International Journal of Engineering Innovation & Research
Volume 2, Issue 4, ISSN: 2277 – 5668

# Optimization of FIR Filter Using PSO Based Algorithm

Shruti Singh, Rekha Agarwal, Sandhya Sharma

***Abstract*** **– In this dissertation, a digital filter is being optimized using Particle Swarm Optimization and compared with conventional frequency sampling method of optimization. Among IIR and FIR filter, here we discuss FIR filter optimization where firstly a single particle is being studied and then iterations are done for the rest. As for FIR filter, an important point to keep in notice is that an FIR filter is designed using convolution rather than recursion which results in higher quality of stability. An FIR filter is a linear phase filter where both phase delay and group delays are constant. The PSO is a population-based algorithm where a set of potential solutions evolve to approach a convenient solution (or set of solutions) for a problem. Being an optimization method, the aim is to find the global optimum of a real-valued function (fitness function) defined in a search space. In this, a population of individuals (referred to as particles) adapts by returning stochastically toward previously successful regions. Every particle's movement is a composition of initial random velocity and two randomly weighted influences i.e. individuality (the tendency to return to the particle's best previous position) and sociality (the tendency to move towards the neighborhood's best previous position).After every cycle of iteration, a new value of velocity for each particle is calculated based on its current velocity, its distance from the previous best position, and its distance from the global best position. The new velocity value is then used to calculate the next position of the particle in the search space. We use PSO as an optimization technique to optimize the output parameters of the FIR filter. PSO initializes a group of random particles (solutions) and then searches for optimal solution by updating generated values. The particle swarm algorithm is used here in terms of social cognitive behavior. It is widely used for problem solving method in engineering. When the search space is too large to search exhaustively, population based searches may be a good alternative.**

***Keywords*** **– FIR Filter, Frequency Sampling Method, Continuous PSO Algorithm, Binary PSO Algorithm.**

## I. Introduction

For a discrete-time FIR filter, the output is a weighted sum of the current and a finite number of previous values of the input. The operation is described by the following equation, which defines the output sequence y[n] in terms of its input sequence x[n]:

$$y[n]=b_o x[n] + b_1 x[n-1]+\ldots\ldots\ldots+b_N x[n-N]$$

$$= \qquad (1)$$

- x[n] is the input signal,
- y[n]is the output signal,
- $b_i$ are the filter coefficients that make up the impulse response,
- N is the filter order [1].

This difference equation gives y (n) in terms of input values only and it does not depend upon the other output values therefore it is called as *non-recursive filters.*

Particle Swarm Optimization was firstly introduced by *Dr. Russell C. Eberhart* and *Dr. James Kennedy* in 1995. The Particle Swarm Optimization algorithm is a biologically inspired algorithm involving social analogy [10]. The PSO is a population-based algorithm where a set of potential solutions evolve to approach a convenient solution (or set of solutions) for a problem. Being an optimization method, the aim is to find the global optimum of a real-valued function (fitness function) defined in a search space [2]. The metaphor that led to this algorithm can be perceived as that the individuals who are a part of a society hold an opinion which is part of a "belief space" (the search space) shared by every possible individual. Individuals may or may not modify this "opinion state" based on three factors:

- The knowledge of the environment (its fitness value)
- The individual's previous history of states (its memory)
- The previous history of states of the individual's neighborhood

The neighborhood of an individual is what configures its social network. Many neighborhood topologies are present such as full, ring; star, etc. depending on whether an individual interacts with all, some, or only one of the rest of the population. Following certain rules of interaction, the individual in the population adapts its scheme of belief to the ones that are more successful among its social network. Gradually, a culture arises wherein the individuals hold opinions that are closely related.

The concept underlying the basic PSO algorithm is that, that in the PSO algorithm each individual is called a "particle", and is subject to a movement in a multidimensional space that represents the belief space. Particles have memory which means they retain part of their previous state or history. There are no restrictions on particles so as to hold the same point in belief space, but in any case their individuality shall be preserved. Every particle's movement is a composition of initial random velocity and two randomly weighted influences i.e. individuality (the tendency to return to the particle's best previous position) and sociality (the tendency to move towards the neighborhood's best previous position).

