−0.7091	1.9587	1.0001	−2.1874	3.8222	47.7018
	−3.3671	3.9091	−3.1263	−3.6220	2.9096	−1.3333	
	1.6821	−0.5584	0.0881	0.5678	−0.0861	0.0285	
GSA	0.1252	−0.7092	1.9587	0.9999	−2.1875	3.8222	52.1714
	−3.3672	3.9090	−3.1264	−3.6220	2.9095	−1.3333	
	1.6820	−0.5585	0.0882	0.5679	−0.0861	0.0285	
GSAWM	0.1065	−0.6162	1.7393	1.0000	−2.0569	3.6329	53.5630
	−3.0661	3.6660	−3.0369	−3.4260	2.7458	−1.2287	
	0.1600	−0.5974	0.1018	0.5476	−0.0778	0.0284	

Table 8 Statistical data for stop band attenuation (dB) for the 8th-order IIR HP filters.

Algorithm	Maximum	Mean	Variance	Standard deviation
RGA	46.2199	49.8589	13.2467	2.6391
PSO	47.7018	50.7807	9.4796	3.0789
GSA	52.1714	53.3914	2.7025	1.6439
GSAWM	53.5630	54.6475	1.3718	1.1712

Fig. 9 shows the pole-zero plot of the 8th-order IIR HP filter designed with the GSAWM optimization technique. All poles are within the unit circle, which ensures the stability condition of the designed filter. The radii of zeroes located above the real part of the z plane are shown in Table 10.

Comparative gain plots in dB are provided in Fig. 10. Fig. 11 also represents the comparative normalized gain plots for the 8th-order IIR BP filters designed using the optimization techniques. The best optimized numerator coefficients (b_k) and denominator coefficients (a_k) obtained are reported in Table 11. Maximum stop band attenuations of 18.2445 dB, 20.1389 dB, 24.3104 dB and 25.2100 dB (the highest) were obtained using the RGA, PSO, GSA and GSAWM optimization techniques, respectively. Gain plots and Tables 12 and 13 also indicate that the proposed 8th-order IIR filter design employing GSAWM attains the highest stop band attenuation and lowest pass band and stop band ripples, variance and standard deviation, with the smallest transition width compared with the results produced by others.

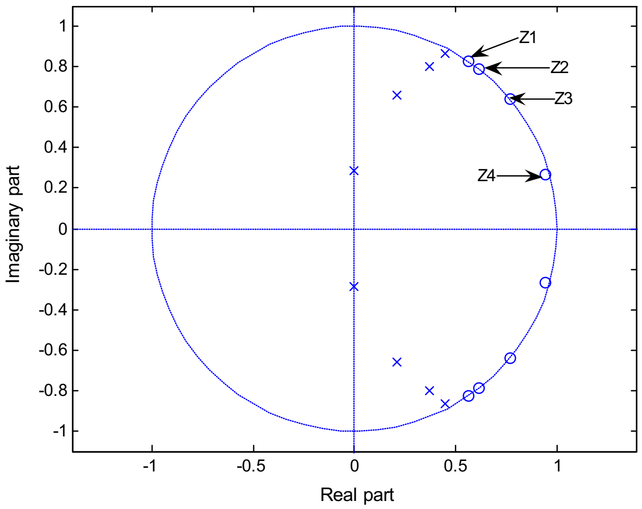

Figure 9 Pole-zero plot of the 8th-order IIR HP filter designed using GSAWM.

Table 10 Radii of zeroes for the 8th order IIR HP filter.

Algorithm	Zeroes			
	*Z*1	*Z*2	*Z*3	*Z*4
GSAWM	1.002023	0.996799	0.999433	0.979007

Fig. 12 shows the pole-zero plot of the 8th-order IIR BP filter designed with the GSAWM optimization technique. The designed filter is stable because of the locations of poles within the unit circle. The radii of zeroes located above the real part of the *z* plane are reported in Table 14.

