

# DAPSO and PSO-VAF in Linear Phase Digital Low Pass FIR Filter Design

Mudit Shukla, G. R. Mishra

Department of Electronics & Communication Engineering, Amity University Uttar Pradesh, Lucknow, India  
Email: [mudit.shuklaec@gmail.com](mailto:mudit.shuklaec@gmail.com), [gmishra@gmail.com](mailto:gmishra@gmail.com)

Received 4 January 2014; revised 4 February 2014; accepted 14 February 2014

Copyright © 2014 by authors and Scientific Research Publishing Inc.  
This work is licensed under the Creative Commons Attribution International License (CC BY).  
<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

Digital filters play a key role in the field of digital signal processing. This paper presents a linear phase digital low pass finite impulse response (FIR) filter design using particle swarm optimization and its two new variants, dynamic and adjustable particle swarm optimization (DAPSO) and particle swarm optimization with variable acceleration factor (PSO-VAF) and illustrates the superiority of the PSO-VAF method over PSO based methods. Two fitness functions are considered. The fitness1 is used to find the possible minimum ripples in pass band and stop band in case of PSO, DAPSO and PSO-VAF. Fitness2 is able to control the ripples in both bands separately. A comparison of simulation results demonstrates the performance of PSO and its methods in designing digital low pass FIR filters.

## Keywords

Digital Filters Design; Low Pass FIR Filters; PSO; DAPSO; PSO-VAF

---

## 1. Introduction

In the modern age, digital signal processing (DSP) is the indispensable part of the human life, due to its numerous applications such as telecommunication, speech processing, consumer electronics systems, biomedical systems, image processing, military and defense electronics systems, aerospace and automotive electronics systems and industrial applications [1]. DSP has two major classes of systems. The first system is used to perform signal filtering in time domain and known as digital filter. The second system gives signal representation in frequency domain and known as spectrum analyzer. Digital filters are the most essential element of the DSP and classified into two types, finite impulse response (FIR) and infinite impulse response (IIR). FIR filters or non-recursive filters are those for which the output of the filter depends only on the present input. FIR filters are widely used due to its advantages like it is inherently stable since the poles lie within the unit circle and can be designed as

linear phase filters, making them a better choice in phase sensitive application [2].

There are many different techniques available for the design of digital filters, such as window methods, frequency sampling methods and Parks-McClellan equiripple algorithm. The simplest and most popular way to design FIR filter is by windowing. In this method, ideal impulse response is multiplied with different window functions, such as Butterworth, Chebyshev, Kaiser and Hamming etc., depending on the requirements of ripples on the pass and stop band, stop band attenuation and transition width. But this method does not allow controlling the approximation errors in different bands. In frequency sampling method, any kind of frequency response can be approximated. But in this method, there is no direct formula to calculate the filter order [3]. Parks-McClellan (PM) method based on Remez Exchange algorithm provides an optimum equiripple approximation to the desired frequency response. But in this method, the relative values of the amplitude error in the frequency bands are specified by weighting function and not by deviations themselves [4]. In these methods, designer always has to compromise on one or more of the design specifications.

The intelligent optimization techniques have been successfully implemented in the design of digital filters and provide better parameter control as well as better approximate the ideal filter. Simulated annealing, Tabu search, Differential evolution and artificial bee colony algorithm are some intelligent optimization techniques, which have proved their capability of designing digital filters. Simulated annealing (SA) was used to design linear phase digital filter, Nyquist filter and cascade form FIR filter. However, in this approach computation time is quite long (especially for higher order filter) and does not guarantee finding the global optimum [5]. The tabu search was applied to design approximation problem of FIR digital filter with quantized coefficients using a flexible realization of the filter taps, which allows getting higher accuracy [6]. Another population based algorithm, differential evolution (DE), was applied to design with different order FIR filter. However, this approach is computationally less expensive than Genetic algorithm [7]. A new design method based on artificial bee colony algorithm (ABC) was used to design IIR filters [8].

The particle swarm optimization (PSO) is population based intelligent optimization technique that has proven to be effective in multidimensional nonlinear environment; all of the constraints of filter design can be effectively taken care of by the use of PSO. PSO and its several modifications have been successfully implemented in digital filter designing problems [9]-[14]. In this paper, we propose two new variants of PSO for designing linear phase low pass FIR (LP FIR) filter.

## 2. FIR Filter Design Issues

The transfer function of FIR digital filter is given by

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

And the frequency response of FIR digital filter is given by

$$H(e^{j\omega}) = \sum_{n=0}^N h(n)e^{-j\omega n} \quad (2)$$

In this paper we design an even order, odd length and even symmetry low pass FIR filter. For this the symmetry condition is given as

$$h(n) = h(N + 2 - n) \quad (3)$$

where  $N$  is the order of the filter and from (3) it is clear that the number of coefficients is to be optimized ( $N/2 + 1$ ). The main objective of FIR filter design is to find the filter coefficients in optimized way that results in optimum filter. The filter is optimum, when maximum weighted error is minimized. In the PM algorithm [15], an approximate error function is defined by

$$E(\omega) = G(\omega) [H_d(e^{j\omega}) - H_i(e^{j\omega})] \quad (4)$$

where  $H_d(e^{j\omega})$  and  $H_i(e^{j\omega})$  are the frequency responses of the designed approximate filter and ideal filter respectively. The weighting function,  $G(\omega)$ , is used to provide the approximation error differently in different frequency bands. For an ideal LP filter,  $H_i(e^{j\omega})$  is defined as

$$H_i(e^{j\omega}) = 1 \text{ for } 0 \leq \omega \leq \omega_c \quad (5)$$

$$0 \text{ otherwise}$$

where  $\omega_c$  is the cutoff frequency. In this paper we consider two fitness functions for designing LP FIR Filter [16]. The first fitness function is used to design optimum filter, means finding the possible minimum ripples in both, pass band ( $\delta_p$ ) and stop band ( $\delta_s$ ). This fitness function is given as

$$fitness1 = \max_{\omega \in F_p \cup F_s} (|E(\omega)|) \quad (6)$$

where  $F_p$  and  $F_s$  are the closed subsets  $0 \leq \omega \leq \omega_p$  and  $\omega_s \leq \omega \leq 1$ , respectively. The other fitness function is used to control the ripples in pass band and stop band separately, and is given as

$$fitness2 = \max_{\omega \leq \omega_p} (|E(\omega)| - \delta_p) + \max_{\omega \geq \omega_s} (|E(\omega)| - \delta_s) \quad (7)$$

where  $\omega_p$  and  $\omega_s$  are the pass band and stop band normalized cutoff frequencies.