There are two versions of the basic PSO algorithm viz. the continuous PSO algorithm and the binary PSO algorithm. The two can be briefly explained as follows:

*i) The Continuous PSO algorithm*: This uses a real-valued multidimensional space as belief space [8], and evolves the position of each particle in that space using the following equations:

$$v_{id}(t) = v_{id}(t-1) + c_1 \varphi_1 (p_{id} - x_{id}(t-1)) + c_2 \varphi_2 (p_{gd} - x_{id}(t-1)) \qquad (2)$$

$$x_{id}(t) = x_{id}(t-1) + v_{id}(t) \qquad (3)$$

where,

International Journal of Engineering Innovation & Research
Volume 2, Issue 4, ISSN: 2277 – 5668

$p(x_{id}(t) = 1)$ is the probability of deciding 1 at the $d_{th}$ position in a bit string
$p(x_{id}(t) = 0)$ is the probability of deciding 0 at the $d_{th}$ position in a bit string
$x_{id}(t)$ is the current state of the bit string at site d of an individual i
$v_{id}(t - 1)$ is a measure of the individual's predisposition or probability of deciding 1
t represents the current time step and t -1 represents the previous time step
$p_{id}$ is supposed as the best state found, for example it is 1 if the individual's when $x_{id}$ was 1 otherwise 0
$p_{gd}$ is represented as the best state of neighborhood. It is 1 if the best success attained by any member of the neighborhood was 1, otherwise it is zero.

The particle which is used to calculate $p_{gd}$ depends on the type of neighborhood selected. In the basic algorithm either a global ($g_{best}t$) or local ($l_{best}$) neighborhood is used. In case of global neighborhood, every particle is considered while calculating $p_g$. In the local neighborhood, a neighborhood is composed of only a certain number of particles amongst the whole of the population. The local neighborhood of a given particle does not change during the iteration of the algorithm. A constraint ($V_{max}$) is imposed on $V_{id}$ to ensure convergence. The value of $V_{id}$ is usually kept within the interval [$-x^{max}_{id}$, + $x^{max}_{id}$], with $x^{max}_{id}$ being the maximum value of the particle position [14]. A large inertia weight ( $\omega$ ) factor initiates global search while a small inertia weight factor initiates local search. In cases where inertia is used, it is sometimes decreased linearly during the iteration of the algorithm, starting at an initial value close to eq. (2). An alternative formulation of eq. (2) adds a constriction coefficient that omits or replaces the velocity constraint ($V_{max}$). The PSO algorithm requires tuning of certain parameters viz., the individual and sociality weights ($c_1$, $c_2$), and the inertia factor ( $\omega$ ) [3]. Both kinds' i.e. theoretical and empirical studies are available to help in selection of proper values for the algorithm [7], [8], [9], [10], [11].

*ii) The Binary PSO:* This algorithm has attracted much lesser attention in previous work. In this version, the particle's position is not a real value, but a 0 or a 1 [7], [12]. The logistic function of the particle velocity is used as the probability distribution for the position, that is, the particle position in a dimension is randomly generated using that distribution [8]. The equation that updates the particle position becomes the following:

$$p(x_{id}(t) = 1) = f(x_{id}(t - 1), v_{id}(t - 1), p_{id}, p_{gd}) \qquad (4)$$

The deciding of zero is given by (1-P)

$$p(x_{id}(t) = 0) = 1 - p(x_{id}(t) = 1) \qquad (5)$$

This implies that a binary PSO algorithm without individual and social influences i.e. with $c_1$=$c_2$=0 would still perform a random search on the space (the position however in each dimension would have a 0.5 chance of being a 0 or 1). The selection of parameters for the binary PSO has not been a subject of deep study in known works. This binary PSO has not been widely studied and some issues are still open. Some modifications on the binary algorithm equations propose a quantum approach [4], [11], [12].

