DESIGN OF AN FIR LOW PASS FILTER USING BARE BONES PARTICLE SWARM OPTIMIZATION

A Dissertation submitted in fulfillment of the requirements for the degree of

MASTER OF ENGINEERING

in

Electronic Instrumentation and Control Engineering

Submitted by

Shashi Shekhar

(801351025)

Under the guidance of

Mr. Nirbhowjap Singh

Assistant Professor,

EIED, Thapar University,

Patiala

Electrical And Instrumentation Engineering Department

Thapar University, Patiala

(Declared as Deemed-to-be-University u/s 3 of the UGC Act., 1956)

Post Bag No. 32, Patiala – 147004

Punjab (India)

July, 2015

DECLARATION

I hereby certify that the work which is presented in dissertation entitled, "Design of an FIR low pass filter using bare bones particle swarm optimization", in partial fulfillment of the requirements for the award of the degree of Master of Engineering in Electronics Instrumentation and Control, submitted to Electrical & Instrumentation Engineering Department of Thapar University, Patiala is as authentic record of my own work carried under the supervision of Mr. Nirbhowjap Singh. It refers others researcher's work which are duly listed in the reference section. The matter contained in this dissertation has not been submitted, neither in part nor in full to any other degree to any other university or institute except as reported in text and references.

Shashi Shekhar
Roll No:801351025

Place: Patiala
Date: 21/07/2015

It is certified that the above statement made by the student is correct to the best of my knowledge and belief.

Date: 21/07/2015

Mr. Nirbhowjap Singh
Assistant Professor,
EIED, Thapar University,
Patiala

Countersigned by:

Head
EIED, Thapar University,
Patiala

Dean (Academic Affairs)
Thapar University, Patiala

ACKNOWLEDGEMENT

I would like to take the opportunity to express my gratitude to some people who were involved in this dissertation work. First, I owe my gratitude to my mentor Mr. Nirbhowjap Singh for doing everything from the inception of the work idea to giving invaluable suggestions at every step.

I am also thankful to Dr. Ravinder Agarwal, Head of the Department and Mr. Nirbhowjap Singh, PG Coordinator as well as all the faculty members and staff of the Department of Electrical and Instrumentation Engineering for being very supportive to me. I would also like to thank all my batchmates for motivating me all the time whenever I needed them and giving me useful tips. I thank all those who have contributed directly or indirectly to this work.

Lastly, I would also like to thank my parents for their years of unyielding love and encourage. They have always wanted the best for me and I admire their determination and sacrifice.

Shashi Shekhar
801351025

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

ADE: Adaptive Differential Evolution.
APA: Adaptive Parameter Adjustment.
APSO: Adaptive Particle Swarm Optimization.
BBDE: Bare Bones Differential Evolution.
BBPSO: Bare Bones Particle Swarm Optimization.
CF-AW-PSO: Adaptive Weight Particle Swarm Optimization Algorithm With Constriction Factor.
CRPSO: Craziness Based Particle Swarm Optimization.
CSD: Canonical Sign Digit.
DE: Differential Evolution.
DEWM: Differential Evolution With Wavelet Mutation.
ENSGA: Elitist Nondominated Sorting Genetic Algorithm.
FIPS: Fully Informed Particle Swarm.
FIR: Finite Impulse Response.
FWL: Finite Word Length.
GA: Genetic Algorithm.
IIR: Infinite Impulse Response.
MDE: Modified Differential Evolution.
MOPSO: Modified Particle Swarm Optimization.
NLMS: Normalized Least Mean Square.
NPSO: Novel Particle Swarm Optimization.
ODE: Opposition-Based Differential Evolution Algorithm.
OGA: Orthogonal Genetic Algorithm.
PLS: Projected Least- Square.
PM: Parks-Mcclellan.
POT: Power-Of-Two.
PSO: Particle Swarm Optimization.
PSOCFIWA: Particle Swarm Optimization With Constriction Factor And Inertia Weight Algorithm.
PSOIIW: Particle Swarm Optimization With Improved Inertia Weight,
PSO-LD: Linearly Decreasing Inertia Weight Particle Swarm Optimization.
QP: Quadratic Programming.
QPSO: Quantum-Behaved Particle Swarm Optimization QPSO-M: Modified Quantum-Behaved Particle Swarm Optimization.
RGA: Real Coded Genetic Algorithm.
SA: Simulated Annealing.
SPT: Sum Of Power Of Two.
VFD: Variable Fractional Delay.

ABSTRACT

Design of an FIR filter as per expected response is a field of wide interest. Now a day evolutionary algorithm are in lime light for filter design problem. Among the various algorithms bare bones particle swarm optimization (BBPSO) has capability to perform operation in multidimensional space with less number of control parameter, this make the algorithm simple and it is applied by simple computer code.