### 3. Intelligent Optimization Techniques

#### 3.1. Particle Swarm Optimization (PSO)

PSO is inspired by the observation of social behavior of bird flocking and fish schooling. This powerful global optimization technique was first found by Kennedy (a social psychologist) and Eberhart (an electrical engineer) in 1995 [17]. PSO is simple, fast, requires less storage and can be coded in few lines. In PSO every particle remembers its best solution ( $X_i^{best}$ ) as well as the group's best solution ( $X_{best}$ ). It means that PSO have good memory. The PSO is worked on the concept of "constructive cooperation" between particles, so that it is easily able to solve multidimensional optimization problems. In PSO global optimum is achieved by an iterative procedure. The PSO technique is based on the five basic principles of the swarm intelligence [18]. These are, *Proximity*, i.e., the swarm must be able to perform simple space and time computations. *Quality*, i.e., the swarm should be able to respond to quality factors in the environment. *Diverse response*, i.e., the swarm should not commit its activities along excessively narrow channels. *Stability*, i.e., the swarm should not change its behavior every time the environment changes. *Adaptability*, i.e., the swarm must be able to change its behavior, when the computation cost is affordable.

PSO starts with a population of random particles (potential solution) in a D-dimension space. A position 'X' and velocity 'V' are associated with each particle. The position and velocity of the  $i^{th}$  particle are given as

$$X_i = (X_{i1}, X_{i2}, \dots, X_{iN}) \quad (8)$$

$$V_i = (V_{i1}, V_{i2}, \dots, V_{iN}) \quad (9)$$

The velocity and position are updated according to the formula given as

$$V_i^{n+1} = \omega^n V_i^n + C_1 * rand_1 * (X_i^{best} - X_i) + C_2 * rand_2 * (X^{best} - X_i)$$

$$X_i^{n+1} = X_i + V_i^{n+1} \quad (11)$$

where  $X_i^{best}$  and  $X^{best}$  is the individual best and global best positions respectively,  $X_i$  is the current position of the  $i^{th}$  particle, n+1 and n denote the current and the previous iterations,  $rand_1$  and  $rand_2$  are random numbers in the range [0,1]. These random numbers are update every time they occur.  $C_1$  and  $C_2$  are the two positive constants, called cognitive and social acceleration factors respectively and  $\omega^n$  is the inertia weight in the  $n^{th}$  iteration. A linearly damped inertia weight is preferred for better convergence [19].

The PSO algorithm for filter designing is as follows:

- 1) Define the filter specifications, fitness function, and population size and set the boundaries, i.e. maximum and minimum value of coefficient.
- 2) Initialize a population array of particles with random positions and velocities in the problem space.
- 3) For each particle, Compare particle's fitness evaluation with its  $X_i^{best}$  and  $X^{best}$ . If fitness(x) better than

fitness ( $X_i^{best}$ ) then  $X_i^{best} = X_i$  (current value) and if fitness(x) better than fitness ( $X_{best}$ ) then  $X^{best} = X_i$ .

- 4) Update the velocity according to (10) and Move each particle to new position according to (11).
- 5) Loop from 2 - 4 until stopping criterion is satisfied.
- 6) Output is the coefficient of the desired filter ( $N/2 + 1$ ).

### 3.2. Dynamic and Adjustable Particle Swarm Optimization (DAPSO)

In order to improve the performance of PSO and maintain the diversities of the particles, a novel algorithm called Dynamic and Adjustable Particle Swarm Optimization (DAPSO) is proposed [20]. The distance from each particle to the  $X^{best}$  position is calculated in order to adjust the velocity suitably of each particle by using following function

$$\Delta X_{di} = \left| (X_{di} - X^{best}) \right| \quad (12)$$

$$FD_d = \text{Max}(\Delta X_{di}) \quad (13)$$

where  $X_{di}$  is the position of  $i^{th}$  particle, and  $FD_d$  is the furthest distance from the particle to  $X^{best}$ .

This algorithm follows the same steps as PSO but a little change at step 4. In this algorithm velocity is updated according to the formula given as

$$ac = \text{rand} * 0.5 \quad (14)$$

$$V_i^{n+1} = \begin{cases} V_i^n \left( 1 + ac_d * \frac{\text{Iter}_{\max} - \text{Iter}}{\text{Iter}_{\max}} \right) & \text{if } \frac{\Delta X_{di}}{FD_d} > 0.5 + ac_d \\ V_i^n \left( 1 - ac_d * \frac{\text{Iter}_{\max} - \text{Iter}}{\text{Iter}_{\max}} \right) & \text{if } \frac{\Delta X_{di}}{FD_d} < 0.5 - ac_d \\ V_i^n & \text{if } 0.5 - ac_d \leq \frac{\Delta X_{di}}{FD_d} \leq 0.5 + ac_d \end{cases}$$

Then this new velocity value put into (11) and updates the position. Finally it gives the desired filter coefficients.

