Linear Phase FIR Low Pass Filter Design Based on Firefly Algorithm

Moath Sababha, Mohamed Zohdy
Electrical and Computer Engineering, Oakland University, USA

Article Info

Article history:

Received Nov 16, 2017
Revised Jul 30, 2018
Accepted Aug 13, 2018

Keyword:

Convergence
Differential Evolution (DE)
Finite Impulse Filter (FIR)
Firefly algorithm
Low pass filter
Parks–McClellan (PM)
Particle Swarm Optimization (PSO)

ABSTRACT

In this paper, a linear phase Low Pass FIR filter is designed and proposed based on Firefly algorithm. We exploit the exploitation and exploration mechanism with a local search routine to improve the convergence and get higher speed computation. The optimum FIR filters are designed based on the Firefly method for which the finite word length is used to represent coefficients. Furthermore, Particle Swarm Optimization (PSO) and Differential Evolution algorithm (DE) will be used to show the solution. The results will be compared with PSO and DE methods. Firefly algorithm and Parks–McClellan (PM) algorithm are also compared in this paper thoroughly. The design goal is successfully achieved in all design examples using the Firefly algorithm. They are compared with that obtained by using the PSO and the DE algorithm. For the problem at hand, the simulation results show that the Firefly algorithm outperforms the PSO and DE methods in some of the presented design examples. It also performs well in a portion of the exhibited design examples particularly in speed and quality.

Corresponding Author:

Moath Sababha,
Electrical and Computer Engineering,
Oakland University, Rochester, MI 48309 USA.
Email: msababha@oakland.edu

1. INTRODUCTION

During processing of signal, unwanted parts of the signal are removed by use of a filter. This paper looks at the digital filter. It uses digital processors to carry out algorithm on different samples of the value of the signal. The digital filter brings arithmetical procedures on models of unmistakable time signals to diminish or enhance highlights of the signals. The get rid of numerous problems associated with additional principle class of electronic Filters such as the Analogue to Digital Converter (ADC). There are two classes of digital signals. That is, the finite impulse response on (FIR) and the infinite impulse response (IIR) signals. They are identified with the length of the impulse response. The FIR signal is an alluring choice because of its effortlessness in design and steadiness. By plotting the signal valves to be equivalent with the middle valve area, the FIR filters have a linear phase. FIR channels are recognized to contain numerous appealing attributes like certain consistency, the chance of exact linear phase feature at all frequencies and advanced execution as non-recursive developments [1]-[9].

Various strategies are employed for the arrangement of digital filters [10], [11]. The windowing strategy is the most preferred. In this strategy, the perfect impulse response is increased by being multiplied with a window's function. There are various assortments of types of window capacities (Butterworth, Chebyshev, Kaiser and so on.). The types depend on the requirements of ripples on the pass band and stop band. Also, they depend on the stop band reduction and the conversion width. These distinctive windows bound the infinite length drive response of ideal filters into a finite window to design a genuine response. In any case, windowing systems don't allow enough composition of the frequency response in the diverse frequency bands and other filter factors such as the progress width. The Remez-exchange algorithms (traditional techniques) introduced by Parks and McClellan PM does not permit express choice of the frequency response contained in the distinct frequency bands. Techniques based on the Remez exchange algorithm are the most primitive ones [12]. However, this algorithm does not give room for explicit selection of the top limit of the absolute ripple in the passband and stopband (δp, δs), rather one can just determine their proportion. Moreover, the PM gives drifting point coefficients which requires quantization if equipment execution is looked for. Based on these reasons, when designing linear phase FIR filters, some digital stochastic worldwide model algorithm such as the Differential Evolution (DE) algorithm, Particle Swarm Optimization (PSO), and Genetic Algorithm (GA) are used [13]-[15].

The significant limitation of the designing procedure is that the similar estimations of the amplitude mistake in the frequency bands are specific by means of the weighting capacity, and not by the divergences themselves. Along these lines, if there should arise an occurrence of scheming band filters with a known stop band uniqueness, filter length and cut-off recurrence, the arrangement must be iterated severally. Numerous copies have been extended for the (FIR) signal techniques and plan designs. This is an investigative area that aims at accomplishing more general and spearheading strategies that are competent to determine as well as advance new and compound designing techniques [16].

Different conventional techniques exist for digital FIR channel design. Out of those one is the designing of the filter using the digital algorithms. This technique has a stunning capacity. That is, it gives alternatives of utilization of various digital algorithms and indecency in configuration. It specifically relies upon execution of algorithms. At first this technique was used by Parks and McClellan by utilizing a straightforward iterative suburbanite program and is named as PM method for filter designing. This strategy was altered later on by supplanting the use of basic program with enhancement algorithms. At first hereditary algorithm was utilized for a wide range of channel design. This was trailed by utilization of different algorithms. As of late, crossover algorithm and adjusted algorithms has been created from fundamental sort of algorithm for changes in the signal designs. Employments of enhanced molecule swarm advancement, versatile development molecule swarm improvements for the filter designing validate the later trends [16]-[24].

