Research Article

Two-Dimensional IIR Filter Design Using Simulated Annealing Based Particle Swarm Optimization

Supriya Dhabal¹ and Palaniandavar Venkateswaran²

¹ Department of Electronics and Communication Engineering, Netaji Subhash Engineering College, Garia, Kolkata, West Bengal 700152, India

² Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata, West Bengal 700032, India

Correspondence should be addressed to Supriya Dhabal; supriya_dhabal@yahoo.co.in

Received 14 May 2014; Revised 16 August 2014; Accepted 23 August 2014; Published 9 September 2014

Academic Editor: Ling Wang

Copyright © 2014 S. Dhabal and P. Venkateswaran. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present a novel hybrid algorithm based on particle swarm optimization (PSO) and simulated annealing (SA) for the design of two-dimensional recursive digital filters. The proposed method, known as SA-PSO, integrates the global search ability of PSO with the local search ability of SA and offsets the weakness of each other. The acceptance criterion of Metropolis is included in the basic algorithm of PSO to increase the swarm's diversity by accepting sometimes weaker solutions also. The experimental results reveal that the performance of the optimal filter designed by the proposed SA-PSO method is improved. Further, the convergence behavior as well as optimization accuracy of proposed method has been improved significantly and computational time is also reduced. In addition, the proposed SA-PSO method also produces the best optimal solution with lower mean and variance which indicates that the algorithm can be used more efficiently in realizing two-dimensional digital filters.

1. Introduction

Design of two-dimensional (2D) filters has been considered extensively over the past two decades as it plays a very significant role in the domain of biomedical image processing, satellite imaging, seismic data processing, and so forth [1]. It is well known that digital filters are typically classified into two groups: recursive or infinite impulse response (IIR) and nonrecursive or finite impulse response (FIR) filters. Design of IIR filters has received much more attention, because IIR filters can provide a better performance than FIR filters, having the same number of filter coefficients. But the main problems of IIR filters are that they have a multimodal error surface and they may also be unstable in some cases. To overcome the problem of multimodal error surface, a global optimization method can be adopted more efficiently. The stability problem can be tackled by limiting the problem space as appropriate constraints from the beginning of optimization routine [1, 2]. Similar to 1D filter, 2D IIR filters can also meet the same desired specifications with less

number of coefficients than required for an equivalent 2D FIR filter. The design methodology for 2D filters grouped in two ways: McClellan transformation based design of 2D filters from 1D prototype and another one based on appropriate optimization techniques [1–6]. In optimization based methods, the design problem can be formulated as a constrained minimization problem and they are solved by various global optimization techniques. Previously reported work on this problem has applied different optimization techniques, like neural network (NN) [1], genetic algorithm (GA) [2], computer language GENETICA [3], Taguchi-based immune algorithm [4], Bees algorithm [5], and particle swarm optimization (PSO) [6], efficiently. Most of these algorithms exhibit slow convergence to achieve a good near-optimum solution and are easily trapped into local optima. These shortcomings can be avoided by introducing SA with PSO because the first one has strong local exploration capabilities and PSO exhibits fast global searching abilities.

Earlier reported works demonstrated that by combining SA with PSO, significant improvements are achieved for

different engineering problems, like partner selection in virtual enterprise [7], job shop scheduling problem [8], holon task allocation scheme [9], energy consumption reduction in embedded systems [10], and numerical verification of several benchmark functions [11]. Zhao et al. [7, 9] and Jamili et al. [8] combined SA with PSO for solving different practical problems, where SA starts with the global best solution generated by PSO. These hybrid algorithms [7–9] can be easily converted to basic PSO by ignoring the SA subroutine, and, at the same time, it can be used as a conventional SA by assuming population size to be one particle. It is demonstrated in [7–9] that the hybrid method outperforms conventional GA, SA, and PSO not only by providing quality solution but also by exhibiting greater convergence speed. Idoumghar et al. [10] and Shieh et al. [11] applied this hybrid algorithm for solving several benchmark functions and also for reducing energy consumption in embedded system memories. It is proved that [10] the hybrid SA-PSO outperforms most of the recently introduced PSO based methods, like QIPSO (quadratic interpolation based PSO), GMPSO (Gaussian mutation based PSO), and so forth, in terms of accuracy, robustness, and convergence speed. Motivated by the improved features of hybrid SA-PSO, here the design problem of 2D IIR filter is implemented more efficiently. Simulation results using this approach ensure better quality solutions with less computational time. The novelty of the proposed work is as follows: (i) new hybrid algorithm for designing 2D recursive filters, (ii) the effective application of Metropolis criterion to escape from local optimum, and (iii) adaptive simulated annealing (ASA) for better control of convergence speed and accuracy.