### 3.3. Particle Swarm Optimization with Variable Acceleration Factor (PSO-VAF)

A novel PSO with variable acceleration factor (PSO-VAF) has been proposed [21]. In this algorithm a modification has been made with  $C_1$  and  $C_2$ , acceleration factors appears in (10).  $C_1$  has been allowed to decrease from its initial value of  $C_{1i}$  to  $C_{1f}$  while  $C_2$  has been increased from  $C_{2i}$  to  $C_{2f}$  using the following equation

$$C_{1t} = (C_{1f} - C_{1i}) \frac{\text{Iter}}{\text{Iter}_{\max}} + C_{1i} \quad (16)$$

$$C_{2t} = (C_{2f} - C_{2i}) \frac{\text{Iter}}{\text{Iter}_{\max}} + C_{2i} \quad (17)$$

Then in this case velocity update formula is given by

$$V_i^{n+1} = \omega^n V_i^n + C_{1t} * \text{rand}_1 * (X_i^{best} - X_i) + C_{2t} * \text{rand}_2 * (X^{best} - X_i)$$

The rest of the algorithm follows the same steps as given in PSO.

## 4. Design Examples and Discussion

In this section, we demonstrate the use of the PSO, DAPSO and PSO-VAF to design a 20 order low pass FIR filter. All the simulation results shown in this paper, has been made on MATLAB 7.10. The specifications of the filter to be designed using these algorithms are:  $\delta_p = 0.1$ ,  $\delta_s = 0.01$ ,  $\omega_p = 0.45$  and  $\omega_s = 0.55$ . **Table 1** shows the parameters and its corresponding value consider during this work.

**Figure 1** shows the frequency response in dB of the LP FIR filter designed using PSO, DAPSO, and PSO-

VAF in the case of fitness1. On the same plot, the same filter designed using the PM is also shown. The main purpose behind using the fitness1 is to find the possible minimum ripples in pass band and stop band in case of PSO, DAPSO and PSO-VAF. **Figure 2** shows the frequency response in dB of the LP FIR filter in the case of fitness2. Fitness2 is able to control the ripples in both bands separately. **Table 2** and **Table 3** show the optimized coefficients of LP FIR using fitness1 and fitness2, respectively. **Figure 3** shows the normalized plot of LP FIR using fitness2. **Figure 4** and **Figure 5** shows the normalized plot of pass band and stop band ripples, respectively in case of fitness2. The simulation results for fitness2 shows that PSO gives 30.90dB stop band attenuation and DAPSO gives 32.32dB stop band attenuation. PSO-VAF results in 33.07dB stop band attenuation. Hence result quality is improved as indicated in **Table 4**.

#### 4.1. Convergence of PSO, DAPSO and PSO-VAF

**Figures 6-8** shows the convergence behavior of PSO, DAPSO and PSO-VAF respectively. These plots provide the error fitness value of the algorithms with number of iterations. PSO converges to the minimum error fitness value of 0.9392 in 55.39 sec. DAPSO converges to the minimum error fitness value of 1.129 in 35.92 sec. PSO-VAF converges to very low error fitness value of 0.0581 in 17.38 sec. PSO-VAF converges very fast as compare to the PSO and DAPSO in finding the desired filter coefficients.

**Table 1.** PSO, DAPSO, PSO-VAF Parameters.

Parameter	PSO	DAPSO	PSO-VAF
Swarm Size	75	55	75
No. of Iteration	500	450	250
$C_1$	2.05	2.05	-
$C_2$	2.05	2.05	-
$\omega_{max}$	0.95	0.95	0.70
$\omega_{min}$	0.40	0.40	0.40
$V_{max}$	1	1	1
ac	-	rand() * 0.5	-
$C_{1i}$	-	-	2.5
$C_{1f}$	-	-	0.5
$C_{2i}$	-	-	2.5
$C_{2f}$	-	-	0.5

**Table 2.** Optimized Coefficient of LP FIR using Fitness1.

h(n)	PM	PSO	DAPSO	PSO-VAF
h(1) = h(21)	0.000016462026203	0.014155467145005	0.010253190363910	0.010282242559259
h(2) = h(20)	0.048051046361716	0.035708201937220	0.040649756828294	0.039332258313641
h(3) = h(19)	-0.000023455414888	-0.002831976074236	-0.008927064743925	-0.011868739559069
h(4) = h(18)	-0.036911143268907	-0.050270040134091	-0.041762390230176	-0.045670923049427
h(5) = h(17)	-0.000014804257488	0.005760792155862	0.001378646244629	-0.003499650522683
h(6) = h(16)	0.057262893095235	0.051661980462882	0.056345040911989	0.052157436660473
h(7) = h(15)	0.000000677226645	0.001362475956305	0.006223456703284	-0.005234338320384
h(8) = h(14)	-0.102172983403192	-0.100010586879689	-0.105707282071110	-0.105116257668940
h(9) = h(13)	0.000011850968750	0.011045373132673	-0.003197815420580	0.004056000199106
h(10) = h(12)	0.316962289494363	0.318950770012236	0.309669414878306	0.316249577468652
h(11)	0.500018538901555	0.500647700995791	0.486849643357095	0.496612235589394