Several approaches were presented to design linear phase FIR filters. For instance, simulated annealing and GA method were applied to design FIR filters with coefficients values expressed as a power of two. However, these approaches are computationally very expensive. The Firefly method is easy to implement, and its convergence is controlled via few parameters. However, other methods, such as Genetic and DE algorithms, involves search procedures using the population genetics and natural selection process. The purpose of this paper is to use the Firefly approach as an alternative method to these approaches due to its ease of implementation in both parameter selection and the context of coding. This paper proposes a linear phase Low Pass FIR filter based onan alternative approach called Firefly method that assures the robustness of the FIR design in terms of performance and computational complexity.

This paper is organized asfollows. In Section II, the system design and analysis of theproposed linear phase FIR low pass filter are presented. The Differential Evolution (DE) algorithm and the Particle Swarm algorithm (PSO) are discussed brieflyin Section III and Section IV, respectively. In Section V, The Firefly algorithm is presented. Simulation and Examples of Design FIR using Firefly algorithm presented in Section VI. Finally, conclusions are outlined in Section VII.

2. THE WINDOWS FRAMEWORK OUTLINE AND EXAMINATION OF THE PROPOSED LINEAR PHASE FIR

Windowing strategy is used to get the simple type of FIR filters. FIR configuration begins with a perfect wanted frequency response obtained as follows:

$$H_d(e^{iw}) = \sum_{m=-\infty}^{\infty} h_d[n]e^{-i\omega m} \quad (1)$$

Where; $h_d[m]$ is the impulse response of the distinct channel. Then, we need to make $h_d[m]$ a causal FIR Filterby two stages. The first $h_d[m]$ is increased by a finite length agreement $d[m]$ (window) keeping in mind the end goal of getting a finite length impulse response. The second one which is causality is presented by time deferring the windowed drive response. One can express this in mathematical terms as follows:

$$h_w[m] = h_d[m]d[m] \quad (2)$$

Where $h_w[m]$denotes the windowed impulse response; so, the frequency response of the windowed impulse response $H_w(e^{iw})$ is the periodic convolution of the desired frequency response $H_d(e^{iw})$ with the Fourier transform$\Gamma(e^{iw})$ of the window and is given by

$$H_w(e^{iw}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{i\varphi})\Gamma(e^{i(w-\varphi)}).d\varphi \tag{3}$$

The frequency response of the approximation filter will be

$$H(e^{iw}) = e^{-iwm_d} H_w(e^{iw}) \tag{4}$$

Where m_d represents the necessary time delay to introduce causality of the approximated filter. Filters designed by windowing will have greatest error on either side of the discontinuity of the ideal frequency response and smaller error for frequencies away from the discontinuity. The frequency response $H(e^{iw})$ of an M length finite word-length FIR linear phase digital filter is given as:

$$H_d(e^{iw}) = \sum_{k=0}^{M-1} h[k]e^{-iwk} \tag{5}$$

Where $h[k]...\forall k = 0,1,2,....,M-1$ are b-bit (sign bit included) filter coefficients. Preferable channels over the windowing strategy result from the minimization of maximum error yields the most preferable filters over the windowing method. These kinds of filters can be gotten by utilizing algorithmic procedures. In an effort to outline FIR filters in which some of the parameters, designed algorithms are developed. Some of these parameters include; the channel length (M), passband and stopband standardized frequencies (w_p, w_s). Researchers have developed algorithms in which M, δ_p, and δ_s are fixed. The other parameters have been enhanced. Different estimates were initially set up by Parks and McClellan in which M, w_p, w_s, and the proportion ${}^{\delta_p}/_{\delta s}$ is fixed. From that point forward, the Parks– McClellan (PM) algorithm is the most famous approach for ideal FIR filters design. This is so because of its adaptability and computational productivity. An approximate error function in the PM algorithm is characterized by:

$$E(w) = G(w)\left[H_d(e^{iw}) - H(e^{iw})\right] \tag{6}$$

Where $H_d(e^{iw})$ is the frequency response of the desired filter and $H(e^{iw})$ isthe approximate filter, respectively, $G\,(w)$ is a weighing function. It is used to give the weighting of the approximation error distinctively in various recurrence groups of frequency bands. The objective of the design is to locate the approximate filter coefficients that are outcomes of the ideal filter. Since the maximum error is minimized, then the filter is optimum. In mathematical terms, the best approximation is to be found in the following sense:

$$\min_{\{h[n]:0\le n\le M\}}(\max_{w\in F} |E(w)|) \tag{7}$$

Where $h(n)$ is the impulse response of the approximate filter and F is the closed subset of $0 \le w \le 0.5$. This model does not permit express determination of the most extreme of the outright ripples in the passband and stopband (δ_p,δ_s), rather one can just indicate their proportion. Moreover, the PM gives drifting point coefficients which require quantization if hardware implementation is sought. This motivates utilizing a recently developed stochastic global optimization algorithm called Firefly algorithm to design linear phase FIR filter in this paper. The Firefly is an insect that mostly produces short and rhythmic flashes that produced by a process of bioluminescence. The function of the flashing light is to attract partners (communication) or attract potential prey and as a protective warning toward the predator. Thus, this intensity of light is the factor of the other fireflies to move toward the other firefly.