The flow diagram below explains the process followed while working with PSO algorithm:

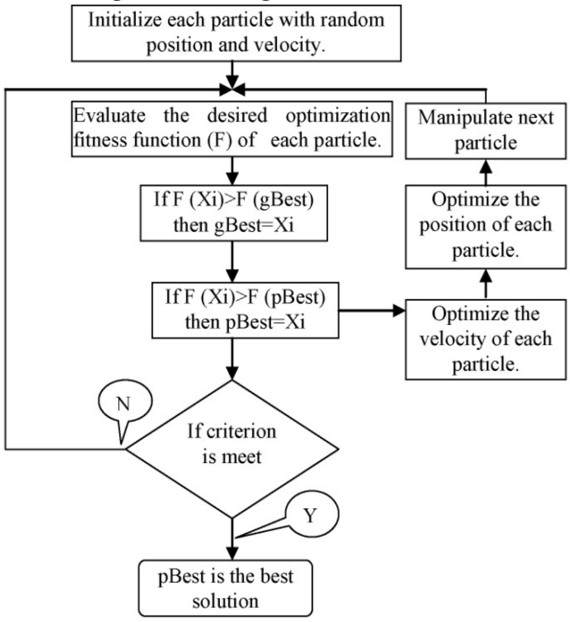


Fig.1. Flow diagram of PSO

## II. Limitations Of Existing Work

The primary disadvantage of FIR filters is that they often require a much higher filter order than Correspondingly, the delay of these filters is often much greater than, for an equal performance IIR filter. The problems with other conventional methods of designing an FIR filter are as follows:

1. The Window method is applicable only if $H_d(w)$ is absolutely integrable. When $H_d(w)$ is complicated or cannot easily be put into a closed form mathematical expression, evaluation of $h_d(n)$ becomes difficult [5].
2. The use of windows offers very little design flexibility e.g. in low pass filter design, the pass-band edge frequency generally cannot be specified exactly since the window smears the discontinuity in frequency [6].
3. Window method is useful for design of prototype filters like low-pass, high-pass, band-pass etc. They are not very suitable for designing of filters with any given frequency response. This makes its use in speech and image processing applications very limited.
4. The frequency sampling technique is suitable for designing of filters with a given magnitude response. The ideal frequency response of the filter is approximated by placing appropriate frequency samples in the z-plane and then calculating the filter coefficients using the IFFT algorithm.
5. Also the frequency sampling technique gives errors in the frequency response at points where it is not sampled. In order to reduce these errors the different optimization technique for FIR filter design were presented wherein the remaining frequency samples are chosen to satisfy an optimization criterion.

International Journal of Engineering Innovation & Research
Volume 2, Issue 4, ISSN: 2277 – 5668

## III. Proposed Work

In this digital filter design method, a comparison is established between a filter designed using PSO algorithm using frequency sampling and a filter designed using the simple frequency sampling method. The FS technique has the advantages that more effective narrow band filters can be found easily, and those filters can be designed with an arbitrary response major drawback is observed where to find the values of the transition band frequency sample values that produce a filter with the maximum stop band attenuation becomes a major task.

Particle swarm optimization (PSO) is an evolutionary computation technique. With very few parameters to adjust, it is extensively used in major applications. In PSO, each potential solution is assigned a randomized velocity, and the potential solutions, named as particles, with an important characteristic of memory are then "flown" through the problem space. Here, we use PSO to optimize the transition band frequency sample values is investigated. To design a FIR filter through frequency sampling technique by particle swarm optimization, there are two issues: establishing the fitness function according to the parameter of FIR filter and implementing PSO based on the fitness function. In order to determine which of the member of the population contain solutions that are good enough to continue to the next time step, the "fitness" of each particle must be found. This is performed by the fitness function using FS. Use FS to generate the desired frequency response according to the parameter of FIR filter, such as pass band edge $w_p$, stop band edge $w_s$, max pass band ripple $R_p$, min stop band attenuation $A_s$,. Then choose the sample number N, and the number of the inserted points D and values in the transition band w ($w_s < w < w_p$). Samples values are placed respectively and represented as T, (0 < T< l, i=1, 2... D).The key task is to find the best T, to produce maximum stop band attenuation using PSO.

## IV. Simulation and Analysis

Here we will represent the results of simple frequency and PSO based frequency sampling designing method of FIR filter and their parallel comparison. All the results are created using FDA Tool in MATLAB. We take two samples in transition band and apply PSO as optimization technique to optimize their values. The two transition samples are represented as $T_1$, $T_2$, and their values lies between 0 and 1, then using PSO we optimize their values. When we take 40 individuals, number of iterations as 20 then the way T1 and T2 get optimized, and how error reduces with the number of iterations is shown below:

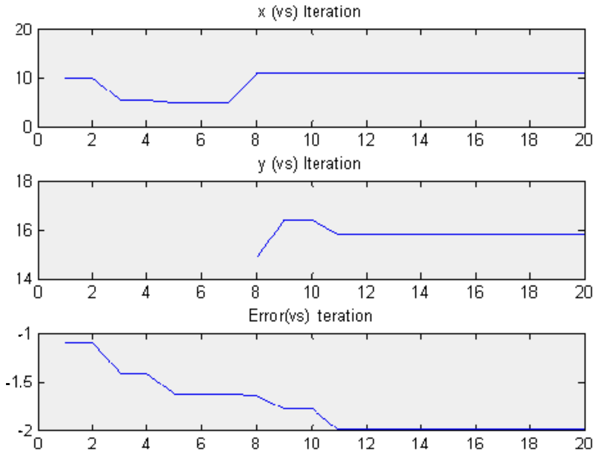

Fig.2. Iteration Vs Optimization

Now we show the magnitude versus normalized frequency plot for both simple frequency sampling designing method and PSO based frequency-sampling method.

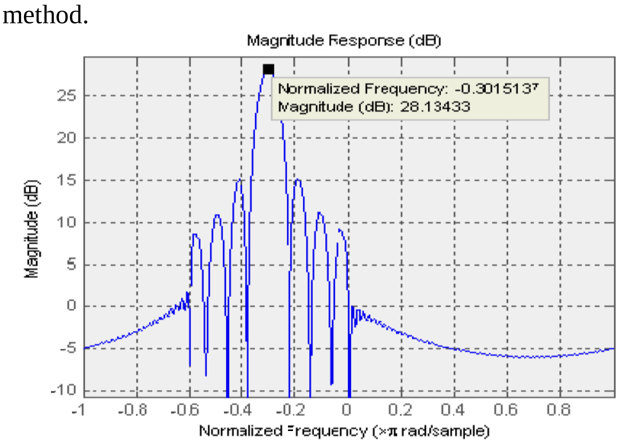

Fig.3. (a) Magnitude vs. Normalized Frequency plot for Simple Frequency Method

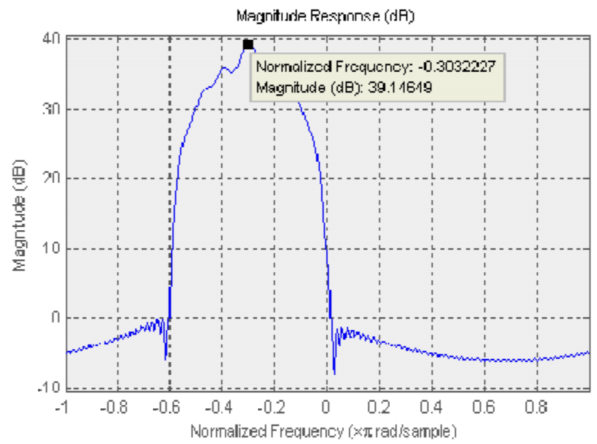

Fig.3. (b): Magnitude Vs Normalized frequency plot for PSO based FS method.

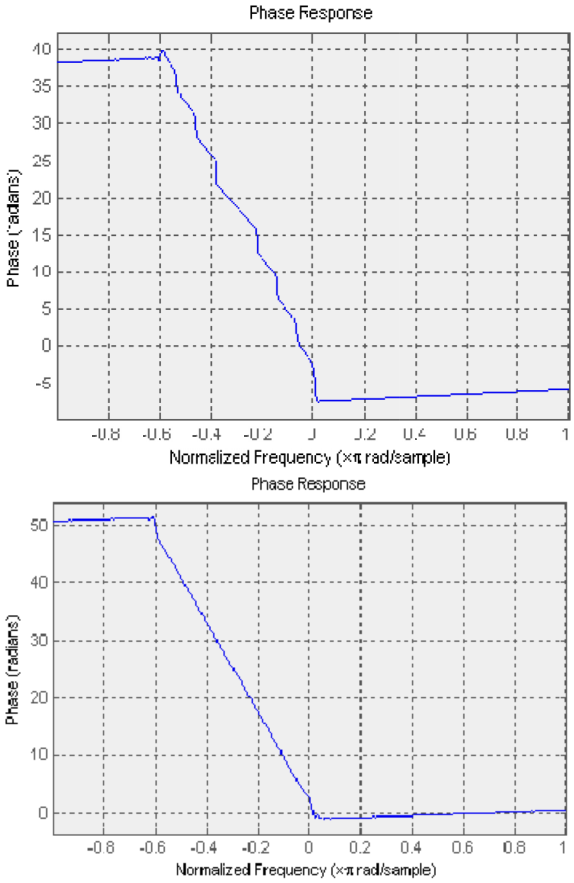


Fig. 4: Phase Vs Normalized frequency

From the fig.2 and fig.3, it is clear that the magnitude response of PSO based frequency sampling is better than simple frequency sampling method. Magnitude gain is also increase nearly about 12 db at 3012 Hz in PSO based frequency sampling method.