The rest of the paper is organized as follows. The design problem of 2D IIR filters and the stability condition are presented in Section 2. Section 3 introduces the preliminary concept of PSO. The proposed design method using hybrid SA-PSO is described in Section 4. The simulation results indicating the comparisons with different design examples are shown in Section 5. Finally, conclusions and future work are given in Section 6.

2. Problem Formulation of 2D IIR Filter

Here we consider the design problem of N th order, 2D IIR filter with transfer function [6]:

$$H(z_1, z_2) = H_0 \frac{\sum_{i=0}^N \sum_{j=0}^N a_{ij} z_1^i z_2^j}{\prod_{k=1}^N (1 + b_k z_1 + c_k z_2 + d_k z_1 z_2)}, \quad a_{00} = 1, \quad (1)$$

where $\{a_{ij}, b_k, c_k, d_k\}$ are filter coefficients. Let the frequencies $\omega_1, \omega_2 \in [-\pi, \pi]$, $z_1 = e^{-j\omega_1}$, and $z_2 = e^{-j\omega_2}$. The design task of 2D filters is to find a transfer function $H(z_1, z_2)$ as in (1), such that the magnitude function $M(\omega_1, \omega_2) = |H(e^{-j\omega_1}, e^{-j\omega_2})|$ approximates the desired amplitude response $M_d(\omega_1, \omega_2)$ in

some optimal sense. This approximation can be achieved by minimizing ϕ , where

$$\phi = \phi(a_{ij}, b_k, c_k, d_k, H_0) = \sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} [|M(\omega_1, \omega_2)| - M_d(\omega_1, \omega_2)]^p. \quad (2)$$

Here $\omega_1 = (\pi/N_1)n_1$, $\omega_2 = (\pi/N_2)n_2$, and p is a positive integer (e.g., $p = 2, 4$, or 8). Hence, the primary aim is to minimize the difference between actual and desired amplitude response of the filter at $N_1 \times N_2$ grid points. Since denominator consists of only 1st degree factors, the stability condition is given by

$$|b_k + c_k| - 1 < d_k < 1 - |b_k - c_k|, \quad \text{for } k = 1, 2, \dots, N. \quad (3)$$

Therefore, the design task of 2D recursive filters reduces to the following constrained minimization problem: minimize $\phi = \sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} [|M(\omega_1, \omega_2)| - M_d(\omega_1, \omega_2)]^p$ satisfying the constraints as given in (3).

Assuming that $N = 2$, the transfer function of second-order 2D filter can be written as follows:

$$\begin{aligned} H(z_1, z_2) = H_0 & (a_{00} + a_{01}z_2 + a_{02}z_2^2 + a_{10}z_1 \\ & + a_{20}z_1^2 + a_{11}z_1z_2 + a_{12}z_1z_2^2 \\ & + a_{21}z_1^2z_2 + a_{22}z_1^2z_2^2) \\ & \times ((1 + b_1z_1 + c_1z_2 + d_1z_1z_2) \\ & \times (1 + b_2z_1 + c_2z_2 + d_2z_1z_2))^{-1}. \end{aligned} \quad (4)$$

Now substitute $z_1 = e^{-j\omega_1}$ and $z_2 = e^{-j\omega_2}$ in (4), then $M(\omega_1, \omega_2)$ can be further written as follows:

$$\begin{aligned} M(\omega_1, \omega_2) = H_0 & [(a_{00} + a_{01}c_{01} + a_{02}c_{02} + a_{10}c_{10} \\ & + a_{20}c_{20} + a_{11}c_{11} + a_{12}c_{12} \\ & + a_{21}c_{21} + a_{22}c_{22}) \\ & \times (D)^{-1} \\ & - j(a_{01}s_{01} + a_{02}s_{02} + a_{10}s_{10} + a_{20}s_{20} \\ & + a_{11}s_{11} + a_{12}s_{12} + a_{21}s_{21} + a_{22}s_{22}) \\ & \times (D)^{-1}], \end{aligned}$$