Table 9 Qualitatively analyzed data for the 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
GSA	0.0207	0.24628×10^{-2}	3.11350×10^{-4}	0.13871×10^{-2}	0.0518
GSAWM	0.0050	2.0982×10^{-3}	2.3257×10^{-5}	1.0607×10^{-3}	0.0379

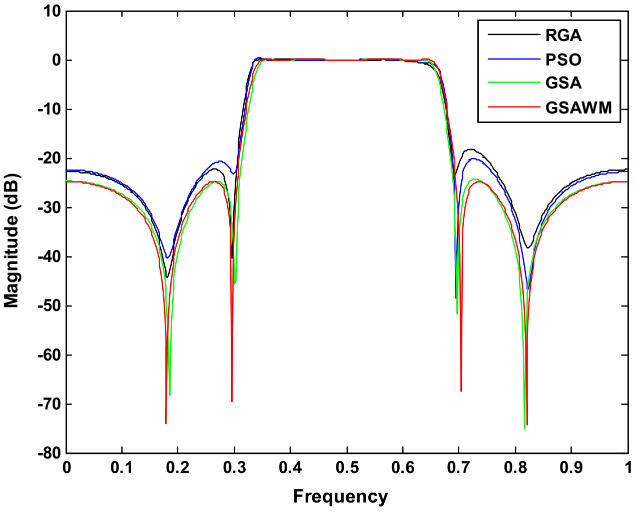

Figure 10 Gain plots in dB for the 8th-order IIR BP filters designed using RGA, PSO, GSA and GSAWM.

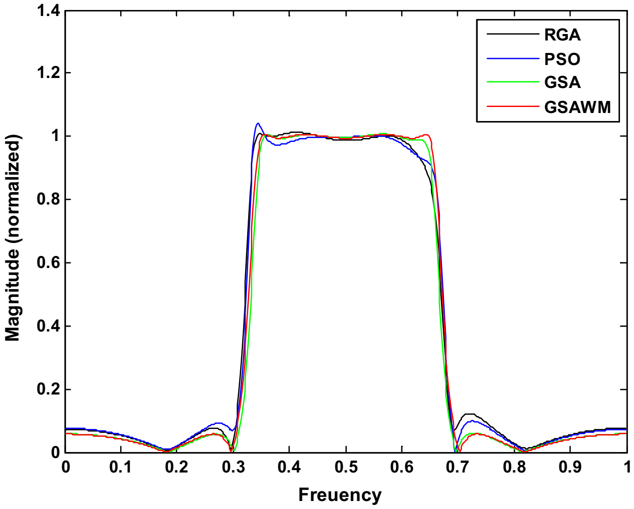

Figure 11 Normalized gain plots for the 8th-order IIR BP filter designed using RGA, PSO, GSA and GSAWM.

Table 12 Statistical data for the stop band attenuation (dB) for the 8th-order IIR BP filters.

Algorithm	Maximum	Mean	Variance	Standard deviation
RGA	18.2445	20.3032	4.2382	2.0587
PSO	20.1389	21.4826	1.8054	1.3437
GSA	24.3104	24.5265	0.0467	0.2161
GSAWM	25.2100	25.22	0.00035	0.0187

Fig. 13 shows the comparative gain plot in dB for the 8th-order IIR BS filters with the application of RGA, PSO, GSA and GSAWM optimization techniques, respectively. Fig. 14 represents the comparative normalized gain plots for the 8th-order IIR BS filters. The best optimized numerator coefficients (b_k) and denominator coefficients (a_k) obtained after extensive simulation study are reported in Table 15. Maximum stop band attenuations of 17.4734 dB, 21.9740 dB, 24.7606 dB and 29.7532 dB (the highest) were obtained for the RGA, PSO, GSA and GSAWM algorithms, respectively. Gain plots and Tables 16 and 17 also explore the proposed optimization technique. GSAWM achieves the highest stop band attenuation, the lowest pass band and stop band ripples and appreciably small transition width compared with the results produced by other algorithms.