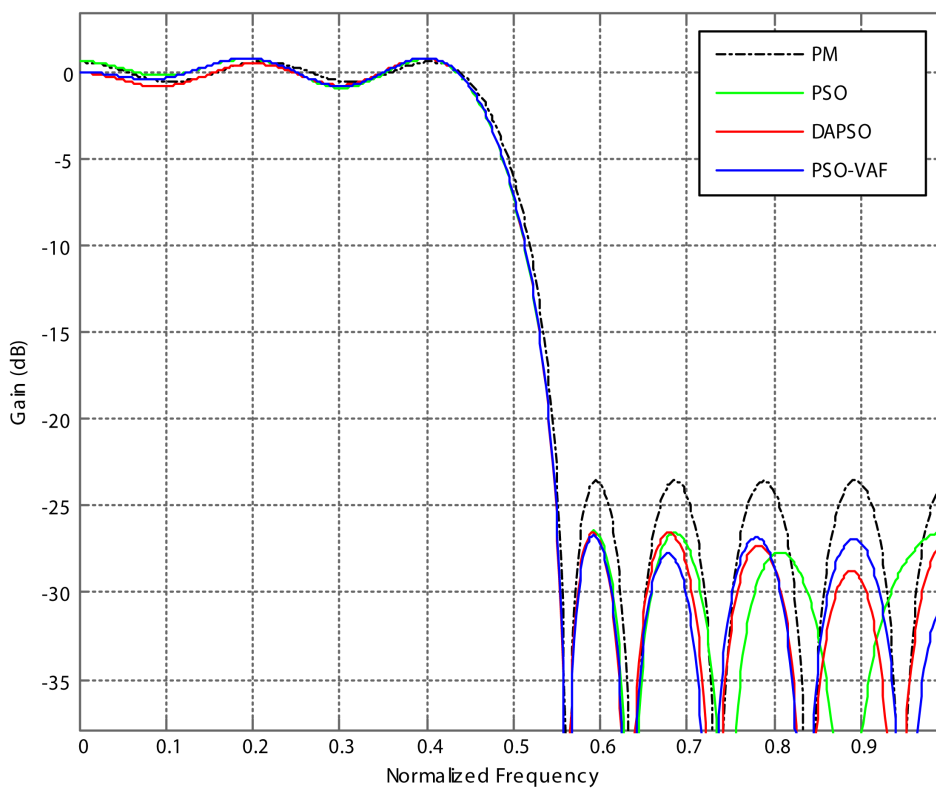


Figure 1. Frequency Response of LP FIR using Fitness 1.

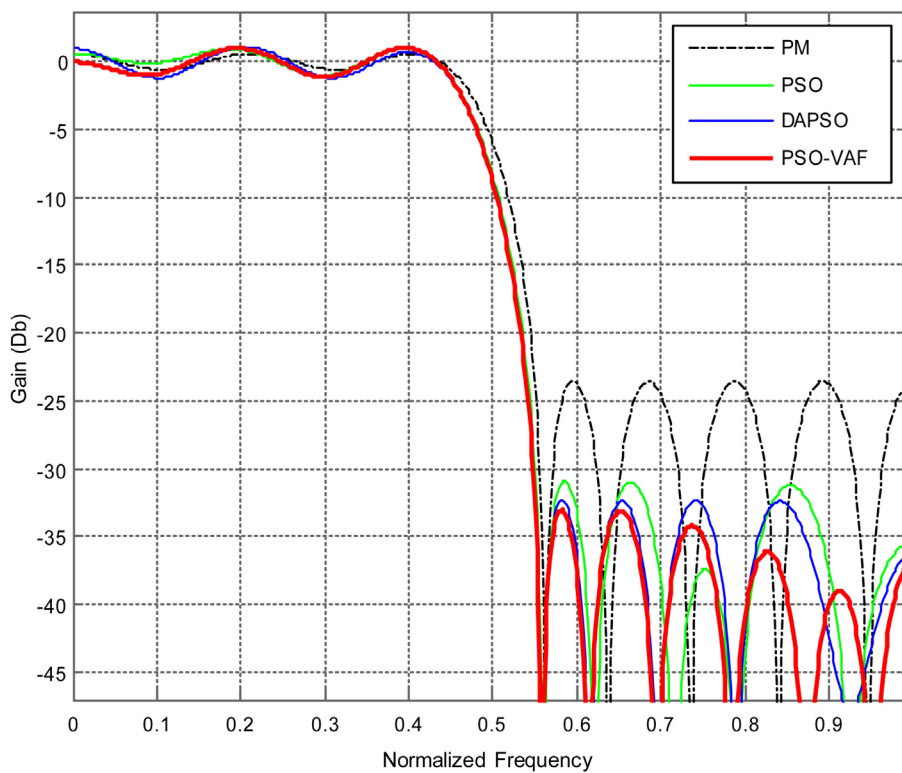


Figure 2. Frequency Response of LP FIR using Fitness 2.

