Optimal Linear Phase FIR Filter Design using Bare Bones Particle Swarm Optimization (BBPSO) with Ripple Constraint

Inderjeet Singh
Department of Electrical and Instrumentation Engineering
Thapar University, Patiala, Punjab

Nirbhowjap Singh
Department of Electrical and Instrumentation Engineering
Thapar University, Patiala, Punjab

Abstract— The utilization of optimization techniques for designing digital filters has become widespread over the recent years. FIR filter optimization is a multi-model problem. The objective of the optimization is to derive an optimum set of filter coefficients for the given specifications. The optimization problem is phrased as least squared error (LSE) between the specified and designed filter response in the frequency domain. This paper presents an alternative approach for the designing of linear phase Finite impulse response (FIR) high pass filter by using Bare Bone particle swarm optimization (BBPSO) technique incorporated with ripple constraint in order to achieve flatter frequency response. It is expedient to use BBPSO for optimization because it do not require any parameter tuning as in the case of other PSO variants. The results justify that proposed BBPSO based technique vanquish the other PSO variants in term of accuracy, robustness, convergence speed and stability.

Keywords—Particle swarm optimization, Bare bones Particle swarm optimization

I. INTRODUCTION

The digital FIR filters are the basic building blocks in a variety of Digital Signal Processing (DSP) system. A digital FIR filter is basically a discrete time and discrete amplitude convolver [1]. The major advantage of FIR filters that they are simple to design and guarantee bounded input-bounded output (BIBO) stability and linear phase [2].They are widely accepted because of their flexibility in implementation, precision in performance, adaptability, linearity in phase and reproducibility [3]. The basic operations performed by FIR filter are noise cancellation, signal restoration and signal manipulation. The application area of digital FIR filters is very vast. Digital FIR filters can be used for both real-time and off-line (recorded signal) applications. Audio and video signal processing, communication, biomedical instrumentation are few of them. Processing of biomedical signals such as EEG, ECG, EMG and MRI also require digital filtration [4]. The frequency response of any digital filter depends on the numerical value of the filter coefficients. These values of the filter coefficients are typically represented by floating point numbers and the accuracy of the frequency response depends on the precision of the floating point numbers. Digital filters can be implemented on both hardware and software. The hardware implementation requires an optimum set of filter coefficients due to the limited size of registers, adders, multiplexers, buses and other hardware resources. The oversized filter ultimately results in higher power consumption and sluggish response.

Various methods based on both linear programming and non-linear techniques have been used for designing FIR filters. Window techniques, frequency sampling, equiripple design techniques and weighted least square error are some of the conventional approaches. Frequency sampling approach involves sampling the frequency response at equispaced points by using discrete Fourier transform (DFT) and then computing the filter coefficients by the mean of inverse discrete Fourier transform (IDFT). In window techniques the infinite impulse response of ideal filter is delayed and truncated. Various types of windows such Rectangular, Hamming and Henning are used in to reduce oscillatory behavior (Gibbs phenomena). Frequency sampling approach provides better control over the critical frequencies as com-pared to window approach [5]. The conventional gradient based methods lack in ability to handle multi-objectives and constraints in a single problem [3]. This fact has attracted re-searchers towards the use various non-linear meta-heuristic optimization techniques for designing optimum filters.

The meta-heuristic optimization techniques such as Genetic algorithm (GA), Differential evolution (DE), Particle swarm optimization (PSO) and artificial bee colony (ABC) have been used in the literature by researchers to handle multi-objective, multi-dimensional and constrained optimization problems. These techniques are based on phenomena of natural selection, evolution and swarm intelligence [6]–[10]. For designing FIR filter different optimization techniques have been reported in the literature such as DE [11], GA [12], ABC [13] and PSO. PSO [14] was developed by Kennedy and Eberhart and it simulates the social behavior of fish schooling, bird flocking and swarm of bees. Its ability to handle multi-model problems with minimum number of control parameters, computational efficiency and ease in implementation in context of both coding and parameter tuning makes it prestigious choice towards the designing of FIR filters [15]. Because of this reason it is one of the most explored algorithms used for FIR filter designing. Despite of these advantages PSO also has some unresolved issues like premature convergence due to homogeneity of particles , larger computational time as compared to other mathematical approaches, dependency on particle initialization and

parameter initialization, inefficiency of mathematical background and stagnancy of particles in the swarm. All these issues highly influence the performance of the algorithm and results in premature convergence ultimately provide sub-optimal solution [16]. The loss of diversity is also a major issue related the premature convergence. Different modifications in the canonical PSO algorithm have been presented to overcome the limitations. An attempt has been by using different communication topologies for enhanced information sharing within the swarm [17] in order to maintain the diversity of population. Other approaches utilize modifications in the basic velocity equation by adding some inertial weight and constriction factor terms [16], [18]. Hybridization of PSO algorithm with other techniques has also been listed in the literature [19]. All these modified variants assure better performance and convergence profile but at the same instance induce an additional complexity in the standard algorithm.