Fig. 15 shows the pole-zero plot of a 8th-order IIR BS filter designed using the GSAWM optimization technique. The designed filter is stable because of the locations of poles within the unit circle. The radii of zeroes located above the real part of the z plane are reported in Table 18.

It is observed from Table 4 that maximum stop band attenuations of 27.5145 dB, 30.3635 dB, 49.3552 dB and 51.9880 dB (the highest) were obtained for the RGA, PSO, GSA and GSAWM algorithms, respectively, for the 8th-order IIR LP filter design. In (Gao et al., 2008), Gao et al. applied the CPSO technique for designing an 8th-order IIR LP filter and reported a maximum stop band attenuation of approximately 34 dB. In this work, the proposed algorithm GSAWM displays a much greater stop band attenuation. Luitel et al. reported the design of 9th-order IIR LP filters employing PSO and PSO-QI and approximate attenuations of 22 dB and 27 dB, respectively,

Table 11 Optimized coefficients and performance comparison of concerned algorithms for the 8th order IIR BP filter.

Algorithms	Numerator coefficients (b_k)			Denominator coefficients (a_k)			Maximum stop band attenuation (dB)
RGA	0.1369	−0.0069	−0.0200	0.9971	−0.0075	1.5866	18.2445
	−0.0043	0.1897	0.0069	−0.0094	1.7020	0.0000	
	−0.0338	−0.0056	0.1253	0.8246	−0.0025	0.2247	
PSO	0.1274	0.0071	−0.0209	0.9927	−0.002	1.5940	20.1389
	0.008	0.1857	0.0001	0.0029	1.6978	−0.0002	
	−0.0292	−0.0052	0.1299	0.8079	−0.0034	0.2058	
GSA	0.1040	−0.0003	−0.0158	1.0005	−0.0000	1.7574	24.3104
	0.0006	0.1543	0.0005	0.0003	1.8299	0.0004	
	−0.0162	−0.0003	0.1043	0.8934	−0.0008	0.2168	
GSAWM	0.1069	0.0000	−0.0289	1.0000	0.0000	1.6826	25.2100
	0.0000	0.1607	0.0000	0.0000	1.7546	0.0000	
	−0.0290	0.0000	0.1067	0.8458	0.0000	0.2084	

Table 13 Qualitatively analyzed data for the 8th-order IIR BP filters.

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
GSA	0.0130	6.09×10^{-2}	0.1756×10^{-3}	3.0538×10^{-2}	0.0366
GSAWM	0.0057	5.7610×10^{-2}	1.9104×10^{-4}	2.8901×10^{-2}	0.037

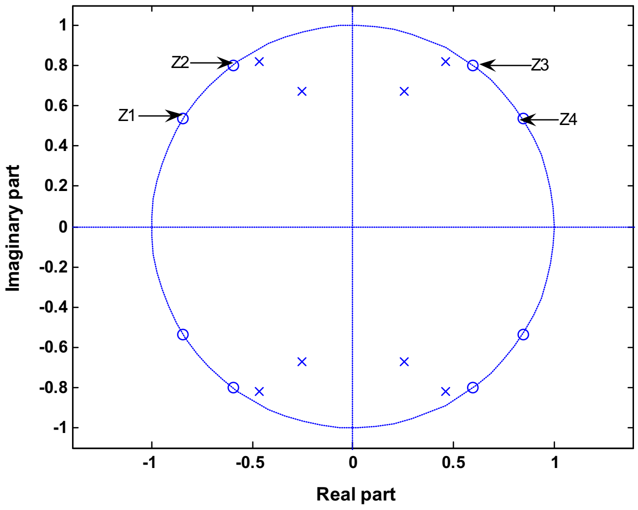

Figure 12 Pole-zero plot of the 8th-order IIR BP filter designed using GSAWM.

Table 14 Radii of zeroes for the 8th order IIR BP filter.