where

$$\begin{aligned}
 D &= [(1 + b_1 c_{10} + c_1 c_{01} + d_1 c_{11}) \\
 &\quad - j(b_1 s_{10} + c_1 s_{01} + d_1 s_{11})] \\
 &\quad \times [(1 + b_2 c_{10} + c_2 c_{01} + d_2 c_{11}) \\
 &\quad - j(b_2 s_{10} + c_2 s_{01} + d_2 s_{11})], \\
 c_{pq} &= \cos(p\omega_1 + q\omega_2), \\
 s_{pq} &= \sin(p\omega_1 + q\omega_2) \quad \text{for } p, q = 0, 1, 2.
 \end{aligned} \tag{5}$$

From (5), $M(\omega_1, \omega_2)$ can be written as follows:

$$M(\omega_1, \omega_2) = H_0 \left(\frac{A_R - jA_I}{(B_{1R} - jB_{1I})(B_{2R} - jB_{2I})} \right),$$

where

$$\begin{aligned}
 A_R &= (a_{00} + a_{01}c_{01} + a_{02}c_{02} + a_{10}c_{10} + a_{20}c_{20} \\
 &\quad + a_{11}c_{11} + a_{12}c_{12} + a_{21}c_{21} + a_{22}c_{22}), \\
 A_I &= (a_{01}s_{01} + a_{02}s_{02} + a_{10}s_{10} + a_{20}s_{20} \\
 &\quad + a_{11}s_{11} + a_{12}s_{12} + a_{21}s_{21} + a_{22}s_{22}), \\
 B_{1R} &= (1 + b_1 c_{10} + c_1 c_{01} + d_1 c_{11}), \\
 B_{1I} &= (b_1 s_{10} + c_1 s_{01} + d_1 s_{11}), \\
 B_{2R} &= (1 + b_2 c_{10} + c_2 c_{01} + d_2 c_{11}), \\
 B_{2I} &= (b_2 s_{10} + c_2 s_{01} + d_2 s_{11}).
 \end{aligned} \tag{6}$$

Hence the corresponding magnitude response of 2D filters can be compactly written as follows:

$$|M(\omega_1, \omega_2)| = H_0 \frac{\sqrt{A_R^2 + A_I^2}}{\sqrt{(B_{1R}^2 + B_{1I}^2)(B_{2R}^2 + B_{2I}^2)}}. \tag{7}$$

To compare the performance of proposed SA-PSO method, the similar design specifications have been used for desired amplitude response [1–6]:

$$M_d(\omega_1, \omega_2) = \begin{cases} 1 & \text{if } \sqrt{\omega_1^2 + \omega_2^2} \leq 0.08 \\ 0.5 & \text{if } 0.08\pi \leq \sqrt{\omega_1^2 + \omega_2^2} \leq 0.12\pi \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Consequently, the desired amplitude response of 2D filter satisfying (8) is depicted in Figure 1.

FIGURE 1: Desired amplitude response $|M_d(\omega_1, \omega_2)|$ of 2D filter.

3. An Overview of Particle Swarm Optimization

Particle swarm optimization (PSO) is a population based evolutionary algorithm which can mimic biological mechanisms, like bird flocking or fish schooling [12]. During the evolutionary process, the swarm particles are attracted towards the location of the best fitness of the particle itself (local best) and the location of the best fitness achieved by the whole population (global best). The position and velocity of i th particle can be represented as follows: $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ and $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. Particle velocities V_i on each dimension are limited to $(V_{\min}, V_{\max})$ which controls the local and global exploration capability of a particle. The best position of each particle is assumed as $pbest_i = (p_{i1}, p_{i2}, \dots, p_{iD})$ and the best fittest particle in the group is $pbest_g = (p_{g1}, p_{g2}, \dots, p_{gD})$. Then, the new velocities and positions for next evaluations are calculated by [12–14]

$$V_{iD} = \omega * V_{iD} + c_1 * rand_1() * (pbest_{iD} - X_{iD}) \tag{9}$$

$$+ c_2 * rand_2() * (gbest_{iD} - X_{iD}),$$

$$X_{iD} = X_{iD} + V_{iD}, \tag{10}$$

where c_1 and c_2 are cognitive and social acceleration control parameters and $rand_1()$ and $rand_2()$ are two random numbers $\in [0, 1]$. One important variant of standard PSO is the constriction factor based PSO (CPSO), which was proposed by Clerc and Kennedy [14]. The CPSO can generate higher-quality solutions than the standard PSO with inertia weight and it guarantees the convergence of the search procedure. In CPSO, the particle velocity is updated by following the equation:

$$\begin{aligned}
 V_{iD} &= \chi (\omega * V_{iD} + c_1 * rand_1() * (pbest_{iD} - X_{iD}) \\
 &\quad + c_2 * rand_2() * (pbest_{gD} - X_{iD})),
 \end{aligned} \tag{11}$$

TABLE 1: Different inertia weight strategies.