In this work design of linear phase low pass FIR filter is presented. The BBPSO is modified form of canonical PSO by removing the velocity term of the canonical PSO and replaced by Gaussian sampling strategy. To investigate the behavior of BBPSO based filter design approach, experiments are performed on standard benchmark test functions and filter design problem. Two cases of filter design with filter order 20^{th} and 30^{th} have been realized using BBPSO based solution approach. This work considers a fitness function based on the mean squared error between the actual and the ideal filter response. In order to compare the performance, the PSO algorithm is also simulated. A comparison of simulation results reveals the optimization efficacy of the BBPSO algorithm over the PSO optimization techniques for the solution of the multimodal, non-differentiable, highly non-linear FIR filter design problems.

Chapter1

INTRODUCTION

Filtering is a process by which the frequency spectrum of a signal can be modified, reshaped, or manipulated according to required specifications. Mainly two operations are performed by digital filter named as signal separation and signal restoration [1]. A digital filter is an algorithmic process to extract discrete time signal by doing mathematical operations to reduce or enhance certain aspects of the original signal. In comparison with analog filter digital filter gives flexibility to used with wide range of applications with better performance. The digital filters main advantages are summarized as: high accuracy, small physical size, high reliability stable performance, adjustable frequency response, versatility, reduced ripple in pass band, quick transition, and large attenuation in stop band [2], [3]. In comparison with analog filter, digital filter may be more expensive due to their software complexity [4].

The digital filters are of two types, finite impulse response (FIR) and infinite impulse response (IIR). An FIR filter is non-recursive in nature and has a finite duration impulse response, whereas, an IIR filter has continuous impulse response and recursive structure. Design simplicity and guaranteed stability are inherent property of FIR filter, which makes them easy realizable by any design process. FIR filters have desirable characteristics of ensured stability, approximate phase linearity characteristic in all frequency bands and their non-recursive structures are digitally implemented [5].

Traditionally, to design digital filters, various approach are used,such as: windowing approach, frequency sampling approach, least square error approach and so on [6]. The window approach is the most popular due to convenience, and robustness, however generally results in a suboptimal solution [4] and lacks frequency response control in pass band and stop band, large transition width[7] etc. In frequency sampling method the given frequency response is sampled at a set of equally spaced frequencies to obtain N samples. The continuous frequency response is obtained by Using interpolation of these N-point filter response [8]. This method has disadvantages such as interpolation produce desired frequency response are only at sampled points and remaining portion always contain a finite error [9].

The other approach to design of weighted Chebyshev FIR filter with exact linear phase and based on the Remez-exchange algorithm proposed by [10]. The major drawback of this algorithm is that, in frequency band the fitness error is calculated by its defined weighting function instead of deviation of respective band. As seen in classical optimization problem of filter design, designer always compromise with some design specifications. So, evolutionary optimization methods are used to design digital filters with better control parameter and approximate the ideal filter [7].

Recently various stochastic optimization algorithm have been developed, which is suitable for digital filter design. Some of examplary cases are GA [3], [11][12], [13], simulated annealing [14], [15], tabu Search [16]. GA only finds local minimum efficiently but in case of finding global minimum in terms of convergence speed, it is unsuccessful as well as it used a large number of control parameter which increased the algorithm complexity . In order to overcome disadvantage of the GA, a differential evolution (DE) method is proposed and implemented to design of digital filters by various researchers [7], [17], [18]. DE algorithms is concluded very slow to give result, control parameters size is large, and dependent on initial population [17]. Another stochastic technique particle swarm optimization (PSO) is developed by Eberhart and Kennedy. [19].The merits of PSO is simple algorithm and controlled by less parameters [20]. PSO is successfully implemented FIR filter design problem [21], [22]. The main limitation of PSO is that it can lead to premature convergence and stagnation at local minimum problem [23]. This conventional PSO improved [20], [24] and implemented in FIR filter design problem[1], [4].

Among the various proposed modifications to improve PSO performance, one of successful technique is BBPSO. BBPSO is proposed by Kennedy in 2003 [19] has advantage of almost parameter free tuning because it excludes velocity term, easy to implement and possesses self adapting mechanism to search optimal solution. It converges much faster than above specified algorithm and gave best optimal global solution accurately.

In this work BBPSO algorithm is successfully applied to optimize the coefficients of 20^{th} order and 30^{th} order linear phase low pass FIR filter. The simulation result reveals that in the both cases filter coefficients are optimized accurately with very fast convergence speed.