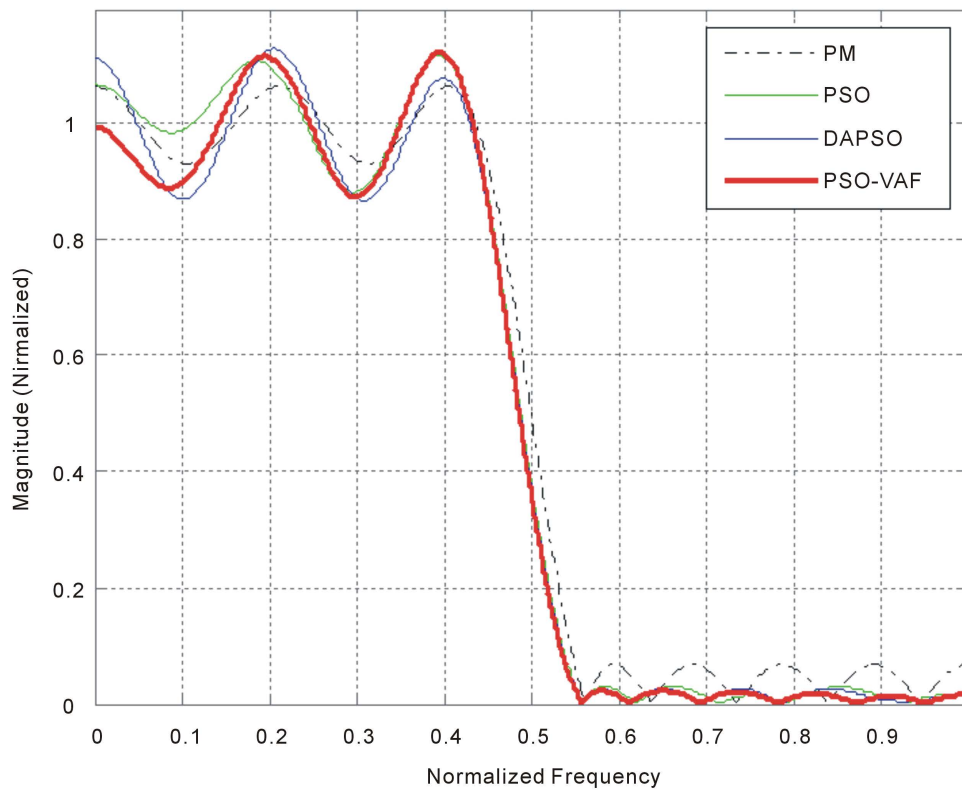


Figure 3. Normalized Plot of LP FIR using fitness 2.

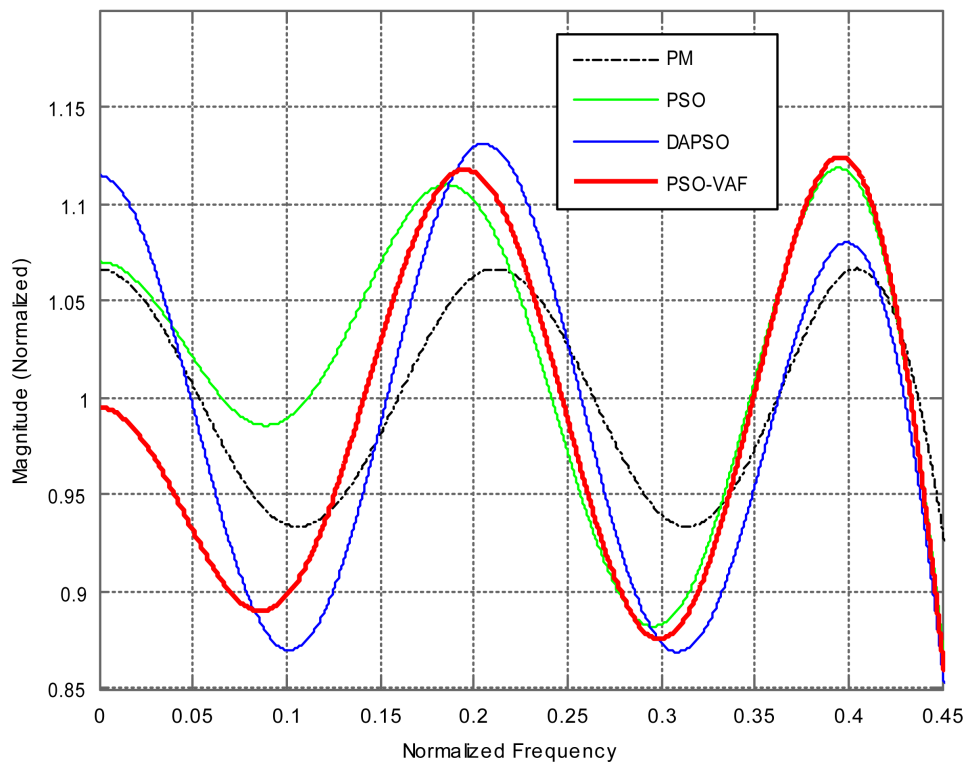


Figure 4. Normalized Pass Band ripples using fitness 2.

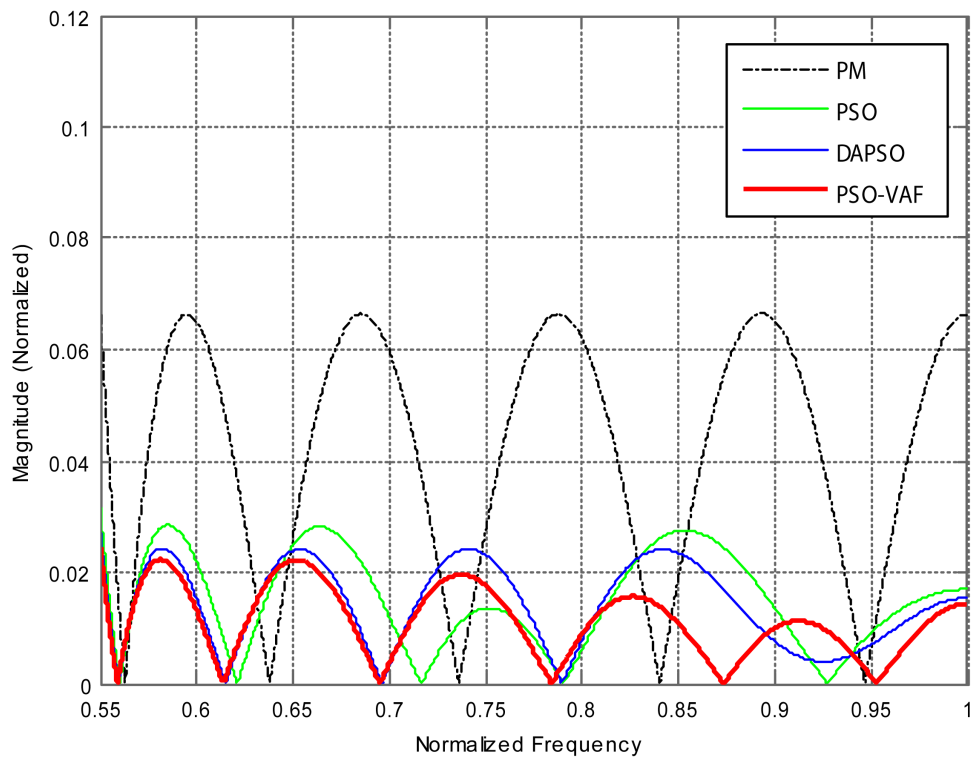


Figure 5. Normalized Stop Band ripples using fitness 2.