Name of inertia weight	Formula of inertia weight	Parameters
Linear decreasing inertia weight (PSO-TVIW)	$\omega = (\omega_{\max} - \omega_{\min}) \times \frac{(\text{iter}_{\max} - \text{iter})}{\text{iter}_{\max}} + \omega_{\min}$	$\omega_{\max} = 0.9$ $\omega_{\min} = 0.4$
Random inertia weight (PSO-RANDIW)	$\omega = 0.5 + \frac{\text{rand}()}{2}$	$\text{rand}() \in [0, 1]$
Chaotic inertia weight (PSO-CIW)	$\omega = (\omega_{\max} - \omega_{\min}) \times \frac{(\text{iter}_{\max} - \text{iter})}{\text{iter}_{\max}} + \omega_{\min} \times z$	$\omega_{\max} = 0.9$ $\omega_{\min} = 0.4$ $z \in [0, 1]$

TABLE 2: Parameter settings for GA and the proposed SA-PSO.

Algorithm	Parameters	Value
GA	Population size	100
	Selection	Roulette wheel, probability 1/3
	Crossover	Two points, rate 0.8
	Mutation	Gaussian, rate 0.01
SA-PSO	Population size	100
	Inertia weight (ω)	$0.9 \rightarrow 0.4$
	c_1, c_2	2.05
	Initial temperature	$T_0 = 50$
	Minimum temperature	$T_{\min} = 10^{-10}$
	Cooling schedule	$C = 0.01, Q = 0.01$

where χ is called constriction factor, given by $\chi = 2 / |2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|$, $\varphi = c_1 + c_2 > 4$.

It is well known that in PSO process, inertia weight (ω) plays an important role for balancing between exploration-exploitation trade-off. In [15, 16], it has been demonstrated that chaotic inertia weight based PSO (PSO-CIW) [17] is the best strategy for better accuracy whereas PSO with random inertia weight (PSO-RANDIW) produces optimal solutions with better efficiency. Therefore, these two best performed inertia weight strategies combined with time varying inertia weight (PSO-TVIW) are considered here to compare the performance of the proposed SA-PSO. The summary of different inertia weight strategies reported earlier is given in Table 1 along with the required constraints.

4. The Proposed SA-PSO Algorithm

The main benefit of PSO is that it is a problem independent, stochastic search optimization method. Due to its stochastic

nature, it reveals insufficient global searching ability at the end of a run, particularly in multimodal functions. Hence to escape from local minima and increase the diversity of particle, the SA is integrated with PSO. Similar to PSO, SA is also a heuristic based random search global optimization method proposed by Kirkpatrick et al. [18]. During the search process, SA accepts not only better candidate solutions but also weakening solutions in certain degree according to Metropolis criterion [19], which can be mathematically represented by

$$a(x) = \begin{cases} 1 & \text{if } \Delta E \leq 0, \\ \exp\left(-\frac{\Delta E}{T}\right) & \text{if } \Delta E > 0, \end{cases} \quad (12)$$

where ΔE denotes change in energy due to perturbation of parameters and T is the current temperature of system. To validate (12), a random number $\rho \in [0, 1]$ is generated and checked for whether $\rho \leq a(x)$ or not. A suitable cooling schedule, based on adaptive simulated annealing (ASA), is introduced here to update temperature of the system [20]. The annealing schedule for k th iteration is written as follows:

$$T(k) = T(0) \exp(-Ck^{Q/D}), \quad (13)$$

where $T(0)$ is starting temperature, Q represents quenching factor, and D is the dimension of search space. Finally, in our proposed method, a well constructed stopping condition is included to minimize the execution time and computational efforts. The proposed algorithm terminates when either of minimum values of objective function and final temperature or maximum number of iterations is reached [21].