Chapter2

LITERATURE SURVEY

The design of digital filter with desired specifications is a challenging problem of digital signal processing. Many conventional design approach [8], [10] based on window technique, frequency sampling technique, weighted least square design technique, and equiripple design techniques are proposed to design FIR filter. These designs technique have some inherent drawbacks such as: many ripples occurred in stop-band and pass-band, design complexity crosses the limit of conventional design approach. Conventional design techniques have very less scope to handle multiple constraints of digital filter and give unsatisfactory result in terms of filter performance [25]. Due to the constraint of conventional method in FIR filter design, leads to complex and tiresome design process.

In order to overcome the drawbacks of conventional method in FIR filter design, the use of meta-heuristic optimization techniques emphasized, such as tabu search, simulated annealing , genetic algorithm, differential evolution algorithm and particle swarm optimization algorithm is used in digital FIR filter problem. These algorithms facilitate the solution of difficult or complicated design problem. Recent research work in area of digital filter designing and related meta-heuristic algorithms are summarized as follows:

McClellan J. *et al.* [1] presented the paper on the general purpose computer program for designing a large class of optimum FIR linear phase digital filter. The algorithm used the Remez exchange method to design filters with minimum weighted Chebyshev error and this error is approximated to find desired ideal frequency response. The authors concluded that this approach has capability for designing a large class of general filter types, differentiators, and Hilbert transformers as well as any desired magnitude response with fast speed, which is useful for a wide variety of design applications.

Qigang Z. *et al.* [26] proposed a method to designing the two dimensional FIR filter by using McClellan transformation. In this design method, first, 1-D FIR filter is designed by using cosin function in transition band, which gave the prototype filter, then design the 2-D filter using this prototype transition band functions. The authors concluded that design by using McClellan transformation can ease the design steps of 2-D FIR, operation quantity reduced and eliminate Gibbs phenomenon which occurs in window design.

Yunlong W. *et al.* [27] discussed about the most frequently used digital FIR filter design method and proposed a new simple method of filter design. The authors introduced a simpler method of digital linear phase FIR filter design using sinc sum function to overcome the disadvantages of window method and PM method of filter design. Finally, a series of filter formulas with the narrow transition width and simple calculation process is achieved. The authors concluded that the derived filter formula could be used to design notch filter, narrow pass-band filter and larger pass-band ripple type filter.

Xiaoping L. [28] presented the projected least square algorithms (PLS) for constrained FIR filter design by transforming the constrained FIR filter design problem with time or frequency domain linear constraints to positive definite quadratic programming (QP) problems. The authors presented the PLS algorithm for the positive definite QP problems and this algorithm applied to minimize unconstrained solution faithfully at the boundaries of active constraints with an active set strategy. PLS algorithm has capability to design constrained least-square digital filter directly and constrained Chebyshev design FIR filters iteratively.

Zhao H. *et al.* [5] presented the FIR filter design method by normalized least mean square (NLMS) adaptive algorithm, by this method any digital filter (FIR and IIR) or analog filter is designed by using its responses to design the desired FIR filter due to inherent property of algorithm. The authors concluded that the FIR filters generated by this method can have the exact amplitude and phase responses of target filter, when the target filter is a low order IIR filter, the adaptive filter can still model the amplitude response perfectly, but the difference in structure of FIR and IIR, phase response may vary.

Kaya T. and Ince M.C. [29] proposed the FIR filter design using Kaiser window and this window's parameter's value is calculated with genetic algorithm. The authors first obtained the Kaiser window function parameter, this value is used as a filter coefficient desired conditions and the amplitude response obtained from available commands are compared with the amplitude response obtained by optimized FIR filter coefficients.

Wang W. *et al.* [30] presented the frequency sampling method to design FIR filter and optimized filter coefficients by the Particle swarm optimization (PSO). This filter is designed by optimizing the variable sample in transition band for best performance. The authors summarized that by using frequency sampling technique, more effective narrow band filter is found easily and PSO algorithm is applied to optimize the transition sample values of the filter with maximum stop band attenuation.

Lu H. *et al.* [31] proposed an efficient method for design of FIR digital filter with arbitrary log magnitude and phase responses. The proposed design concept is based on GA with iterative approach to design the FIR log filters. The filter is designed by method of least-squares approximation which produces the log magnitude function and the weighted chebyshev function is updated each iteration according to the final value of previous iteration, such that error function is approximated as real or complex lograthmic error. .

Krusienski D.J. and Jenkins W.K. [32] presented an adaptive non-linear IIR filter design structure using Particle swarm optimization (PSO), by performing a randomized structured search of an unknown parameter space with manipulating the population to converge to the desired solution. The authors concluded that by using PSO, search technique became structure independent in case of adaptive filter and converging capability towards global solutions is increased for multimodal optimization problems, which makes them very useful to optimize coefficients of nonlinear and infinite impulse response (IIR) adaptive filters.