Algorithm	Zeroes			
	Z1	*Z2*	*Z3*	*Z4*
GSAWM	0.999441	0.999956	0.999956	0.999441

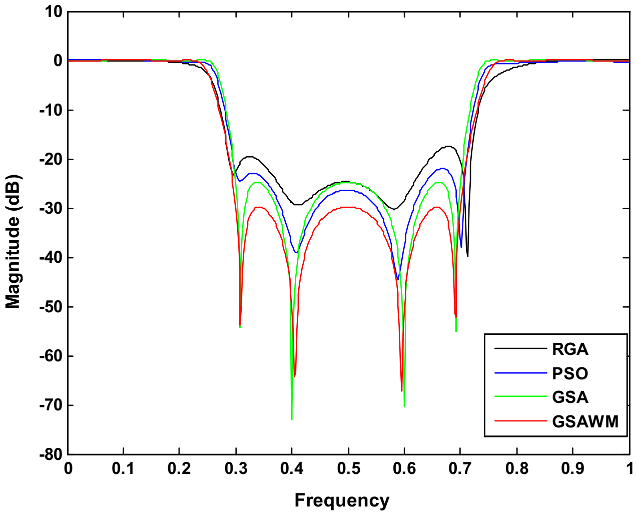

Figure 13 Gain plots in dB for the 8th-order IIR BS filters designed using RGA, PSO, GSA and GSAWM.

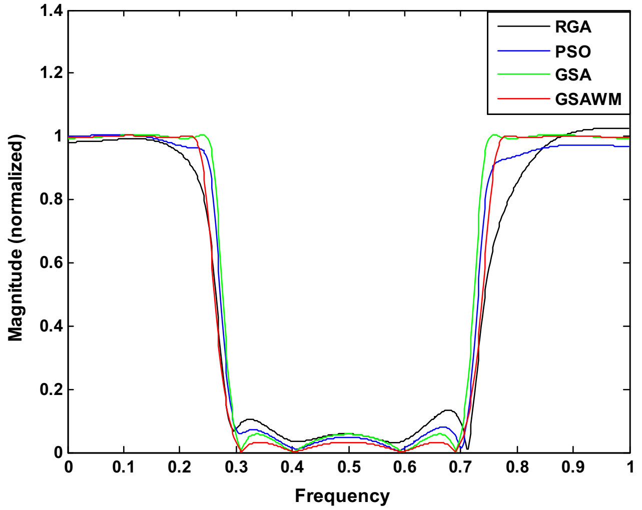

Figure 14 Normalized gain plots for the 8th-order IIR BS filters designed using RGA, PSO, GSA and GSAWM.

were achieved in Luitel and Venayagamoorthy (2008b). Luitel and Venayagamoorthy (2008a) reported for a 9th-order IIR LP filter employing PSO and DEPSO in which maximum attenuations approximately of 25 dB and 22 dB, respectively, were reported. In this paper, the maximum attenuation obtained for PSO is higher even though it is designed with a lower order. In (Karaboga and Cetinkaya, 2004), Karaboga et al. have reported for a 10th-order minimum phase IIR LP filter with a maximum attenuation approximately of 14 dB when GA was employed. In our work, the maximum attenuation of 27.5145 dB for RGA with a lower order was achieved. In (Wang et al., 2011), Wang et al. reported maximum stop band ripples of approximately 0.12, 0.16, 0.15 and 0.05 for 11th-order IIR LP, HP, BP and BS filters, respectively, when the LS-MOEA technique is adopted. In this study, GSAWM yields the improved stop band ripples even with lower-order IIR LP, HP, BP and BS filters, as 2.5154×10^{-3}, 2.0982×10^{-3}, 5.7610×10^{-2} and 3.25×10^{-2}, respectively. The aforementioned results can be verified from Table 19.

5. Comparative effectiveness and convergence profiles of RGA, PSO, GSA and GSAWM

To compare the algorithms in terms of the error fitness values, Fig. 16 depicts the comparative convergences of the error

Table 15 Optimized coefficients and performance comparison of concerned algorithms for the 8th-order IIR BS filters.