3. THE DIFFERENTIAL EVOLUTION (DE) ALGORITHM

DE algorithm is one of the powerful evolutionary algorithms to solve real parameter optimization problems. Several state-of-the-art evolutionaryalgorithms use DE as evolutionary mechanism [15]-[19]. The DE algorithm is a population based algorithm like genetic algorithms using the similar operators; crossover, mutation and selection. The major contrast in building better arrangements is that genetic estimates depend on combination while DE depends on conversion operation. This major operation depends on the distinctions of arbitrarily tested sets of methods in the populace. As a rule, DE algorithm has three stages [15]:

a. Mutation: At generation k for each parent vector u_k^i is generated according to following equation

$$u_k^i = x_k^{r1} + F(x_k^{r2} - x_k^{r3}) \tag{8}$$

Where $r_1 \neq r_2 \neq r_3 \neq i$ and F is referred as the scaling factor.

b. Crossover: This operator is essential due to its affectability to confrontdistinct and non-distinguishable problems. A trial child vectors $y_k^i = [y_{1k}^i, y_{2k}^i, ..., y_{nk}^i]$are developed by the hybrid of its parent vector x_k^i and its mutant vectoru_k^i as follows:

$$y_{jk}^i = \begin{cases} u_k^i & if \quad U_j \leq C_r \quad or \quad j = l \\ x_{jk}^i & if \quad U_j > C_r \quad and \quad j \neq l, \end{cases} \tag{9}$$

In the above model, *l* is anarbitrarilypicked integer from *{*1*;* 2*;...; d},U*1*;U*2*;... ;Ud*are arbitrary independent variables uniformly disseminated in [0*;* 1), and *C_r* € [0*;* 1] is an input parameter affecting the number of components to be exchanged by the crossover.

c. Selection (Determination): A greedy Selection employed where a child substitutes its parent in next generation if it has a superior or equivalent fitness function value.

In [19], they proposed the hybrid DE algorithm with local search techniques and an adaptive parameter value for C_r. This adaptive mechanism combines the binary crossover and line recombination. And, refreshment mechanism is used to avoid stagnation.

The major steps of the DE are [19]:

```
1. S1: Initialize P(0) with $N_p$ individual randomly selected from the searching space.
2. S2: Evaluate initial population P(0).
3. repeat
4. S3: For k=1 :MaxGen or Convergence Criterion reached
   a. Select randomly a subpopulation of $N_s$(k) individuals with proposed solutions on P(k).
   b. Apply the fundamental DE operator " Mutation and Cross over" to get the offspring O(k)
   c. For every child, perform a local search.
   d. Evaluate offspring O(k): if child $\leq$parent, the parent is substituted by its child
   e. If interquartile range (IQR) <$\hat{V}$ ; update the population.
5. until stopping condition;
6. Algorithm End
```

4. THE PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization (PSO) is an evolutionary algorithm presented and designed by Kennedy and Eberhart in 1995 [20]. Some efforts have been presented toward the advancement of the FIR Filter based on the PSO algorithm. The PSO easily applicable. Its merging might be overseen as using few variables. PSO is a versatile, incredible populace based stochastic pursuit or enhancement strategy with comprehended parallelism, which can, with no inconvenience, handle non-differential reason capacities. PSO is less vulnerable to get trapped on restricted optima dissimilar Genetic Algorithm (GA), replicated Annealing, etc. in [22-23], they expanded a similar PSO idea to present a swarm of birds. PSO is expanded throughout replication of bird flocking in multidimensional space. According to [24], flocking birds can be used to optimize a certain objective function. In PSO, every possible solution is denoted as a particle. Every particle is associated with a position x and a velocity v. one can express the position and velocity of the ith particle as following:

$$\vec{x}_i = (x_{i,1}, x_{i,2}, ..., x_{i,N}) \tag{10}$$

$$\vec{\vec{v}}_i = (v_{i,1}, v_{i,2}, ..., v_{i,N}) \tag{11}$$

The length of every vector N denotes the dimension of the problem or number of unknown variables. In each iteration, the cost function is processed for every one of particles in the swarm. This capacity ought to be precisely designed to give a reflection of the expected outcome. The position and velocity of each particle are updated according to individual best, xbesti, and global best, xbest, fitness as shown below:

$$v_i^{n+1} = \gamma^n v_i^n + c_1\beta_1^n(x_i^{best} - x_i) + c_2\beta_2^n(x^{best} - x_i),$$
$$X_i^{n+1} = x_i + \Delta t v_i^{n+1} \quad (12)$$