Ghosal S.P. *et al.* [33] presented a new filter design algorithm and named as Particle Swarm Optimization with Constriction Factor and Inertia Weight (PSOCFIWA), this method is based on manipulating the velocity term with constriction factor. The fitness function is taken as squared error between actual and desired filter and this error is minimized iteratively. The band stop filters of order 30 and 40 have been designed. The authors concluded that the ideal frequency characteristic of digital filters is met by PSOCFIWA and gave better result than GA in terms of speed of convergence as well as peak ripple value attenuation in pass band and stop band.

Ghosal S.P *et al.* [4] presented an FIR band stop filter design using PSO with improved inertia weight (PSOIIW). The authors described a new fitness function and modify weight of PSO such that, to achieve better ripple suppression in stop band. The authors concluded that modifying the inertia weight in PSO, magnifies the search process of PSO to obtain the global optimal solution.

Ghosal S.P. *et al.* [34] presented FIR band pass filter and high pass filter design [35] using novel particle swarm optimization algorithm (NPSO), by modifying the inertia weight. This modified weight mechanism main purpose is to control and monitor the linearly decreasing inertia weight, which affect the PSO largely. In this filter design process the authors have specified various desired filter specifications and used an improved fitness function based on PM technique.

Saha S. *et al.* [36] presented an IIR high pass filter design of 8^{th} order using craziness based particle swarm optimization (CRPSO) technique. This algorithm is based on the global heuristic search algorithm with exploration and exploitation of search space and this is obtained by using craziness operator for velocity and position, which is based on probability of finding best solution in desired search space. The authors concluded that CRPSO approach minimizes the fitness error function efficiently and produce optimized coefficients in comparatively less time .

Saha S. *et al.* [37] presented an design of linear phase low pass FIR filter using adaptive PSO (APSO), which is an improved method of conventional PSO. Here modifications are made for searching best optimal solution by balancing the exploration and exploitation stage in multidimensional problem. The authors used an improved error function based on fuzzy classification method (ESE) for filter design, compared the result using GA,PSO,PM and concluded that APSO converged exactly near best optimal solution at the much less execution time among the other specified algorithms.

Sarangi A. *et al.* [38] introduced two modifications to particle swarm optimization (PSO) and its variant quantum-behaved particle swarm optimization (QPSO), termed as QPSO-M and PSO-LD. The first one is based on quantum mechanics by replacing random vector with certain probability. The second modification is based on the controlling dependency of time of constriction factor. These two algorithms are used to design the digital IIR filter for the purpose of system identification. The authors concluded that the performance of QPSO-M and PSO-LD are better to that of PSO in the design of digital IIR filter also they have capability to become effective method for designing digital filtter.

Ababneh J.I. and Bataineh M.H. [39] proposed a linear phase FIR filter design using particle swarm optimization (PSO) and genetic algorithm (GA). The authors took two filter design approach, the first approach is based on PM algorithm as fitness function and defines filter specification while in another method a new fitness function is used for controlling the stop band and pass band ripples efficiently. The authors compared the filter design result obtained by the both the algorithms and concluded that the PSO outperforms the GA in proposed case.

Soleimani A. [40] presented the design of a digital FIR filter using the particle swarm optimization (PSO) algorithm with combined canonical sign digit (CSD) representation. The design method performs matching certain frequency response and the filter coefficients with limited bits in CSD representation. The authors concluded that combining PSO and CSD representation simultaneously give better result than combining PSO and CSD sequentially.

Boudjelaba K. *et al.* [41] studied the performance of two meta-heuristics, particle swarm optimization (PSO) and genetic algorithm (GA), for FIR filter design. In [19], the standard genetic algorithm, adaptive genetic algorithm, particle swarm optimization, novel particle swarm optimization, and craziness particle swarm optimization techniques have been applied to the design of FIR filters. The authors said that PSO algorithm is simpler and more intuitive to implement than GA, also CRPSO can converge best quality around optimal solution shows better convergence characteristics in much shorter run times than the GA, PSO, and NPSO algorithms in filter design problem.

Vasundhara *et al.* [42] presented an efficient way to design a linear phase low pass and high pass FIR filter using ADE-PSO, in which fitness function contains the property of both adaptive differential evolution (ADE) and PSO. The proposed algorithm has ability to overcome the disadvantages of individual algorithm which is instability and sub-optimality faced by DE and PSO respectively, when used alone. The authors designed the 20th order high pass and low pass filter using ADE-PSO and concluded that it gives a better magnitude response as well as lowest error value as compared to other algorithms.