The design of 2D recursive filter starts by assuming a population of random solutions in multidimensional search space. Here, each particle consists of 15 positional coordinates which is represented by

$$X = [a_{01} \ a_{02} \ a_{10} \ a_{11} \ a_{12} \ a_{20} \ a_{21} \ a_{22} \ b_1 \ b_2 \ c_1 \ c_2 \ d_1 \ d_2 \ H_0]. \quad (14)$$

Therefore, 2D filter coefficients are represented as a vector X , selected in the interval of $[-3, 3]$. Each entry of X is

optimized based on the flowchart presented in Figure 2 and the proposed design flow is given below.

FIGURE 2: Flowchart of the proposed algorithm.

TABLE 3: Best results of 2D filter coefficients obtained by different approaches for $\rho = 2$.

	NN [1]	GA [2]	GENETICA [3]	TBIA [4]	QPSO DGM [6]	ASA	PSO TVIW	SA-PSO TVIW	SA-PSO RANDIW
a_{01}	1.8922	1.8162	-0.2247	-0.2347	-0.5993	-0.2655	1.8569	0.1947	0.3069
a_{02}	-1.2154	-1.1060	2.5248	-0.3259	-1.9587	-0.0338	1.5657	1.1058	-0.9806
a_{10}	0.0387	0.0712	-0.3498	-0.2724	-0.0447	-0.1779	-1.638	1.0302	0.1681
a_{11}	-2.5298	-2.5132	-2.0915	0.1304	1.6032	0.6370	0.7365	-1.5462	-0.0431
a_{12}	0.3879	0.4279	0.0317	-0.0577	-0.616	-0.2470	1.4300	-0.1602	-0.1820
a_{20}	0.6115	0.5926	2.4656	-0.2779	-2.4065	-0.1133	0.6666	0.8950	-0.7270
a_{21}	-1.4619	-1.3690	0.1652	-0.1220	-0.3271	-0.1192	1.4897	0.0121	-0.3249
a_{22}	2.5206	2.4326	0.7713	0.8314	0.8995	0.9396	0.1710	1.4223	1.6358
b_1	-0.8707	-0.8662	-0.9316	-0.8604	-0.2269	-0.4635	-0.6196	-0.1249	-1.4201
b_2	-0.8729	-0.8907	-0.0249	-0.8607	-0.9187	-0.4357	-0.9312	-1.0553	-0.9178
c_1	-0.8705	-0.8531	-0.9309	-0.8598	-0.4113	-0.5550	-0.6380	-0.1890	-0.6530
c_2	-0.8732	-0.8388	-0.0326	-0.8598	-0.8874	-0.3300	-0.9328	-1.0594	-0.9127
d_1	0.7756	0.7346	0.8862	0.7517	-0.2572	0.1222	0.3233	-0.8986	1.0081
d_2	0.7799	0.8025	-0.8082	0.7517	0.8304	-0.1034	0.8829	1.1366	0.8545
H_0	0.0010	0.0009	0.0010	0.0019	-0.0012	0.0111	0.0002	0.0013	0.0022
ϕ_2	3.7772	6.0276	3.0528	2.9686	3.7731	4.0084	3.0475	2.8932	2.629

Step 1. Specify desired filter (M_d) specifications as given in (8). Assume initial temperature is (T_0) and minimum level of temperature is ($T_{\min}$), initialize $\rho = 2, 4$, or 8 and $N_1 = N_2 = 50$, and use (5) to calculate c_{pq} and s_{pq} for $p, q = 0, 1, 2$ before optimization starts.

Step 2. Initialize swarm size, minimum value of objective function (min.E), random population of coefficients vector (x_i) and maximum number of evaluations ($k_{\max} > 0$).

Step 3. Set the iteration number $k = 1$, assign best solution $x_g = x$, and use (2) for computing the fitness of the best solution p_{best} .

Step 4 (calculate fitness). Compute fitness values for each particle in the swarm.

Step 5 (update p_{best}). Particle best position p_{best} is updated based on Metropolis criterion (12) as follows: if current fitness of selected particle $<$ fitness of p_{best} , then current position of the particle is accepted as new p_{best} with probability 1; otherwise current particle will be accepted according to $\rho \leq a$, where $\rho \in [0, 1]$ and $a = \exp(-(\phi(p_{\text{current}}) - \phi(p_{\text{best}}))/T)$.