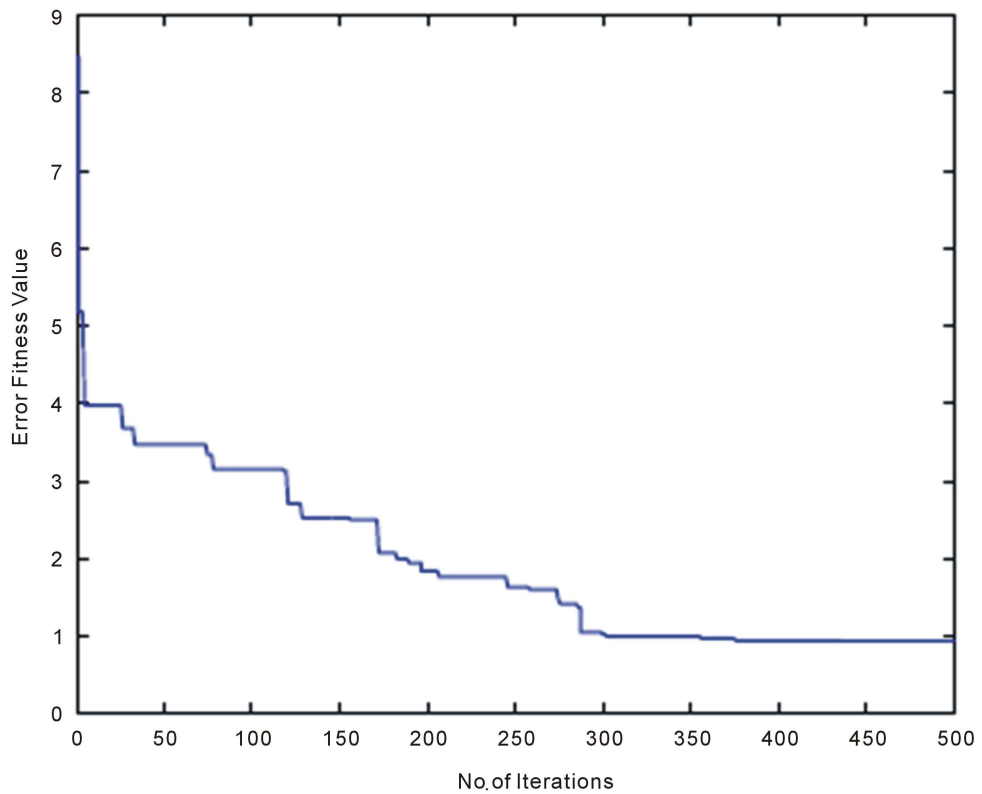


Figure 6. Convergence plot for PSO using fitness2.



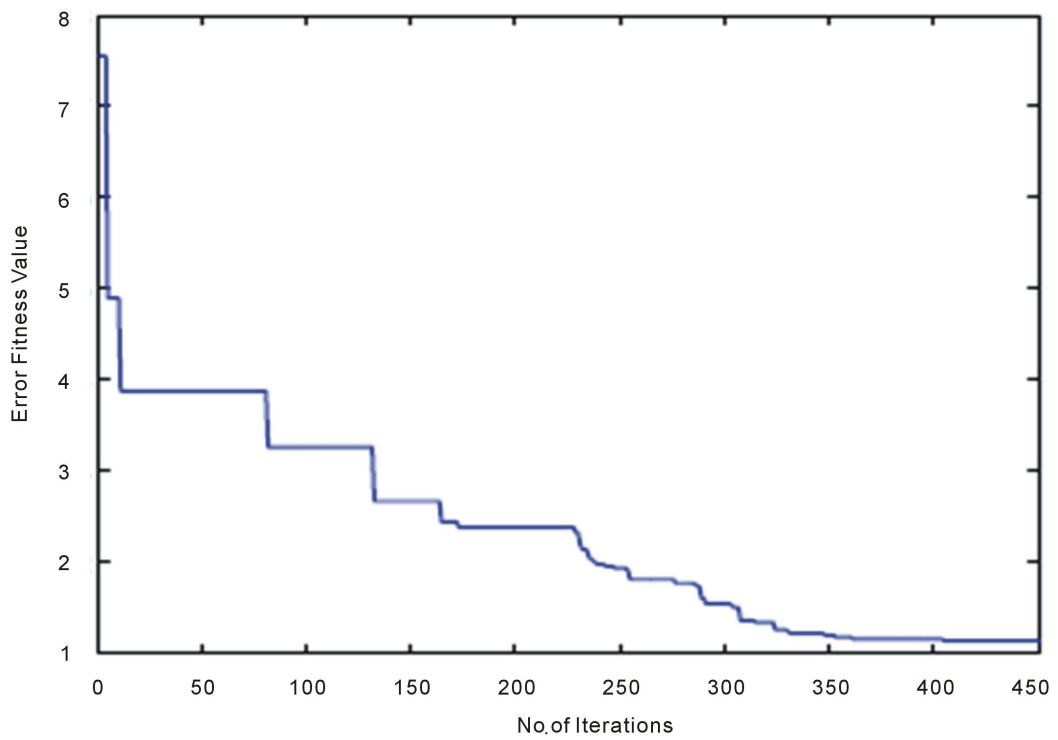


Figure 7. Convergence plot for DAPSO using fitness 2.