Kar R. *et al.* [43] presented the FIR band stop filter design using craziness particle swarm optimization (CRPSO). The authors introduced a craziness operator and assign velocity and position with previously defined probability known as craziness to maintain the diversity of the particles and avoid stagnation in multidimensional problem of filter design. The authors compared the CRPSO algorithm performance with real coded genetic algorithm (RGA), conventional PSO, PM and concluded that proposed CRPSO algorithm has the least probability of premature convergence than other existing algorithm.

Karaboga N. and Cetinkaya B. [44] presented the digital FIR filter design using differential evolution (DE) algorithm and the performance comparison is made with genetic algorithm (GA) and least squares methods. The DE approach property is described as: fast convergence, few control parameters and provide true global optimum. The authors designed the digital FIR filters of different orders using DE and concluded that DE has a simpler structure than GA, and achieved desired optimal filter much quicker than the GA.

Chandra A. and Chattopadhyay S. [45] proposed a new approach for low pass FIR filter design using DE. This optimization technique has been explored to search the coefficients of impulse response of the FIR filter in the form of sum of power of two (SPT) to avoid the multipliers during the design process. In this paper, the authors have proposed a novel design strategy of multiplier-less low pass FIR filter with the help of differential evolution (DE) technique in to shorten its computational complexity without compromising the performance of system to a great extent.

Chattopadhyay S. *et al.* [46] presented the FIR pulse shaping filter design by convex optimization method and performance of the filter is compared with that of obtained by differential evolution (DE) optimization. In this paper, an attempt has been made to analyze the performance of traditional as well as evolutionary optimization technique to design the FIR filter. The FIR filter performance under both Convex and DE optimization techniques have been compared in terms of their magnitude response.

Reddy K.S. *et al.* [47] presented the low power, high performance FIR filter design using a modified differential evolution algorithm (MDE). The main advantage of MDE is that crossover operation is defined semi randomly, which increase the probability of finding global optimum accurately. The authors concluded that by using MDE hardware requirement reduces, performance of the designed filter improved.

Samanta G. and Chandra A. [48] proposed an alternative algorithm of differential evolution (DE) called opposition-based differential evolution algorithm (ODE), which is motivated by the fundamental steps of DE and based on an opposite member of the population and jumping factor, for the design of linear phase low-pass FIR filter. The authors designed the FIR filter using ODE and performance is compared with DE, based on these comparison, the authors concluded that ODE is fast enough to find the optimum response than DE, also the quality of solutions produced by ODE is better than DE.

Chandra A. and Chattopadhyay S. [28] presented the FIR filter design using DE and canonical signed digit (CSD) representation. The authors convert the discrete tap coefficients, which is organized as sum and/or differences of power of two, to canonical signed digit (CSD) and then optimized by the differential evolution (DE) algorithm. The authors concluded that the effect of the CSD representation of any binary number reduced the hardware complexity to designing low-pass FIR filter using the DE approach.

Reddy K.S. and Sahoo S.K. [49] proposed an approach for hardware efficient FIR filter coefficient optimization using differential evolution (DE) and common sub expression (CSE) elimination algorithm. The authors used DE algorithm and first find a set of filter coefficients with reduced number of sign-power-of-two (SPT) terms and then applied CSE elimination algorithm and in terms of adders hardware requirement is discussed.

Upadhyay P. *et al.* [50] proposed an improved version of differential evolution (DE) called differential evolution with wavelet mutation (DEWM) is applied to design the infinite impulse response (IIR) system identification problem. The authors adopted a scaling factor which is based on iteration, and in process of mutation a wavelet function is used. The proposed mutation process enhances the searching capability and gave true global optimum in less time. The authors concluded that, although complexity of algorithm is increased to design IIR system, but it gave the best quality output.

Liu G. *et al.* [51] proposed a modified differential algorithm (DE) based on reverse genes and applied this algorithm to digital FIR filter design. The proposed new DE algorithm use reserved genes of desired individual to enhance the diversity of the population and newly individuals are continuously generated during the evolution. The authors concluded that the new DE algorithm has capability to design the filter with higher order by performing much faster and better to get an approximation of filter coefficients but some ripples are present, these are further minimized by widening filter transition width.

Sharma S. *et al.* [52] presented the digital FIR filter design using differential evolution optimization with ripple constraint. The authors used optimization algorithm that is rely on frequency characteristics of the filter, and DE algorithm is used to minimize the ripples by constraint handling method based on penalty function. The authors concluded that DE algorithm with Ripple Constraint indicates less ripple value and almost uniform ripple size, in the stop band and this ripple is much less than the filter which designed by DE without ripple constraint.