Step 6 (update g_{best}). Now assign best particles p_{best} value to g_{best} and calculate the new velocity and position by (9) and (10) for all particles.

Step 7 (reduce Temperature). Calculate new temperature T specified in cooling schedule (13). If $T \leq T_{\min}$, then terminate searching for best solution; otherwise go to Step 8.

Step 8 (update velocities and positions). Calculate velocities V_i using (9) and positions for each particle by (10).

Step 9. Update $k = k + 1$, and check whether $k \geq k_{\max}$ or minimum value of cost function is achieved by any particle. If yes, then assign $x = g_{\text{best}}$; otherwise go back to Step 4.

Step 10. Design 2D filter based on coefficients of $x = g_{\text{best}}$ and compute the magnitude response.

5. Simulation Results and Discussion

In order to validate the performance of proposed SA-PSO, we implemented the algorithm using MATLAB 2009 on Genuine Intel(R) Core 2 Duo CPU E7300 @2.66 GHz, 2 GB RAM. Here, all the associated algorithms are executed for 20 independent runs with 40,000 functional evaluations (FE). Table 2 shows the choice of several required parameters for GA and SA-PSO based methods. In case of proposed SA-PSO based methods, initial population size is chosen as 100. It is also observed that with the further increase in population size, the solution quality remains unchanged. In SA-PSO-TVIW, the inertia weight is decreased from 0.9 to 0.4 to obtain the best possible solutions whereas in SA-PSO-RANDIW it is varied randomly in between 0.5 and 1. The starting temperature for annealing is chosen to be very high, so that it can escape from local optima easily. The parameters for cooling schedule are chosen to be very small to assure global convergence; otherwise it can skip the true global solutions [20, 21].

Table 3 shows an analysis of the experimental results of the proposed method with NN (2001), GA (2003), GENETICA (2006), TBIA (2008), QPSO DGM (2010), ASA, and PSO-TVIW with respect to the best known solutions. The best results among different methods are highlighted in bold

FIGURE 3: Amplitude response $|M(\omega_1, \omega_2)|$ of 2D filter for $\rho = 2$; (a) SA-PSO-TVIW, (b) SA-PSO-RANDIW, (c) NN, (d) GA, (e) GENETICA, and (f) TBIA.

and it can be easily verified that the experimental results of the proposed method, that is, SA-PSO-TVIW and SA-PSO-RANDIW, are better than all other reported methods [1–6]. For example, the percentage improvement in ϕ_2 provided by the proposed SA-PSO-TVIW method in comparison to

earlier reported methods [1–4, 6] is 23.4%, 52%, 5.23%, 2.54%, and 23.32%, respectively.

Further, in Table 4, for different values of ρ , the best possible solutions (best) and worst solutions (worst) are listed for different methods which clearly reveal that the

FIGURE 4: Amplitude response $|M(\omega_1, \omega_2)|$ of 2D filter for $\rho = 4$; (a) SA-PSO-TVIW and (b) SA-PSO-RANDIW.

TABLE 4: Comparative results of different algorithms in terms of best and worst case solution.

Algorithms	$\rho = 2$		$\rho = 4$		$\rho = 8$	
	Best	Worst	Best	Worst	Best	Worst
GA	3.1574	7.4938	0.2596	0.8510	0.0025	0.0145
ASA	4.0084	4.7473	0.2472	0.3161	0.0032	0.0097
PSO-TVIW	3.0475	6.5456	0.2024	0.6223	0.0020	0.0075
PSO-CIW	2.9985	4.7608	0.2008	0.4276	0.0016	0.0063
PSO-RANDIW	2.9604	5.2970	0.1899	0.3158	0.0017	0.0098
SA-PSO (TVIW)	2.8932	4.7777	0.2263	0.4319	0.0030	0.0106
SA-PSO (RANDIW)	2.629	3.0264	0.1599	0.2009	0.0011	0.0040

best/worst solutions obtained by the proposed SA-PSO based algorithm are best for all runs. Furthermore, Table 5 presents the statistics for average (mean) and variance (VAR) of the optimal solutions for different values of ρ . From Table 5, it can be observed that the proposed SA-PSO algorithm exhibits better performance with less computational time (i.e., less than half time as compared to GA or ASA) and produces optimal solution with very lower variance. Hence, it can be implemented more efficiently in real-time designing of 2D filters.