The above context reveals that the performance of PSO algorithm is highly influenced by the control parameters used by the algorithm itself. BBPSO algorithm takes the advantage over the other algorithm by minimizing the parameter requirement and eliminates the velocity formula by directly updating the position by sampling the search space using Gaussian distribution [20].

The paper is organized in 5 sections. In the succeeding section FIR high pass filter problem formulation part and constraint handling procedure have been present. Section 3 introduces the proposed BBPSO algorithm followed by comparison of results with other PSO variants is presented in section 4. Finally, section 5 concludes the paper

II. PROBLEM FORMULATION

A digital FIR filter in Z domain is characterized by

$$H(Z) = \sum_{n=0}^{N} h(n)Z^{-n}; \qquad n = 0,1,\cdots,N$$

(1)

Where $h(n)$ is the finite impulse response and N is the order of the filter having length $(N+1)$ Depending on the length and type of symmetry FIR filter are categorized in four types namely Type 1, Type 2, Type 3 and Type 4. This study uses type 1 filter having odd length and satisfies even symmetry i.e. $h(n) = h(N - n)$ for $0 \leq n \leq N$. Type 1 filters are preferred over the other filter due to their versatility [5] in the usage. The property of symmetry also reduces the computational task by a factor of 2. This means that instead of N only $\frac{N}{2} + 1$ filter coefficient are requires to be optimized. The rest of coefficients can be found by using the symmetry property. In order to satisfy frequency domain specifications the filter needs to be represented in the frequency domain. The normalized frequency response of the FIR digital filter can be estimated by taking the Discrete Fourier Transform (DFT) of the impulse response at finite number of frequency samples in $[0, \pi]$ interval as:

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

(1)

Where $\omega_k = \dfrac{2\pi k}{n}$ is the k_{th} frequency sample frequency sample and $H(e^{j\omega_k})$ is the Fourier transform complex vector.

The very first step in designing an optimum filter is to characterize the filter design problem as an optimization problem with an objective to minimize an error function. The deviation in frequency response of designed filter from the desired response is treated as an error function by many researchers. Further various specifications can also be incorporated with LSE to form a constrained optimization problem. Maximum ripple magnitude is phrased as a constraint to the FIR design problem. The non-linear constrained optimization problem can be solved by using either direct or indirect constraint handling methods. Indirect methods explicitly convert a constrained problem to an unconstrained problem and are easier to implement. The LSE and the maximum ripple constraints have been formulated in the following sub-sections.

A. Least Squared Error

The objective function is mathematically formulated as least squared error between the ideal filter frequency response and the designed filter frequency response give as:

$$L(\omega) = min \mid H_i(e^{j\omega_k}) - H_d(e^{j\omega_k}) \mid^2$$

(3)

where, $L(\omega)$ is the error sequence difference, H_i represents the magnitude response of the ideal filter and is given as:

$$H_i(e^{j\omega_k}) = \begin{cases} 1 & \text{for } \omega \geq \omega_c \\ 0 & \text{otherwise} \end{cases}$$

B. Ripple Constraint Handling

A traditional penalty function based constraint handling technique has been used to handle the maximum ripple constraint. The penalty function admonishes the premature convergence by adding some penalty to the objective function whenever any constraint is violated [21].

$$\delta(\omega) = \begin{cases} max(\mid H_d(e^{j\omega_k}) \mid) & \text{for } 0 \leq \omega \leq \omega_c \\ max(\mid 1 - H_d(e^{j\omega_k}) \mid) & \text{for } \omega \geq \omega_c \end{cases}$$

The objective function LSE is modified by incorporating maximum ripple constraint $\delta(\omega)$ to formulate constrained optimization problem.

$$E(\omega) = c_L * L(\omega) + c_R \delta(\omega)$$

(2)

where c_L and c_R are suitable weight parameters.