The superscripts n+1 and n indicate the time record of the current and the past cycles and β1 and β2 are arbitrary numbers that are consistently circulated in the interim (0, 1). These random numbers are refreshed each time they happen. The comparative weights of the personal best position corresponding to the global best position are determined by the parameters c1 and c2, respectively. Both c1 and c2 are typically set to a value of 2.0. The parameter γ n is the "inertia weight" in the nth iteration. It is a number in the range (0, 1) that selects the weight by which the particle's current velocity relies on its previous velocity, and the distance between the particle's position and its personal best and global best positions.

The population of particles is then moved according to equation (12) and tends to cluster together from different directions. Since, a maximum velocity, Vmax, should not increase with any particle to stay the search within a desired solution space. Assessment of the cost function is done utilizing the particle's new position. The algorithm goes through these procedures iteratively until the point that a particular end paradigm is met.

5. THE THE FIREFLY ALGORITHM

In [26], there is a clarification of how the Firefly algorithm that follows the firefly behavior. Firefly is an insect that for the most part delivers short and cadenced flashes that created by a procedure of bioluminescence. The capacity of the glimmering light is to pull in accomplices (correspondence) or draw in potential prey and as a defensive cautioning toward the predator. Along these lines, this power of light is the factor of other fireflies in advancing toward the other firefly.

The light power is changed at the separation from the eyes of the onlooker. It is protected to state that the light power is diminished as the distance increment. The light power likewise the impact of the air retains by the environment, in this way the force turns out to be less engaging as the separation increment. Firefly algorithm originallypresented based on three idealize rules, 1) Fireflies are attracted toward each other's regardless of gender. 2) The engaging quality of the fireflies is correlative with the splendor of the fireflies. Consequently, the less appealing firefly will push ahead to the more alluring firefly. 3) The shine of fireflies is relyingupon the cost function [27-30].

5.1. Structure of Firefly Algorithm

In firefly algorithm, there are two essential factors, which is the light intensity force and appeal. Firefly is pulled in toward the other firefly that has brighter blaze than itself. The engaging quality is depended with the light power. The light intensity accordingly attractiveness is inversely relative with the particular distance$\boldsymbol{r}$ from the light source. In this manner, the light and engaging qualityis diminishing as the distance increment. One can express it as follows:

$$I(r) = I_0 e^{-\gamma r^2} \quad (13)$$

where,

I = light intensity,
I_0 = light intensity at initial or original light intensity,
γ = the light absorption coefficient
r = distance between firefly *i* and *j*

Attractiveness is proportionally to the light intensity seen by another fireflies, thus attractiveness (β) is given by

$$\beta = \beta_0 e^{-\gamma r^2} \quad (14)$$

where β_0 represents the attractiveness at zero distance $(r = 0)$. The distance$r$ between two fireflies can be defined based on the Cartesian distance as follows:

$$r_{ij} = |x_i - x_j| = \sqrt{\sum_{k=1}^{d}(x_{i,k} - x_{j,k})^2} \tag{15}$$

Firefly i is attracted toward the more attractive firefly j, the updated movement is expressed as follows:

$$\Delta x_i = \beta_0 e^{-\gamma r_{ij}^2}(x_j^t - x_i^t) + \alpha\varepsilon_i, \qquad x_i^{t+1} + \Delta x_i \tag{16}$$

where β_0 is for attraction, γ is the limitation when the valuetends to zero or too large. If γ approaching zero $(\gamma \to 0)$, the attractiveness and brightness become constant, $\beta = \beta_0$. In another word, a firefly can be seen in any position, easy to complete global search. If the γ is nearing infinity or too large $(\gamma \to \infty)$, the attractiveness and brightness become decrease. The firefly movements become random. The implementation of firefly algorithm can be done in these two asymptotic behaviors. While the second the term is for randomization, as α is the randomizeparameter. The ε_i can be replaced by ran -1/2 which is ran is random number generated from 0 to 1. Please refer to Algorithm 1.