Figures 3(a)–3(f) show the calculated amplitude response of 2D filter for $\rho = 2$. The best results obtained by SA-PSO-TVIW and SA-PSO-RANDIW are shown in Figures 3(a) and 3(b), respectively. For comparison purpose, the results obtained by NN, GA, GENETICA, and TBIA are also included in Figure 3(c) to Figure 3(f). A closer look at these figures reveals that SA-PSO based methods, that is, SA-PSO-TVIW and SA-PSO RANDIW, yield a better approximation to the desired response and stop-band ripple is much less with respect to other competitive methods [1–6]. Figures 4(a)

and 4(b) show the magnitude response obtained using SA-PSO based methods for $\rho = 4$ whereas Figures 5(a) and 5(b) represents the magnitude response for $\rho = 8$.

Figures 6(a) and 6(b) show the best convergence profiles of SA-PSO based methods for $\rho = 2$ and 4, respectively. Among the 20 independent executions of four different algorithms (the existing two algorithms: GA and PSO-RANDIW and the proposed two algorithms: SA-PSO-TVIW and SA-PSO-RANDIW), the best performed runs are plotted here. Initially for few thousands FEs, PSO-RANDIW converges faster than GA, SA-PSO-RANDIW, and SA-PSO-TVIW. After certain number of iterations, the PSO-RANDIW and GA both exhibit premature convergence and settle to near optimal solutions. After 30,000 FEs, SA-PSO based methods fall below the curves of both GA and PSO, because SA helps particles to jump out from local optima from the beginning of a search. Due to this, at the end of search procedure, the proposed hybrid method provides best candidate solution. Hence, the proposed hybrid algorithm produces better optimal solutions during the search process of 2D recursive filters.

6. Conclusions

In this paper, a novel hybrid evolutionary algorithm based on PSO and SA is proposed for finding the global best solution of second-order, two-dimensional recursive digital filters. The proposed hybrid method integrates the global search capability of PSO with SA to escape from local minima. The experimental results are compared with earlier reported algorithms, namely, NN, GA, GENETICA, and TBIA, in addition to different variants of PSO. The comparison results indicate that SA-PSO based methods exhibit better performance in all experiments and provide best optimum solution during search mechanism. Also, the proposed method produces the best solution with lower mean and variance. Hence, the proposed approach could be an alternative to implement the two-dimensional digital filters for real-time applications.

TABLE 5: Comparative results of different algorithms in terms of mean and VAR.

Algorithms	$\rho = 2$		$\rho = 4$		$\rho = 8$		Time (S)
	Mean	VAR	Mean	VAR	Mean	VAR	
GA	4.6137	0.6160	0.4300	0.0688	0.0066	$7.08E - 06$	89.80
ASA	4.2475	0.0288	0.2679	$3.14E - 04$	0.0043	$3.14E - 06$	62.04
PSO-TVIW	4.4858	0.2347	0.3044	0.0086	0.0039	$1.60E - 06$	46.84
PSO-CIW	4.2127	0.2778	0.2847	0.0022	0.0035	$1.42E - 06$	39.02
PSO-RANDIW	4.066	0.3453	0.2549	0.0013	0.0037	$5.50E - 07$	40.54
SA-PSO (TVIW)	3.8762	0.5053	0.3123	0.0029	0.0051	$3.61E - 06$	35.68
SA-PSO (RANDIW)	2.9059	0.0094	0.1845	$1.58E - 04$	0.0023	$7.28E - 07$	30.07

FIGURE 5: Amplitude response $|M(\omega_1, \omega_2)|$ of 2D filter for $\rho = 8$; (a) SA-PSO-TVIW and (b) SA-PSO-RANDIW. FIGURE 6: Best fitness profile for GA, PSO-RANDIW, SA-PSO-TVIW, and SA-PSO-RANDIW; (a) $\rho = 2$ and (b) $\rho = 4$.