From the fig.4, it is clear that phase response is more linear in PSO based frequency sampling method as compared to simple frequency sampling method.

## V. Conclusion

A new method of designing is shown to be better in terms of magnitude and phase response than the conventional method. In addition, it is observed to produce maximum stop band attenuation with optimal solution. The design method and selection of parameters is also shown to be easier. Moreover, the feasibility and advantages of PSO is obviously represented. As shown above the output parameters of FIR filters have been improved using the PSO as the optimization technique. There is lot of difference between SFS method and PSO based method. Magnitude response is better by using PSO as compare to SFS method. Gain also increases in PSO based method. Phase response is also better in PSO based method. We can observe that all the parameters of FIR filter are better after using PSO. Both the samples $T_1$ and $T_2$ in transition band optimize and error decrease with iterations. Therefore, we conclude that the PSO is an efficient global optimizer for continuous variable problems. PSO can be easily implemented, with very little parameters to fine tune and algorithm modifications improves the ability of local search of PSO.

## Future Work

The further scope of work may be forecasted in the following terms, such as:

1. Incorporating evolutionary algorithms for multi-modal optimization issues.
2. Making SIMULINK model of the filter designed by this technique for digital circuits.
3. VHDL simulation & synthesis of the final design for targeting a FPGA or CPLD chip.
4. Adding visualization effects to make it user friendly.

## Acknowledgment

I am extremely grateful to my parents for their continuous support and motivation, for all the patience they have shown and for guiding me during my dark times.

## References

[1] Rabiner, Lawrence R., and Gold, Bernard, 1975: Theory and Application of Digital Signal Processing (Englewood Cliffs, New Jersey: Prentice-Hall, Inc.)

[2] Yan Zhe-ping; Deng Chao; Zhou Jia-jia; Chi Dong-nan. A novel two-subpopulation particle swarm optimization Intelligent Control and Automation (WCICA), 2012 10th World Congress on Publication Year: 2012, Page(s): 4113 – 4117.

[3] Xinhua Wu A density adjustment based particle swarm optimization learning algorithm for neural network design Electrical and Control Engineering (ICECE), 2011 International Conference on Publication Year: 2011, Page(s): 2829 – 2832.

[4] Tiago Sousa, Arlindo Silva, and Ana Neves- Particle swarm based data mining algorithms for classification tasks. Parallel Comput, 30(5-6):767-783, 2004.

[5] H. J. Trussell and D. M. Rouse, Reducing non-zero coefficient in fir designing using POCS, Department of Electrical and Computer Engineering, North Carolina State University.

[6] Wan-Ping, Li-Fang Zhou, Ji-Xin Qian Fir Filter Design: Frequency Sampling Filters By Particle Swarm Optimization Algorithm, Institute of Systems Engineering, Zhejiang University, Hangzhou, 3 10027.

[7] J. Kennedy and R. Eberhat, Particle Swarm Optimization in IEEE International Conference on Neural Networks, Perth, Australia, 1995.

[8] Maurice Clerc and James Kennedy-The particle swarm explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evolutionary Computation, 6(1):58-73, 2002.

[9] Frans van den Bergh- An analysis of particle swarm optimizers. Ph.D. thesis, University of Pretoria, South Africa, 2002.

[10] Ioan Cristian Trelea. The particle swarm optimization algorithm: convergence analysis and parameter selection. Inf. Process. Lett. 85(6):317-325, 2003.

[11] Y. Shi and R.C. Eberhart- Parameter selection in particle swarm optimization. In Proceedings of the Seventh Annual Conference on Evolutionary Programming, pages 591-600, 1998.

[12] J. Kennedy and R.C. Eberhart- A discrete binary version of the particle swarm algorithm. In Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics, pages 4104-4109, 1997.