Lee A. *et al.* [53] presented an algorithm for designing the 1-D FIR filter using genetic algorithm (GA). The proposed algorithm ensured that the resulting filter coefficients are guaranteed to be in canonical signed-digits (CSD) format and by allowing more than one non-zero power-of-two (POT) digits present in each of the Coefficients, for better performance and less complexity. The authors examined the effect of many error norms, and their impact on convergence speed and optimal result.

Xu D.J. and Daley M.L. [54] presented a parallel genetic algorithm (GA) to design optimal finite word length (FWL) low pass Chebyshev FIR digital filter. In the proposed algorithm, individuals within each population are expressed with a set of FWL coefficients, and optimization of these FWL coefficients occurs through the operation of selection, migration, mutation and crossover. The evolution of the population is guided by minimizing the frequency error function. The authors successfully designed the two FIR filters using parallel GA, first filter are low pass filter with a length of 40 coefficients and word length of 10 bits and second, is a high order filter with a length of 140 coefficients and word length of 16 bits with desired error tolerance.

An-xin Z. *et al.* [55] presented frequency sampling FIR filter design and approximation error is optimized using adaptive parameter adjustment (APA) genetic algorithm. The authors developed a method for evaluating stagnation problem, which can adjust the parameters in the process of genetic expression and enhance speed of convergence and global search finding capability. The authors applied APA genetic algorithms to optimize low pass, high pass and band pass FIR digital filter and compared this result to traditional GA, which showed that the APA genetic algorithm gives better result than traditional GA at the same problem.

Ahmad S.U. and Antoniou A. [56] presented a genetic algorithm (GA) based optimization approach called orthogonal genetic algorithm (OGA) for design of multiplierless digital FIR filter. In the proposed approach for the FIR filter realization, each coefficient is represented by the SPT terms. Each filter coefficients are treated as Chromosomes, which are constructed in matrix form with each row representing the power-of-two terms of the coefficients and the values of the coefficients are optimized with the OGA so as to achieve a desired amplitude response.

Khamei K. *et al.* [57] presented a new method for design of variable fractional delay (VFD) FIR digital filter using genetic algorithm (GA) by designing each subfilter of farrow structure separately with defined accuracy and bandwidth using GA to reduce computational complexity as well as design flexibility enhanced. This technique reduces the truncation error by using the SPT filter coefficient representation.

Ahmad S.U. and Antoniou A. [58] proposed the genetic algorithm (GA) for the design of asymmetric FIR filters that would satisfy various constraints imposed on the amplitude response and group delay characteristics. The authors used an approach based on an elitist non dominated sorting genetic algorithm (ENSGA) to find a Pareto - optimal solution for FIR filter. The proposed approach defined three individual objective functions based on the stop-band and pass band amplitude response error and a measure for the flatness of group-delay characteristics with respect to pass band with constraint. The authors concluded that using ENSGA minimum stop band attenuation and reduced peak pass band amplitude-response ripples are achieved.

Wysocka-Schillak F. [59] proposed a method for designing 2-D FIR linear-phase half-band filters based on two error criteria, equiripple error criterion in pass-band and least-squared error criterion in the stop-band. The author presented the two step solution procedure for the filter design, in the first step genetic algorithm (GA) is applied and the second step local optimization method is applied, whose starting point is same as the final point, obtained from the GA. The author concluded that proposed hybrid approach combined the advantage of GA with faster convergence and accuracy of the local optimization method.

Kennedy J. and Eberhart R. [19] introduced the PSO algorithm for continuous non-linear optimization problem. This concept of artificial life swarming theory is changed into algorithm, which can implement on problems with few lines of simple computer code. The authors describe the several concepts related to this algorithm such as: particle, pbest and gbest.

Eberhart R.C. and Shi Y. [60] presented a modified particle swarm optimizer (PSO) by introducing a parameter called inertia weight in original PSO algorithm. In order to examine the affect of the inertia weight on the PSO performance, the authors adopted the benchmark test problem of Schaffer's f6 function as a testing problem. The authors concluded that the PSO with the inertia weight in the range $[0.9, 1.2]$ on average will have a better performance; that is, it has a great probability to find the global optimum within a less number of iterations, again, when a time decreasing inertia weight is introduced a significant improvement on the PSO performance is obtained.

Eberhart R.C. and Shi Y. [61] presented the empirical study the performance of the particle swarm optimizer with linearly decreasing inertia weight and tested at four different standard benchmark test functions. The authors concluded that the testing results depicts that the quickly convergence behavior of PSO, the PSO efficiency is not sensitive to the swarm size, and PSO scales well. But PSO may be lacking the ability of global search at the end of last iteration because of the using the linearly decreasing inertia weight property and this disadvantage can be overcome by employing a self-adapting strategy for adjusting the inertia weight.