Algorithms	Numerator coefficients (b_k)			Denominator coefficients (a_k)			Maximum stop band attenuation (dB)
RGA	0.2269	−0.0189	0.5039	1.0190	−0.0067	0.0968	17.4734
	0.0170	0.6409	−0.0136	0.0109	0.8671	0.0180	
	0.4866	0.0093	0.2189	−0.0322	0.0177	0.1182	
PSO	0.2142	−0.0058	0.4833	1.0073	−0.0069	0.0980	21.9740
	−0.0008	0.6503	0.0097	−0.0077	0.8902	−0.0073	
	0.4976	0.0041	0.2091	−0.0198	−0.0048	−0.1089	
GSA	0.2215	0.0000	0.5175	1.0000	−0.0001	0.1572	24.7606
	0.0001	0.6995	−0.0000	0.0000	0.9085	0.0000	
	0.5172	0.0001	0.2211	0.0055	−0.0001	0.1181	
GSAWM	0.1642	0.0001	0.3912	1.0001	−0.0001	−0.1697	29.7532
	−0.0000	0.5260	0.0001	0.0001	0.8363	0.0000	
	0.3909	0.0001	0.1638	−0.1179	−0.0000	0.0928	

Table 16 Statistical data for stop band attenuation (dB) for the 8th-order IIR BS filters.

Algorithm	Maximum	Mean	Variance	Standard deviation
RGA	17.4734	21.0867	13.0559	3.6133
PSO	21.9740	24.1658	4.8038	2.1918
GSA	24.7606	24.7761	0.0001	0.0116
GSAWM	29.7532	29.7877	0.0010	0.0318

fitness values obtained by RGA, PSO, GSA and GSAWM for the 8th-order IIR LP filters. Similar plots were also obtained for the other filters and are not shown.

As shown in Fig. 16, RGA converges to the minimum error fitness value of 4.054 in 38.9791 s. PSO converges to the minimum error fitness value of 2.850 in 28.4670 s. GSA converges to the minimum error fitness value of 1.825 in 20.185515 s, whereas GSAWM converges to the minimum error fitness value of 1.231 in 22.751 s. The above-mentioned execution times may be verified from Table 20. Similar observations were made for the other filters but are not shown. Table 20 summarizes the convergence profile results for RGA, PSO, GSA and GSAWM applied for the design of IIR LP filters.

From Fig. 16, it can be concluded that the proposed filter design technique using GSAWM obtains the minimum error fitness values compared with PSO, RGA, and GSA. Given the above, it may finally be inferred that the performance of GSAWM is the best among all the mentioned algorithms.

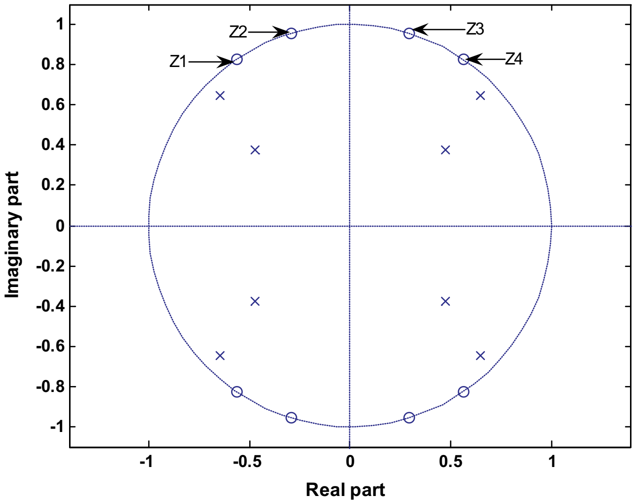

Figure 15 Pole-zero plot of the 8th-order IIR BS filter designed using GSAWM.

Table 18 Radii of zeroes for the 8th-order IIR BS filter.

Algorithm	Zeroes			
	*Z*1	*Z*2	*Z*3	*Z*4
GSAWM	0.999874	0.999708	0.999157	1.000129

Table 17 Qualitatively analyzed data for the 8th-order IIR BS filters.