Algorithm 1:Firefly algorithm

```
Input: Cost Function $f(x)$, Initial population of Fireflies $\boldsymbol{x}_0$,
Define the light absorption coefficient $\gamma$, Max number of
Iteration $ITR$, initialize the Light Intensity $I_0$ at $\boldsymbol{x}_0$ by $f(x_o)$
While Loop: till $t \leq ITR$
  For loop: for each $i = 1 \dots n$ all $n$ fireflies
  Inner loop: for each $j = 1 \dots n$ all $n$ fireflies
       If $(I_i < I_j)$, move firefly i towards j; end if.
     Varying attractiveness with distance $r$ due to $e^{-\gamma r}$
    Evaluate new solutions and update the light intensity I
End For j
  End For  i
Rank the fireflies and find the current global best values
End While
Output: the best fireflies solutions$\boldsymbol{x}$
```

6. SIMULATION RESULTS

In this section, we show the utilization of the firefly algorithm to design optimum FIR filters for different cases. In all cases, for the sake of simplicity, the filter to be designed is assumed to be a linear phase LPF with even length to construct just 50% of the coefficients rather than all coefficients. Phase linearity of the filter is guaranteed by assuming symmetry of the approximate filter which reduces the dimension of the optimization problem to *M/*2. In the first case, the firefly algorithm is used to design optimum FIR filters in which *M*, *w*p, *w*s, and the ratio *δ*p/*δ*s are fixed. For this case, the fitness function to be minimized is defined as:

$$F_{\cos t} = \max_{w \in F_p \cup F_s}(|\, E(w)\, |), \tag{17}$$

Where F_p and F_s are the closed subsets $0 \leq w \leq w_p$ and $w_s \leq w \leq 0.5$, respectively. Figure 1 presents the frequency response of an approximate filter with *M*=30, *w*p=0.25, *w*s=0.3, *δ*p/*δ*s=1, based on theFirefly algorithm, DE, PSO and PM algorithms, on the same graph. The figure shows that the Firefly response is precisely giving the standard FIR optimal design. This signifies that the firefly algorithm is performs well along the PSO and DE algorithm in terms of accuracy. In this case, the firefly, DE and PSO were randomly initialized and converged within a reasonable time.

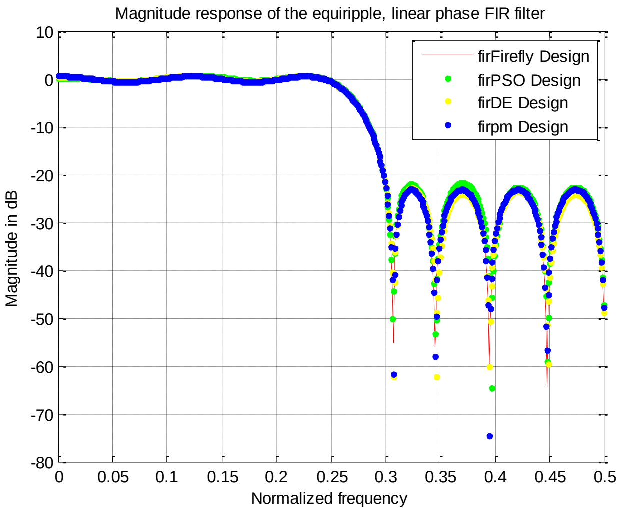

Figure 1. Frequency response of linear phase FIR filter for M=30, wp=0.2, ws=0.25, δp/δs=1

For settled parameters case, the firefly algorithm is utilized to design a linear phase FIR filters in which *M*, *w*p, *w*s, *δ*p, and *δ*s are chosen by the designer. In Figure 2, the design process begins with the desired filter parameters (*M*, *w*p, *w*s*)*, and the PM algorithm is employed to obtain the filter coefficients. Then, the designer selects a feasible value for *δ*p and *δ*s based on the maximum ripple size (δ_{PM}) obtained by the PM.To have the capacity to control the ripples in the two bands independently, one can re-write the cost function in (17) as follows:

$$F_{cost} = \left|\max_{w\in F_p}(|E(w)|-\delta_p)\right| + \left|\max_{w\in F_s}(|E(w)|-\delta_s)\right| \qquad (18)$$

Figure 2 shows the frequency response of a linear phase FIR designed based on the firefly algorithm in which the filter parameters are set to *M*=30, *w*p=0.25, *w*s=0.3, *δ*p=0.1,*δs*=0.01. The PM response for *M*=30, *w*p=0.25, *w*s=0.3, and *δ*p/*δ*s=100 is also shown in the same figure. Obviously, Figure 2 shows that the PM algorithm does not allow to control the passband and stopband ripples. However, the Firefly algorithm can control these qualities of ripple in the passband and stopband simultaneously. The firefly algorithm achieves to the desired ripples in both bands.