Clerc M. and Kennedy J. [62] presented the paper on particle swarm optimization and discussed various improvements in conventional PSO algorithm. The authors explored how the particle swarm algorithm works from the inside, i.e., from the individual particle's point of view, and analyzed a particle's trajectory as it moves in discrete time (the algebraic view), then progresses to the view of it in continuous time (the analytical view) which lead to a generalized model of the algorithm, containing a set of coefficients to control the system's convergence tendencies. The authors introduced various constriction coefficient models to find optimum solution. The author suggests methods to change conventional PSO, which will eliminate the stagnation problem and give the true global optimum accurately.

Mendes R. *et al.* [63] presented a fully informed particle swarm (FIPS) optimization algorithm. The authors proposed an modified approach that is conceptually more precise and guaranteed to perform more efficiently than the conventional particle swarm algorithm. In this new PSO model, the particle exercise the information from all its neighbors, rather than using only best one, i.e. made the individuals fully informed. The authors tested this algorithm on several benchmark test functions and concluded that the results provided by this approach is better, as fully informed individuals try to find better solutions in all the benchmark functions.

Jin Y. *et al.* [64] proposed an improved particle swarm optimization algorithm by introducing the improved simulated annealing operator, chaotic disturbance operator and Cauchy mutation operator into the variable inertia weight PSO. The authors said that the proposed algorithm increases the diversity of the population, and enhances the ability of global searching. The two typical benchmark functions (sphere function and Ackley function) are used to test the performance of the proposed algorithm and the authors concluded that proposed algorithm has simple structure, few control parameters and easy implementation which improve the convergence and precision largely.

Trelea I.C. [65]presented the paper on convergence analysis as well as selection of parameters of PSO algorithm. The authors discussed the exploration-exploitation tradeoff of PSO algorithm that greatly influences the algorithm performance, by performing the experiment with several benchmark test functions.

Tan D. *et al.* [66] proposed a modified particle swarm optimization (MOPSO) algorithm for engineering optimization problems with constraints, in which the penalty function is employed to the traditional PSO algorithm. The proposed algorithm adjusted the personal best and global best, the multi-objective problem can be converted into single objective problems. The constraint term is used here to generate particle and this guaranteed that each particle is generated within specified limits. The author tested this algorithm in actual engineering design optimization problem with GA and concluded that, MOPSO converge quickly and give better result than that of GA.

You Z. *et al.* [67] proposed an adaptive weight Particle Swarm Optimization algorithm with constriction factor (CF-AW-PSO) combined with an analysis of convergence of Particle Swarm Optimization algorithm to overcome the premature convergence caused by local optimization in the process of global optimization. The authors set the inertia weight according to dynamic information about the changes in the objective function to overcome the local optimum problem and increased global search capability. The performance of CF-AW-PSO algorithm is tested by using four benchmark test function and the simulation result is compared with the iterative evolution curve of Standard-PSO algorithm, CF-PSO algorithm, LDW-PSO algorithm and RandW-PSO algorithm. The authors concluded that CF-AW-PSO has a good ability to slow down premature convergence, compared to other improved particle swarm algorithm.

Ji W. and Wang K. [19] proposed an improved particle swarm optimization algorithm, by using, the gradient descent method (BP algorithm) as PSO operator. In this algorithm every evolutionary process used the BP operator to go to the next generation, and also this improved algorithm of particle position will not give limit restraint. The authors concluded that this improved algorithm has faster convergence velocity and improve the global optimum solution.

Jin-yue L. and Bao-ling Z. [68] presented the improved particle swarm algorithm by introducing mutation operation in the standard particle swarm optimization to increase the possibility of algorithm to search the optimal value. The mutation operator is introduced in PSO by re-initialized some variables according to a certain probability which can be any value in the range of $[0.1, 0.3]$. The linearly decreasing inertia weight particle swarm optimization is adopted to balance the global search and local search ability of the algorithm. The authors conducted the performance test for two benchmark test functions and concluded that convergence speed and global search capability of this improved algorithm has been significantly improved to some extent.

Jun X. and Chang H. [69] presented the discrete binary version of improved particle swarm optimization (BIPSO) algorithm to solve discrete optimization problem. This algorithm introduces the cross operation of genetic algorithm (GA) to implement the individual evolution that is, updating the formula to replace the speed and location.

The author made the best solution to evolutes by integrating simulated annealing in the evolutionary process, which can effectively avoid the blindly random particle evolution and the trap in locally optimal solution, thus enhances the overall search capabilities of the algorithm.