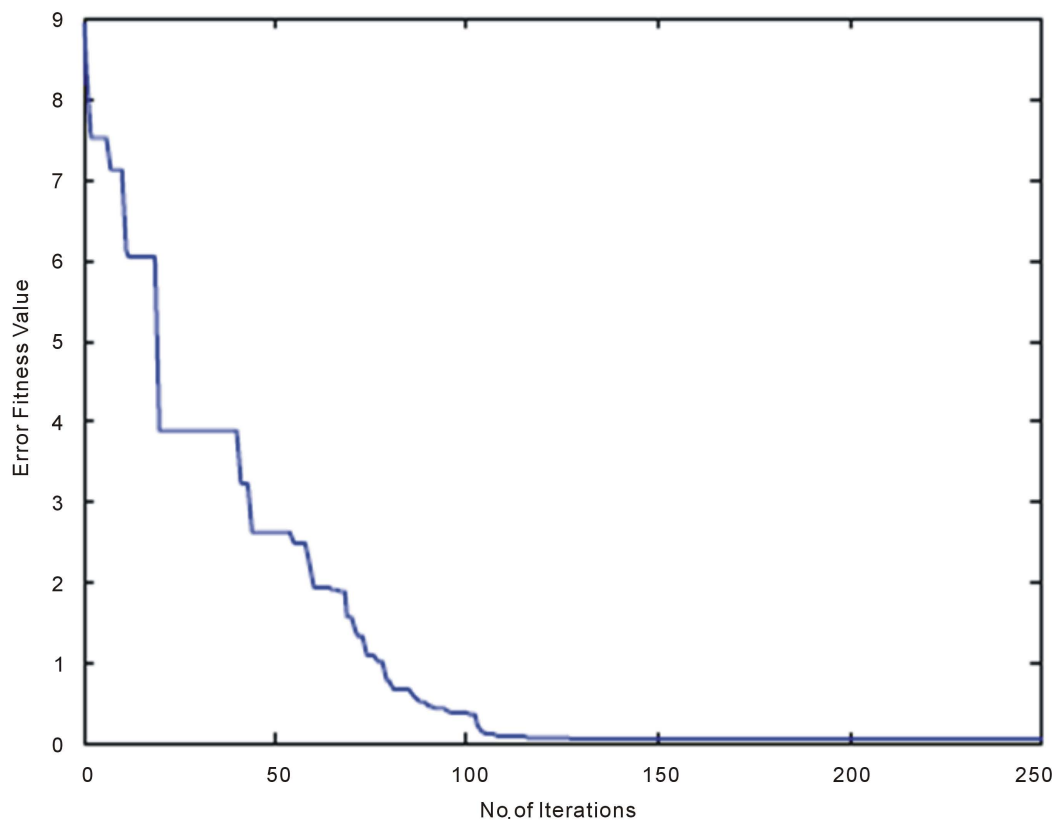


Figure 8. Convergence plot for PSO-VAF using fitness2.

**Table 3.** Optimized Coefficient of LP FIR using Fitness 2.

h(n)	PM	PSO	DAPSO	PSO-VAF
h(1) = h(21)	0.000016462026203	0.022587593788748	0.032377230675938	0.031004395567887
h(2) = h(20)	0.048051046361716	0.034243459454764	0.045898218784143	0.035783460880827
h(3) = h(19)	-0.000023455414888	-0.017086577157864	-0.001610645182605	-0.013335903393145
h(4) = h(18)	-0.036911143268907	-0.046131029539532	-0.035541394360043	-0.046776092048069
h(5) = h(17)	-0.000014804257488	0.000229126085020	0.003739660599334	0.000789458882427
h(6) = h(16)	0.057262893095235	0.058644950198106	0.052387868501172	0.051862590888302
h(7) = h(15)	0.000000677226645	-0.006546001812412	-0.008870863459771	-0.013369533411985
h(8) = h(14)	-0.102172983403192	-0.097122365891821	-0.101043272077621	-0.107381433337863
h(9) = h(13)	0.000011850968750	0.013760466527049	0.012605696966625	0.004297114513232
h(10) = h(12)	0.316962289494363	0.322156879042563	0.313121199216998	0.311753055773516
h(11)	0.500018538901555	0.500748972249363	0.488555464313817	0.486002781290836

**Table 4.** Results statistics.

Fitness	Method	FIR LP filter of Order 20									
		Pass band ripple $\delta_p$ (normalized)					Stop band ripple $\delta_s$ (normalized)				
		Avg.	Min.	Max	Var.	Std.	Avg.	Min.	Max.	Var.	Std.
Fitness1	PM	0.066	0.066	0.066	0	0	0.066	0.066	0.066	0	0
	PSO	0.0747	0.028	0.1018	0.00080	0.02831	0.04448	0.04128	0.04761	0.00001	0.00330
	DAPSO	0.0815	0.0529	0.0941	0.00037	0.0193	0.0432	0.0364	0.0468	0.00002	0.0048
	PSO-VAF	0.0819	0.0504	0.0925	0.00044	0.0210	0.0442	0.0409	0.0457	0.000005	0.0022
Fitness2	PM	0.066	0.066	0.066	0	0	0.066	0.066	0.066	0	0
	PSO	0.1041	0.0702	0.1184	0.00052	0.0229	0.0189	0.0095	0.0285	0.000074	0.0087
	DAPSO	0.1169	0.0800	0.1310	0.00048	0.02202	0.0193	0.0152	0.0242	0.000069	0.0083
	PSO-VAF	0.1190	0.1098	0.1240	0.000045	0.0067	0.0182	0.01127	0.02227	0.000022	0.004723

## 5. Conclusion

In this paper, the application of PSO and its two new variants, DAPSO and PSO-VAF, to design linear phase low pass FIR filter, have been carried out. Comparison of results of PM, PSO, DAPSO and PSO-VAF has been made. It is analyzed that the performance of DAPSO is better than PSO in context of stop band attenuation. On the other hand, PSO-VAF provides better stop band attenuation as compared to both, PSO and DAPSO. The convergence behavior of the algorithms is also compared and it is examined that PSO-VAF quickly converge with very low error fitness value. Two fitness functions are considered. The obtained results show that fitness2 is the better choice for designing optimal digital low pass FIR filter.