III. BBPSO ALORITHM

BBPSO was developed by Kennedy as a Gaussian variant of the standard PSO. An implementation of this technique is provided in Algorithm 1. It also eliminates the requirement of velocity equation by directly estimating the position. This study not only presents the aspect of center of oscillation but also the variability in the deviation from the center. This algorithm uses Gaussian distribution to sample the search space instead of uniform distribution used by other PSO variants. On the other hand BBPSO also decimate the use of

various control parameters such as inertial weights and constriction factors. Thus it seems to be a parameter free algorithm which updates the positions of particles given by:

$$X_{ij}^{t+1} = N(\mu, \sigma) \qquad (5)$$

where N a set of Gaussian is distributed numbers centered around the mean (μ) of *Globle* and *local best* values and standard deviation (σ) given as the absolute difference between the two values.

$$\mu = 0.5(X_{gbest_j} + X_{lbest_{ij}}); \qquad (3)$$
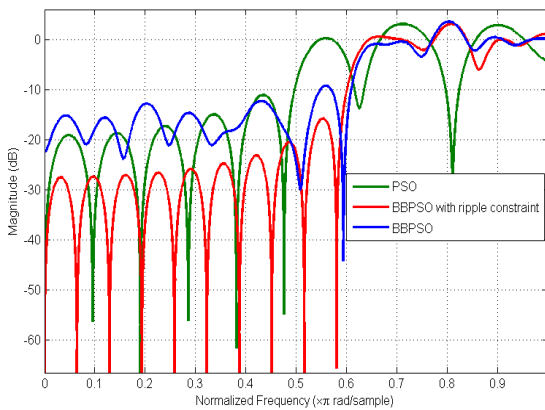
$$\sigma = | X_{gbest_j} - X_{lbestij} | \qquad (4)$$

Fig.1: Magnitude response for 20 order filter in dB

Algorithm 1 Psudocode for BBPSO algorithm

Input: swarm size P, Dimensions D, maximum bound x_{max}, minimum bound x_{min}
Initialize each particle X_{ij} using uniform probability distribution in $[x_{min}, x_{max}]$
for $i = 1 \rightarrow P$ **do**
  for $j = 1 \rightarrow D$ **do**
    $X_{ij} = [x_{min} + (x_{max} - x_{min}) * U(0,1)$
    $X_{lbest_i} = X_i$
  end for
  Evaluate fitness $f(X_{ij})$ $X_{gbest} = min(X_{ij})$
end for
for $i = 1 \rightarrow i_{max}$ **do**
  Update the position X_i according to equation (7)
  if $f(X_i) < f(X_{lbest_i})$ **then**
    $X_i = X_{lbest_i}$
  end if
  if $f(X_i) < f(X_{gbest_i})$ **then**
    $X_i = X_{gbest_i}$
  end if
end for
Output: X_{gbest}

IV. RESULTS AND DISCUSSION

A. Magnitude response analysis for high pass fir filter

The implementation and simulation of high pass FIR filters is done in Matlab environment running on Intel(R) CORE (TM) i3 processor having clock speed 2.27 GHz. Two cases of high pass FIR filter having order 20 and 30 have been discussed in this study and filter length is 21 and 31

respectively . The sampling frequency is selected as 1 Hz and magnitude response analysis is done for 128 sampling points in the normalized frequency domain. Pass band (!s) and stop band cut off (!p) frequencies are chosen as 0:6 and 0:65 respectively. All of these parameters have been summarized in Table 1. The control parameters used by PSO and BBPSO algorithms are presented in Table 2.
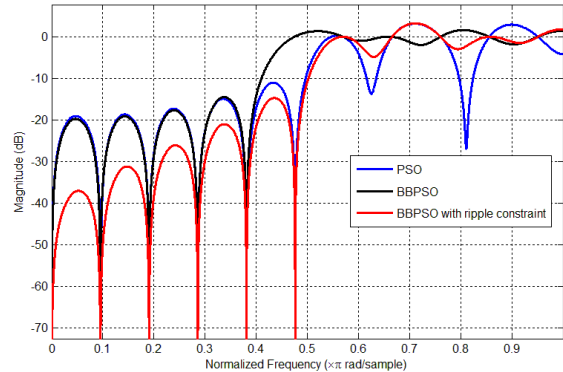

Fig.2: Magnitude response for 30 order filter in dB

Table-1: FIR filter parameters

Filter parameters	Value
sampling frequency f_s	$1Hz$
number of frequency samples	128
stop band(normalized) edge frequency ω_p	0.60
pass band(normalized) edge frequency ω_s	0.65

Table-2: PSO and BBPSO parameters

Parameter	BBPSO	PSO
inertia weight (W)	-	[0.4,0.9]
C_1	-	2
C_2	-	2
X_{min}	-2	-2
X_{max}	2	2
V_{min}	-	.01
V_{max}	-	1
swarm size (p_size)	25	25
number of iteration (n_itr)	500	500

Fig.1 and Fig.2 shows magnitude response plot of filters of order 20 and 30 respectively in dB scale. The best set of filter coefficients has been obtained by running 30 trials for each algorithm. Both the figure demonstrate that BBPSO algorithm produces better filter response in terms of ripple magnitude, flatness, and stop band attenuation for both 20 and 30 order. Finally the analysis of magnitude repose and various performance parameters manifest the better performance of BBPSO algorithm with minimum number of control parameter.