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
GSA	0.0063	5.78×10^{-2}	0.2207×10^{-3}	2.9010×10^{-2}	0.0395
GSAWM	0.0037	3.25×10^{-2}	4.3684×10^{-4}	1.6468×10^{-2}	0.0563

Table 19 Comparison of performance criteria from by other reported works.

Reference	Algorithm considered	Filter type	Order	Stop band attenuation (dB)	Pass band RIPPLE	Stop band ripple	Transition Width
Karaboga and Cetinkaya (2004)	GA	LP	10th	14	–	–	–
Gao et al. (2008)	CPSO	LP	8th	34	–	–	–
Luitel and Venayagamoorthy (2008a)	PSO, DEPSO	LP	9th	25, 22	–	–	
Luitel and Venayagamoorthy (2008b)	PSO, PSO-QI	LP	9th	22, 27	–	–	–
Wang et al. (2011)	LS-MOEA	LP, HP, BP, BS	11th	–	–	0.12, 0.16, 0.15, 0.05	–
Present paper	GSAWM	LP, HP, BP, BS	8th	51.9880	0.0246	2.5154×10^{-3}	0.0423
				53.5630	0.0050	2.0982×10^{-3}	0.0373
				25.2100	0.0057	5.7610×10^{-2}	0.0370
				29.7532	0.0037	3.2500×10^{-2}	0.0563

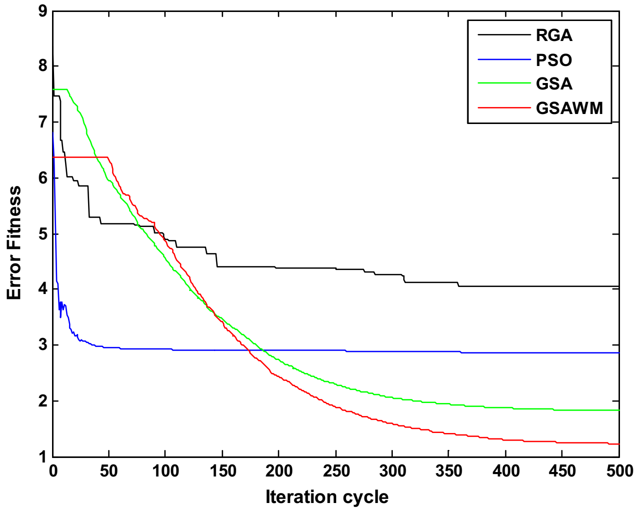

Figure 16 Convergence profiles for RGA, PSO, GSA and GSAWM for the 8th-order IIR LP filters.

Table 20 Convergence profile results for RGA, PSO, GSA and GSAWM for the 8th order IIR LP filter.

Algorithms	Minimum error value	Iteration cycles	Convergence speed (per cycle)	Execution time for 100 cycles (s)
RGA	4.054	500	8.608×10^{-3}	7.795833
PSO	2.850	500	7.918×10^{-3}	5.693405
GSA	1.825	500	11.530×10^{-3}	4.037103
GSAWM	1.231	500	10.284×10^{-3}	4.568521

6. Conclusions

In this paper, the Gravitational Search Algorithm with Wavelet Mutation (GSAWM) along with RGA, PSO and GSA algorithms was applied to the solution of the constrained, multi-modal, non-differentiable, and highly nonlinear IIR low pass, high pass, band pass and band stop filter design problems. It has been established from the results obtained after extensive simulation that the optimal filters, designed with GSAWM, meet the stability criterion and display the best attenuation characteristics with reasonably good transition widths and ripple profiles. GSAWM converges fast to the best quality optimal solutions and reaches the lowest minimum error fitness value in moderately low execution time. It is also evident from the results obtained with a large number of trials that GSAWM is consistently free from the shortcoming of premature convergence exhibited by other optimization algorithms. The statistically improved results obtained for the GSAWM justify the potential of the proposed algorithm in the design of digital IIR filters. It should be noted that the only limitation of the method is the rigorous trials required for the tuning of the long queue of control parameters for the gravitational search algorithm and the wavelet mutation method.