## References

- [1] Sheno, B.A. (2006) Introduction to Digital Signal Processing and Filter Design. John Wiley & Sons, Inc., Hoboken.
- [2] Oppenheim, A.V., Schaffer, R.W. and Buck, J.R. (1999) Discrete-Time Signal Processing. Prentice Hall, Englewood Cliffs.
- [3] Proakis, J.G. and Manolakis, D.G. (2007) Digital Signal Processing: Principles, Algorithms, and Applications. 4th Edition, Pearson Education, Inc., New Delhi.
- [4] Parks, T.W. and McClellan, J.H. (1972) A Program for the Design of Linear Phase Finite Impulse Response Filters. *IEEE Transactions on Audio and Electroacoustics*, **20**, 195-199. <http://dx.doi.org/10.1109/TAU.1972.1162381>
- [5] Benvenuto, N., Marchesi, M. and Uncini, A. (1992) Applications of Simulated Annealing for the Design of Special Digital Filters. *IEEE Transactions on Signal Processing*, **40**, 323-332. <http://dx.doi.org/10.1109/78.124942>
- [6] Traferro, S., Capparelli, F., Piazza, F. and Uncini, A. (1999) Efficient Allocation of Power of Two Terms in FIR Digital Filter Design Using Tabu Search. *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, Orlando, 30 May 1999-2 Jun 1999, 411-414.

- [7] Nurhan, K. and Cetinkaya, B. (2006) Design of Digital FIR Filters Using Differential Evolution Algorithm. *Circuits, Systems and Signal Processing*, **25**, 649-660. <http://dx.doi.org/10.1007/s00034-005-0721-7>
- [8] Nurhan, K. (2009) A New Design Method Based on Artificial Bee Colony Algorithm for Digital IIR Filters. *Journal of the Franklin Institute*, **346**, 328-348. <http://dx.doi.org/10.1016/j.jfranklin.2008.11.003>
- [9] Mondal, S., Ghoshal, S.P., Kar, R. and Mandal, D. (2012) Novel Particle Swarm Optimization for Low Pass FIR Filter Design. 2012 *IEEE Symposium on Humanities, Science and Engineering Research (SHUSER)*, Kuala Lumpur, 24-27 June 2012, 413-418.
- [10] Mukherjee, S., Kar, R., Mandal, D., Mondal, S. and Ghoshal, S.P. (2011) Linear Phase Low Pass FIR Filter Design Using Improved Particle Swarm Optimization. 2011 *IEEE Student Conference on Research and Development (SCORED)*, Cyberjaya, 19-20 December 2011, 358-363.
- [11] Archana, S., Mahapatra, R.K. and Panigrahi, S.P. (2011) DEPSO and PSO-QI in Digital Filter Design. *Expert Systems with Applications*, **38**, 10966-10973. <http://dx.doi.org/10.1016/j.eswa.2011.02.140>
- [12] Zhao, Z.K. and Gao, H.Y. (2009) FIR Digital Filters Based on Cultural Particle Swarm Optimization. *Proceedings of 2009 International Workshop on Information Security and Application (IWISA 2009)*, Qingdao, 21-22 November 2009, 252-255.
- [13] Meisam, N. and Ayatollahi, A. (2008) A Comparison between Genetic Algorithm and PSO for Linear Phase FIR Digital Filter Design. *9th International Conference on Signal Processing*, Beijing, 26-29 October 2008, 2134-2137.
- [14] Luitel, B. and Venayagamoorthy, G.K. (2008) Particle Swarm Optimization with Quantum Infusion for the Design of Digital Filters. *IEEE Swarm Intelligence Symposium*, St. Louis, 21-23 September 2008, 1-8.
- [15] Parks, T.W. and McClellan, J.H. (1972) Chebyshev Approximation for Non Recursive Digital Filters with Linear Phase. *IEEE Transactions on Circuits Theory*, **19**, 189-194. <http://dx.doi.org/10.1109/TCT.1972.1083419>
- [16] Ababneh, J.I. and Bataineh, M.H. (2008) Linear Phase FIR Filter Design Using Particle Swarm Optimization and Genetic Algorithms. *Digital Signal Processing*, **18**, 657-668. <http://dx.doi.org/10.1016/j.dsp.2007.05.011>
- [17] Kennedy, J. and Eberhart, R.C. (1995) Particle Swarm Optimization. *Proceedings of IEEE International Conference on Neural Networks*, Perth, 27 November 1995-1 December 1995, 1942-1948. <http://dx.doi.org/10.1109/ICNN.1995.488968>
- [18] Kennedy, J. and Eberhart, R.C. (2001) *Swarm Intelligence*. Morgan Kaufmann, San Francisco.
- [19] Robinson, J. and Rahmat-Samii, Y. (2004) Particle Swarm Optimization in Electromagnetics. *IEEE Transactions on Antennas and Propagation*, **52**, 397-402. <http://dx.doi.org/10.1109/TAP.2004.823969>
- [20] Liao, C.-Y., Lee, W.-P., Chen, X.H. and Chiang, C.-W. (2007) Dynamic and Adjustable Particle Swarm Optimization. *Proceedings of the 8th WSEAS International Conference on Evolutionary Computing*, Vancouver, 23-25 September 2008, 301-306.
- [21] Tripathi, P.K., Bandyopadhyay, S. and Kumar Pal, S. (2007) Multi Objective Particle Swarm Optimization with Time Variant Inertia and Acceleration Coefficients. *Information Sciences*, **177**, 5033-5049. <http://dx.doi.org/10.1016/j.ins.2007.06.018>.