Emphasis on higher order two-dimensional recursive filter design with guaranteed stability and further reduction in computational complexity is left as future work.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] V. M. Mladenov and N. E. Mastorakis, "Design of two-dimensional recursive filters by using neural networks," *IEEE Transactions on Neural Networks*, vol. 12, no. 3, pp. 585–590, 2001.
- [2] N. Mastorakis, I. F. Gonos, and M. N. S. Swamy, "Design of two-dimensional recursive filters using genetic algorithms," *IEEE Transactions on Circuits and Systems*, vol. 50, no. 5, pp. 634–639, 2003.
- [3] I. F. Gonos, L. I. Virirakis, N. E. Mastorakis, and M. N. S. Swamy, "Evolutionary design of 2-dimensional recursive filters via the computer language GENETICA," *IEEE Transactions on Circuits and Systems II*, vol. 53, no. 4, pp. 254–258, 2006.
- [4] J.-T. Tsai, W.-H. Ho, and J.-H. Chou, "Design of two-dimensional recursive filters by using Taguchi-based immune algorithm," *IET Signal Processing*, vol. 2, no. 2, pp. 110–117, 2008.
- [5] D. T. Pham and E. Koç, "Design of a two-dimensional recursive filter using the bees algorithm," *International Journal of Automation and Computing*, vol. 7, no. 3, pp. 399–402, 2010.
- [6] J. Sun, W. Fang, and W. Xu, "A quantum-behaved particle swarm optimization with diversity-guided mutation for the design of two-dimensional IIR digital filters," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 2, pp. 141–145, 2010.
- [7] F. Zhao, Q. Zhang, D. Yu, X. Chen, and Y. Yang, "A hybrid algorithm based on PSO and simulated annealing and its applications for partner selection in virtual enterprise," in *Proceedings of the International Conference on Advances in Intelligent Computing*, pp. 380–389, Springer, 2005.
- [8] A. Jamili, M. A. Shafia, and R. Tavakkoli-Moghaddam, "A hybrid algorithm based on particle swarm optimization and simulated annealing for a periodic job shop scheduling problem," *International Journal of Advanced Manufacturing Technology*, vol. 54, no. 1–4, pp. 309–322, 2011.
- [9] F. Zhao, Y. Hong, D. Yu, Y. Yang, Q. Zhang, and H. Yi, "A hybrid algorithm based on particle swarm optimization and simulated annealing to holon task allocation for holonic manufacturing system," *International Journal of Advanced Manufacturing Technology*, vol. 32, no. 9–10, pp. 1021–1032, 2007.
- [10] L. Idoumghar, M. Melkemi, R. Schott, and M. I. Aouad, "Hybrid PSO-SA type algorithms for multimodal function optimization and reducing energy consumption in embedded systems," *Applied Computational Intelligence and Soft Computing*, vol. 2011, Article ID 138078, 12 pages, 2011.
- [11] H.-L. Shieh, C.-C. Kuo, and C.-M. Chiang, "Modified particle swarm optimization algorithm with simulated annealing behavior and its numerical verification," *Applied Mathematics and Computation*, vol. 218, no. 8, pp. 4365–4383, 2011.
- [12] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.
- [13] Y. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings of the IEEE International Congress on Evolutionary Computation*, vol. 3, pp. 101–106, 1999.
- [14] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.
- [15] A. Nickabadi, M. M. Ebadzadeh, and R. Safabakhsh, "A novel particle swarm optimization algorithm with adaptive inertia weight," *Applied Soft Computing Journal*, vol. 11, no. 4, pp. 3658–3670, 2011.
- [16] J. C. Bansal, P. K. Singh, M. Saraswat, A. Verma, S. S. Jadon, and A. Abraham, "Inertia weight strategies in particle swarm optimization," in *Proceedings of the 3rd World Congress on Nature and Biologically Inspired Computing (NaBIC '11)*, pp. 633–640, Salamanca, Spain, October 2011.
- [17] Y. Feng, G.-F. Teng, A.-X. Wang, and Y.-M. Yao, "Chaotic inertia weight in particle swarm optimization," in *Proceedings of the 2nd International Conference on Innovative Computing, Information and Control (ICICIC '07)*, p. 475, Kumamoto, Japan, September 2007.
- [18] S. Kirkpatrick, J. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [19] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.
- [20] L. Ingber, "Adaptive simulated annealing (ASA): lessons learned," *Journal of Control and Cybernetics*, vol. 25, no. 1, pp. 33–54, 1996.
- [21] S. Dhabal and P. Venkateswaran, "An efficient nonuniform cosine modulated filter bank design using simulated annealing," *Journal of Signal and Information Processing*, vol. 3, no. 3, pp. 330–338, 2012.