References

Antoniou, A., 2005. Digital Signal Processing: Signals, Systems and Filters. McGraw Hill.

Bahrololoum, A., Nezamabadi-pour, H., Bahrololoum, H., Saeed, M., tul = 0? > Bahrololoum et al., February 2012. A prototype classifier based on gravitational search algorithm. Appl. Soft. Comput. 12 (2), 819–825.

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

Chen, S., Luk, B.L., 2010. Digital IIR filter design using particle swarm optimization. Int. J. Modell. Identif. Control 9 (4), 327–335.

Chen, S., Istepanian, R., Luk, B.L., 2001. Digital IIR filter design using adaptive simulated annealing. Digital Signal Process. 11 (3), 241–251.

Das, S., Konar, A., 2007. A swarm intelligence approach to the synthesis of two-dimensional IIR filters. Eng. Appl. Artif. Intell. 20 (8), 1086–1096.

Daubechies, I., 1990. The wavelet transform, time frequency localization and signal analysis. IEEE Trans. Inf. Theory 36 (5), 961–1005.

Eberhart, R., Shi, Y., 1998. Comparison between genetic algorithm and particle swarm optimization. In: Evolutionary Programming VII. Springer, pp. 611–616.

Fang, W., Sun, J., Xu, W., 2009. A new mutated quantum-behaved particle swarm optimization for digital IIR filter design. EURASIP J. Adv. Signal Process., 1–7.

Gao, Y., Li, Y., Qian, H., 2008. The design of IIR digital filter based on chaos particle swarm optimization algorithm. In: IEEE Int. Conf. on Genetic and Evolutionary Computing, pp. 303–306.

Holland, J.H., 1975. Adaptation in Natural and Artificial Systems. Univ. Michigan Press, Ann Arbor.

Hussain, Z.M., Sadik, A.Z., O'Shea, P., 2011. Digital Signal Processing-An Introduction with MATLAB Applications. Springer-Verlag.

Jackson, L.B., Lemay, G.J., 1990. 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, vol. 3. USA, pp. 1333–1336.

Kalinli, A., Karaboga, N., 2005. Artificial immune algorithm for IIR filter design. Eng. Appl. Artif. Intell. 18 (8), 919–929.

Karaboga, N., Cetinkaya, B., 2004. Design of minimum phase digital IIR filters by using genetic algorithm. In: Proc. IEEE 6th Nordic Signal Processing Symposium, Finland, pp. 29–32.

Karaboga, N., Cetinkaya, M.H., 2011. A novel and efficient algorithm for adaptive filtering: artificial bee colony algorithm. Turk. J. Electr. Eng. Comput. Sci. 19 (1), 175–190.

Karaboga, N., Kalinli, A., Karaboga, D., 2004. Designing digital IIR filters using ant colony optimization algorithm. Eng. Appl. Artif. Intell. 17 (3), 301–309.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proc. IEEE Int. Conf. on Neural Network, vol. 4, pp. 1942–1948.

Lang, M.C., 2000. Least-squares design of IIR filters with prescribed magnitude and phase responses and pole radius constraint. IEEE Trans. Signal Process. 48 (11), 3109–3121.

Ling, S.H., Yeung, C.W., Chan, K.Y., Iu, H.H.C., Leung, F.H.F., 2007. A new hybrid particle swarm optimization with wavelet theory based mutation operation. In: Proc. IEEE Congress on Evolutionary Computation (CEC 2007), Sept. 25–28, pp. 1977–1984.

Ling, S.H., Iu, H.H.C., Leung, F.H.F., Chan, K.Y., 2008. 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.

Ling, S.H., Iu, H.H.C., Chan, K.Y., Lam, H.K., Yeung, C.W., Leung, F.H.F, 2008. Hybrid particle swarm optimization with wavelet mutation and its industrial applications. IEEE Trans. Syst. Man Cybern. Part B 38 (3), 743–763.