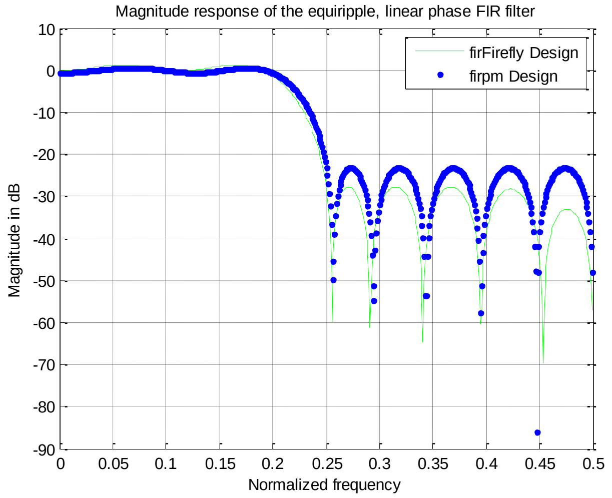

Figure 2. Frequency response of linear phase FIR filter, forM=30, wp=0.25, ws=0.3, δp=0.1, δs=0.01

It is unrealistic to acknowledge FIR channels without coefficients quantization utilizing finite word length. This outline requirement has not been considered in the prior work or even in the PM algorithm. Considering the limitations in filter implementation, the suitable word length reduce algorithm complexity and equipment prerequisites can prompts a dramatic change in the frequency response and design specification of the desired FIR. However, the capacity of firefly algorithm to deal with this sort of issues is exhibited in the accompanying cases. First, the impact of the quantizing process for the filter coefficients on the frequency response of filters designed based on the PM is demonstrated in Figure 3. In this case, the coefficients acquired based on the PM method are rounded by eight bits word length. As shown in Figure 3, the response of the PM filter with 8-bits quantized coefficients changes and its' performance diminishes comparing to the floating-point PM case. In Figure 4, we present the result of the presented algorithm to construct the optimum FIR filter for which their coefficients are rounded using 8 bits word length for the same filter in previous case.

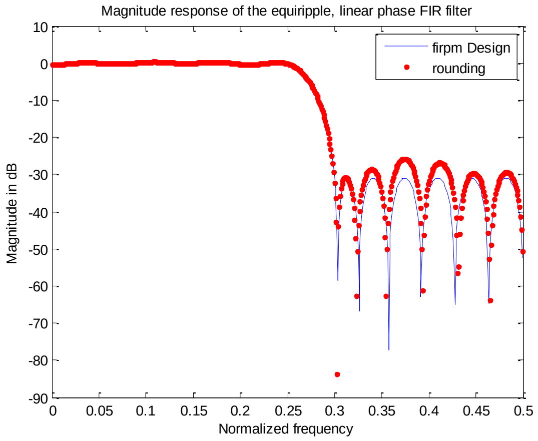

Figure 3. Effect of rounding on FIR filters designed by PM

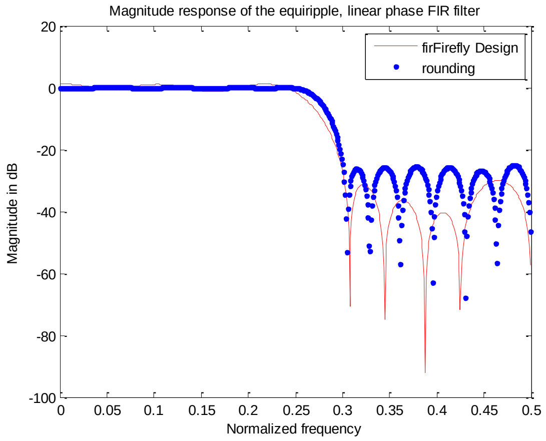

Figure 4. FIR filter designed using DE with coefficients quantized using 8 bits word length for M=30, wp=0.25, ws=0.3, δp/δs=1

Clearly, the Firefly performance is better than the one acquired from quantizing of the coefficients of the PM method. It is worth to study the impact of the order of the FIR filter on the presented method. In another example, the Firefly algorithm is employed to design an FIR filter with specifications M=15, wp=0.25, ws=0.3, δp/δs=1 in which the coefficients are rounded using 8 bits word length. Figure 5 presents the performance of using the presented method to construct the optimum filter for which the coefficients are quantized using eight bits word length. Obviously, the Firefly method preforms well and better than the one acquired from rounding of the coefficients of the PM method with smaller order filter.

In another example, we increase the quantization bits to equal 10 bits. The firefly algorithm is employed to design an FIR filter with specifications M=30, wp=0.25, ws=0.3, δp/δs=1 in which the coefficients are rounded by 10 bits word length. The results depicted in Figure 6 show that the Firefly method get an enhancement on the performance and still outperforms the rounding response of PM method.

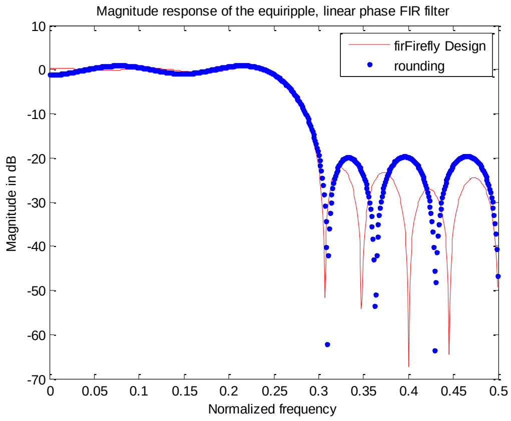

Figure 5. FIR filter designed using DE with coefficients quantized using 8 bits word length for M=15, wp=0.25, ws=0.3, δp/δs=1.