V. CONCLUSION AND FUTURE SCOPE

In this paper an efficient and parameter independent technique based on BBPSO algorithm with maximum ripple is

Special Issue - 2016

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
ACMEE - 2016 Conference Proceedings

discussed for designing optimum High pass FIR filter. From the simulation results it can be concluded that the frequency response of BBPSO filter with ripple constraint is comparable with the results derived using unconstrained BBPSO and other PSO variants. Thus because of its faster convergence and minimum dependency on control parameters, BBPSO optimization algorithm with ripple constraint can be utilized efficiently as an alternative of other PSO variants for FIR filter designing and optimization as an alternative of other PSO variants. In future more promising results can be obtained by increasing the number of trials and incorporating other filter specifications as constraints to the optimization problem.

REFERENCES

[1] S. Mandal, S. P. Ghshal, R. Kar, D. Mandal, and A. Shivare, "Swarm intelligence based optimal linear fir high pass filter design using particle swarm optimization with constriction factor and inertia weight approach," in IEEE Student Conference on Research and Development (SCOReD), pp. 352–357, Dec 2011.

[2] L. Litwin, "FIR and IIR digital filters," IEEE Potentials, vol. 19, no. 4, pp. 28–31, Oct 2000.

[3] A. Aggarwal, T. K. Rawat, and D. K. Upadhyay, "Design of op-timal digital fir filters using evolutionary and swarm optimization techniques," AEU-International Journal of Electronics and commumication 2015.

[4] T. R. Miller, R. J. Hagge, J. W. Wallis, and K. S. Sampathkumaran, "Interactive digital filtering of gated cardiac studies during cine display," IEEE Transactions on Medical Imaging, vol. 7, no. 3, pp. 188–192, Sep 1988.

[5] G. Proakis John and G. Manolakis Dimitris, "Digital signal processing: principles, algorithms, and applications," Pentice Hall, 1996.

[6] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm", Journal of global optimization, vol. 39, no. 3, pp. 459–471, 2007.

[7] X.-S. Yang, Nature-inspired optimization algorithms Elsevier.

[8] T. Back, D. B. Fogel, and Z. Michalewicz, "Handbook of evolutionary computation", IOP Publishing Ltd., 1997.

[9] X.-S. Yang, Z. Cui, R. Xiao, A. H. Gandomi, and M. Karamanoglu, "Swarm intelligence and bio-inspired computation: theory and applications", Newnes, 2013.

[10] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," Journal of global optimization, vol. 11, no. 4, pp. 341–359, 1997.

[11] S. Sharma, L. D. Arya, and S. Katiyal, "Design of linear-phase digital fir filter using differential evolution optimization with ripple constraint," in International Conference on Computing for Sustainable Global Development (INDIACom), pp. 474–480, March 2014.

[12] D. Suckley, "Genetic algorithm in the design of fir filters," in IEE Proceedings G Circuits, Devices and Systems, vol. 138, no. 2. IET, 1991, pp. 234–238.

[13] N. Karaboga, "A new design method based on artificial bee colony algorithm for digital IIR filters," Journal of the Franklin Institute, vol. 346, no. 4, pp. 328 – 348, 2009.

[14] J. Kenndy and R. Eberhart, "Particle swarm optimization", in Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948, 1995.

[15] J. I. Ababneh and M. H. Bataineh, "Linear phase fir filter design using particle swarm optimization and genetic algorithms," Digital Signal Processing, vol. 18, no. 4, pp. 657–668, 2008.

[16] J. Kennedy, "Some issues and practices for particle swarms," in IEEE Swarm Intelligence Symposium, SIS 2007, pp. 162–169, April 2007.

[17] T. Tsujimoto, T. Shindo, and K. Jin'no, "The neighborhood of canonical deterministic pso," in 2011 IEEE Congress on Evolutionary Computation (CEC), pp. 1811–1817, June 2011.

[18] R. C. Eberhart and Y. Shi, "Comparing inertia weights and constriction factors in particle swarm optimization," in Proceedings of the 2000 Congress on Evolutionary Computation, vol. 1, pp. 84–88, 2000.

[19] C.-F. Juang, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design," IEEE Transactions on Systems, Man, and Cybernetics-Part B (Cybernetics), vol. 34, no. 2, pp. 997–1006, April 2004.

[20] J. Kennedy, "Bare bones particle swarms," in Proceedings of the 2003 IEEE Swarm Intelligence Symposium, SIS '03, pp. 80–87, April 2003.

[21] G. Coath and S. K. Halgamuge, "A comparison of constraint-handling methods for the application of particle swarm optimization to constrained nonlinear optimization problems," in The 2003 Congress on Evolutionary Computation, CEC '03, vol. 4, pp 2419–2425, Dec 2003.