Lu, W.-S., Antoniou, A., 2000. Design of digital filters and filter banks by optimization: a state of the art review. In: Proc. European Signal Processing Conf., vol. 1. Finland, pp. 351–354.

Luitel, B., Venayagamoorthy, G.K., 2008. Differential evolution particle swarm optimization for digital filter design. In: IEEE Congress on Evolutionary Computation, pp. 3954–3961.

Luitel, B., Venayagamoorthy, G.K., 2008. Particle swarm optimization with quantum infusion for the design of digital filters. In: *IEEE Swarm Intelligence Symposium*, St. Louis, MO, USA, pp. 1–8.

Mandal, S., Ghoshal, S.P., Kar, R., Mandal, D., 2011. Optimal linear phase FIR band pass filter design using craziness based particle swarm optimization algorithm. J. Shanghai Jiaotong Univ. (Sci. Springer) 16 (6), 696–703.

Mandal, S., Ghoshal, S.P., Kar, R., Mandal, D., 2012. Design of optimal linear phase FIR high pass filter using craziness based particle swarm optimization technique. J. King Saud Univ. Comput. Inf. Sci. (Elsevier) 24, 83–92.

Mandal, S., Mandal, D., Kar, R., Ghoshal, S.P., 2012. Non-recursive fir band pass filter optimization by improved particle swarm optimization. In: Int. Conf. on Information Systems Design and Intelligent Applications-2012, (Springer-Verlag), AISC 132, pp. 405–412, India.

Mondal, D., Ghosal, S.P., Bhattacharya, A.K., 2010. Radiation pattern optimization for concentric circular antenna array with central element feeding using craziness based particle swarm optimization. Int. J. RF Microwave Comput. Aided Eng. (Wiley) 20 (5), 577–586.

Mondal, D., Ghosal, S.P., Bhattacharya, A.K., 2011. Application of evolutionary optimization techniques for finding the optimal set of concentric circular antenna array. Expert Syst. Appl., (Elsevier) 38 (4), 2942–2950.

Oppenheim, A.V., Schafer, R.W., Buck, J.R., 1999. Discrete-Time Signal Processing. Prentice-Hall, NJ.

Pan, S.T., Chang, C.Y., 2011. Particle swarm optimization on D-stable IIR filter design. In: IEEE World Congress on Intelligent Control Automation, WCICA '11, Taipei, pp. 621–626.

Panda, G., Pradhan, P.M., Majhi, B., 2011. IIR system identification using cat swarm optimization. Expert Syst. Appl. 38 (10), 12671–12683.

Proakis, J.G., Manolakis, D.G., 1996. Digital Signal Processing. Prentice-Hall, NJ.

Rashedi, E., Nezamabadi-pour, H., Saryazdi, S., 2009. GSA: a gravitational search algorithm. Inf. Sci. (Elsevier) 179 (13), 2232–2248.

Rashedi, E., Nezamabadi-pour, H., Saryazdi, S., Saeed, M., 2011. Filter modelling using gravitational search algorithm. Eng. Appl. Artif. Intell. (Elsevier) 24 (1), 117–122.

Sun, J., Fang, W., Xu, W., 2010. A quantum-behaved particle swarm optimization with diversity-guided mutation for the design of two-dimensional IIR digital filters. IEEE Trans. Circuits Syst. II 57 (2), 141–145.

Tsai, J.T., Chou, J.H., Liu, T.K., 2006. Optimal design of digital IIR filters by using hybrid Taguchi genetic algorithm. IEEE Trans. Ind. Electron. 53 (3), 867–879.

Wang, B., Li, B., Chen, Y., 2011. Digital IIR filter design using multi-objective optimization evolutionary algorithm. Appl. Soft Comput. (Elsevier) 11, 1851–1857.

Yu, Y., Xinjie, Y., 2007. Cooperative co-evolutionary genetic algorithm for digital IIR filter design. IEEE Trans. Ind. Electron. 54 (3), 1311–1318.