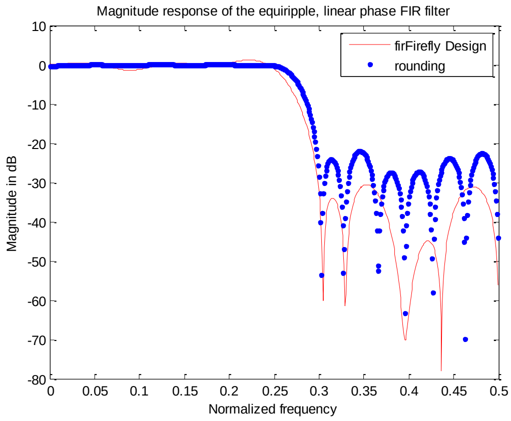

Figure 6. FIR filter designed using DE with coefficients quantized using 10 bits word length for M=30, wp=0.25, ws=0.3, δp/δs=1

7. CONCLUSION

Designing of linear phase FIR filter has been carried out through application of firefly algorithm. The FIR was compared with the PSO and DE algorithms. The firefly algorithm shows the ability to design such filters both in finite and infinite word length coefficients. There are three advantages tied to the firefly algorithm. That is getting to the true global minimum regardless the initial parameter, a few controlled parameters are used yet there is fast convergence. The results validate the fact that it is possible to design optimal finite word length FIR filters by use of firefly algorithm. Also, in allcases, the firefly algorithm performs well along with the DE and PSO algorithms. Although software implementation is important to investigate the capabilities of Firefly method and to simulate significant aspects of FIR Filter applications, hardware implementation provides real time solutions and an optimal parallelism method in terms of fast convergence. Hence, hardware implementations can be considered as a promising approach to implement the presented method which can be executed by Integrated Circuit (ICs).

REFERENCES

[1] L. Litwin, FIR and IIR digital filters, IEEE Potentials. 0278-6648, 2000, 28–31.

[2] Harpreet Kaur and Balwinder Dhaliwal, “Design of low pass FIR filter using artificial neural network”, International Journal of Information and Electronics Engineering, Vol. 3, No. 2, March 2013.

[3] Sonika Gupta, Aman Panghal, “Performance Analysis of FIR Filter Design by Using Rectangular, Hanning and Hamming Windows Methods”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 6, June 2012.

[4] J.H. McClellan, T.W. Parks, L.R. Rabiner, A computer program for designing optimum FIR linear phase digital filters, IEEE Trans. Audio Electroacoust. AU-21 (1973) 506–526.

[5] L.R. Rabiner, Approximate design relationships for low-pass FIR digital filters, IEEE Trans. Audio Electroacoust. AU-21 (1973) 456–460.

[6] Z. Albataineh, F. Salem, "New Blind Multiuser Detection in DS-CDMA Using H-DE and ICA Algorithms", International Conference on 2013 4th Intelligent Systems Modelling & Simulation (ISMS), pp. 569-574, Jan. 2013

[7] G. Singh, N. Prakash, “FPGA Implementation of Higher Order FIR Filter”, International Journal of Electrical and Computer Engineering (IJECE), Vol. 7, No. 4, pp. 1874~1881, August 2017.

[8] M. Ali, “Cascaded Ripple Carry Adder based SRCSA for Efficient FIR Filter”, Indonesian Journal of Electrical Engineering and Computer Science, Vol. 9, No. 2, pp. 253~256, February 2018.

[9] S. Das and P. Suganthan, "Problem definitions and evaluation criteria for cec 2011 competition on testing evolutionary algorithms on real world optimization problems", Jadavpur university and Nanyang Technological University, Tech. Rep., 2011.

[10] Gopinathan S., et al., “Wavelet and FFT Based Image Denoising Using Non-Linear Filters,” International Journal of Electrical and Computer Engineering, vol/issue: 5(5), pp. 1018-1026, 2015.
[11] M. Sharma, Dr S.K. Singh, “New Technique Based Peasant Multiplication for Effcient Signal Processing Applications”, Indonesian Journal of Electrical Engineering and Computer Science, Vol. 8, No. 3, pp. 726 ~ 729 December 2017.
[12] T.W. Parks, J.H. McClellan, Chebyshev approximation for non-recursive digital filters with linear phase, IEEE Trans. Circuits Theory CT-19 (1972) 189–194.
[13] Z. Albataineh, F. Salem and J.I. Ababneh, “Linear phase FIR Low Pass Filter Design Using Hybrid Differential Evolution”, International Journal of Research in Wireless Systems (IJRWS), Vol. 1, Issue 2, pp. 43-49, December 2012.
[14] Jehad I. Ababneh, Mohammad H. Bataineh. "Linear phase FIR filter design using particle swarm optimization and genetic algorithm". Elsevier, Digital Signal Processing, 2006.
[15] Q. Zhao, G. Meng, “Design of Digital FIR Filters using Differential Evolution algorithm based on Reserved Gene”, International Conference of Information Science and Management Engineering (ISME), pp. 177-180, 2010.
[16] Yang, X. and Karamanoglu, M., Swarm Intelligence and Bio-Inspired Computation. Swarm Intelligence and Bio-Inspired Computation, pp.3-23, 2013.
[17] F. Neri and V. Tirronen, "Scale factor local search in differential evolution", Memetic Computing, vol. 1, pp. 153-171, 2009, 10.1007/s12293-009-0008-9.
[18] S. Das and P.N. Suganthan, "Differential evolution: A survey of the state-of-the-art", Evolutionary Computation, IEEE Transactions on, 2010.
[19] Farook, S. and Raju, P. (2013). Evolutionary Hybrid Genetic-Firefly Algorithm for Global Optimization. IJCEM International Journal of Computational Engineering & Management, 16(3), pp.37-45.
[20] J. Kennedy, R. Eberhart,’ "Particle Swarm Optimization", in Proc. IEEE int. Conf. On Neural Network, 1995
[21] Y. Shi and R.C. Eberhart (1998), Parameter Selection in Particle Swarm Optimization, Evolutionary Programming VII, Springer, Lecture Notes in Computer Science 1447, 591–600.
[22] S. Das, A. Abraham and A. Konar, “Particle Swarm Optimization and Differential Evolution Algorithms: Technical Analysis, Applications and Hybridization Perspectives”, Studies in Computational Intelligence (SCI) 116, 1-38 (2008).
[23] El-Sawy, A., Zaki, E. and Rizk-Allah, R. (2012). A Novel Hybrid Ant Colony Optimization and Firefly Algorithm for Solving Constrained Engineering Design Problems. Journal of Natural Sciences and Mathematics, 6(1), pp.1-22.
[24] D. Mandal, S.P. Ghoshal, and A.K. Bhattacharjee, “A Novel Particle Swarm Optimization Based Optimal Design of Three-Ring Concentric Circular Antenna Array,” IEEE International Conference on Advances in Computing, Control, and Telecommunication Technologies, (ACT’09), pp. 385-389, 2009.
[25] Ab Wahab MN, Nefti-Meziani S, Atyabi A (2015) A Comprehensive Review of Swarm Optimization Algorithms. PLoS ONE 10(5): e0122827. doi:10.1371/journal.pone.0122827.
[26] Yang, X. (2010). Nature-Inspired Metaheuristic Algorithms. 2nd ed. Frome: Luniver Press.
[27] Yu, S., Yang, S. and Su, S. (2013). Self-Adaptive Step Firefly Algorithm. Journal of Applied Mathematics, pp.1-8.
[28] Tilahun, S. and Ong, H. Modified firefly algorithm. Journal of Applied Mathematics, pp.1-12, 2012.
[29] Wang, G., Guo, L., Duan, H., Liu, L. and Wang, H., A Modified Firefly Algorithm for UCAV Path Planning. International Journal of Hybrid Information Technology, 5(3), pp.123-144, 2012.
[30] Kumbharana, S. and Pandey, G. (2013). Solving Travelling Salesman Problem using Firefly Algorithm. International Journal for Research in Science & Advanced Technologies, 2(2), pp.53--57.

BIOGRAPHIES OF AUTHORS

Moath Sababha was born in Irbid, Jordan in 1987. He received his B. Sc. degree in Electrical Engineering from Jordan University of Sc. and Tech. JUST, Irbid, Jordan in 2010. He received his M. Sc. degree in Electrical Engineering from Oakland University, Rochester, Michigan, U.S.A in 2013. He is currently a Ph. D. candidate in Electrical and Computer Engineering at Oakland University, Rochester, Michigan, U.S.A. His research interests are in the areas of digital signal processing, nonlinear estimation and prediction, fuzzy logic and decision making.

Mohamed A. Zohdy was born in Caro, Egypt. He received B.Sc. degree in Electrical Engineering, Cairo University, Egypt in 1968 and he received his M. Sc. and Ph. D. degrees in Electrical Engineering from University of Waterloo, Canada in1974, and 1977 respectively. He worked in variousindustries: dowty, iron and steel, and spar. He is currentlya Professor at Oakland University, Rochester Hills, Michigan, U.S.A. His research interests are in areas ofcontrol, estimation, communication, neural networks, fuzzy logic and hybrid systems.