Chen G. [70] proposed a simplified particle swarm optimization algorithm based on particle classification. The author first divided the all particles into three categories, such as the better, the ordinary and the worse according their fitness values, and then, having a rational analysis on speed equation of the basic PSO, velocity term is removed, so the optimization process in basic PSO is more simple, and its convergence track is analyzed much conveniently. Next, three corresponding simplified models, such as "cognitive model", "social model" and "complete model" are used to adjust the iteration process of particles. In those three models, is adjusted linearly as setting in PSO-LDW. In the "complete model", c1 and c2 are adjusted linearly as the setting in time-varying acceleration coefficients PSO. The author used this algorithm for optimize benchmark test function and concluded that simplified PSO has better optimization performance than others.

Zhan Z.H. *et al.* [71] presented an adaptive particle swarm optimization (APSO) that features better search efficiency than classical particle swarm optimization (PSO). The APSO consists of two main steps. First, by evaluating the population distribution and particle fitness, a real-time evolutionary state estimation procedure is performed to identify one of the following four defined evolutionary states, including exploration, exploitation, convergence, and jumping out in each generation. It enables the automatic control of inertia weight, acceleration coefficients, and other algorithmic parameters at run time to improve the search efficiency and convergence speed. Then, an elitist learning strategy is performed when the evolutionary state is classified as convergence state. The authors concluded that APSO perform a global search over the entire search space with faster convergence.

Kennedy J. [56] presented the bare bones particle swarm algorithm. The author modified the traditional particle swarms optimization algorithm by eliminating the velocity formula. The Gaussian distribution is mean μ and deviation σ is used to update particles position. The position of particles converges very fast towards global optimum.

Pan F. *et al.* [72] presented the paper on the analysis of bare bones particle swarm optimization (BBPSO) algorithm, which is evolved from the canonical particle swarm optimizer. The authors demonstrated that the BBPSO can be mathematically deduced by the canonical PSO with neglecting the velocity term and a Gaussian sampling, based on the swarm best (gbest or lbest) and personal best (pbest) information, is used in place of the canonical formulas because from a mathematical point of view, the velocity is an auxiliary item in the PSO update equation, especially when there has been no new gbest or pbest updated for a couple of iterations, the impact of initial velocity rapidly decreases. The author said that compared with the canonical PSO, the BBPSO is more compact and there is no parameter tuning. So it is easy to be implemented and combined with other optimization methods

Richer T.J. and Blackwell T. [73] presented a new algorithm called Lévy bare bones particle swarm optimization. This algorithm is based on the Lévy distribution with large step length ("fat tails") which will induce exploration at any stage of the convergence and enabling escape from local minima. The authors replace the Gaussian distribution by Lévy distribution, and described an algorithm to generate random numbers within Lévy distribution. The authors tested the performance of this algorithm on various standard benchmark test functions and concluded that Lévy PSO is performed as well, or better, than a standard PSO or equivalent Gaussian models over a range of benchmark problems.

Krohling R.A. and Mendel E. [74] presented the bare bones particle swarm optimization (BBPSO) algorithm with jump strategy to overcome the local optima problem in high dimensional search space. The objective of this approach is to allow particles of the swarm escape from local minima when they are prematurely attracted to a local attractor. The authors implemented the jump strategy with standard BBPSO based on Gaussian or the Cauchy probability distribution, when no fitness improvement is observed in desired objective function, and concluded that the introduced jump strategy increases the chances to escape from local minima, therefore improving the effectiveness of the algorithm.

Omran M.G.H. *et al.* [75] presented the paper on bare bones differential evolution (BBDE). The author proposed the approach, which combined concept of BBPSO and recombination operator of DE, differential evolution is used to mutate, for each particle, the attractor associated with that particle, defined as a weighted average of its personal and neighborhood best positions and BBPSO is used to update particle position probabilistically. The authors said that the resulting algorithm, referred to as the bare bones differential evolution (BBDE) eliminates the control parameters of PSO and replaces the static DE control parameters with dynamically changing parameters to produce an almost parameter-free, self-adaptive, optimization algorithm.

Chen C.H. [76] presented the paper on bare bones particle swarm optimization (BBPSO) in which a new method combining the global and local learning strategy is used to improve the performance of BBPSO. The authors proposed two new variants of modification in standard BBPSO; the first one uses a fix combination scheme and the second uses adaptive combination scheme, to improve the performance of BBPSO. The author tested this two algorithms on standard benchmark test function (unimodal, multimodal), to justify the feasibility of the strategy and conclude that the proposed algorithms do not impose much computation burden and is as easy to implement as the original bare bone particle swarm optimization.

Blackwell T. [77] presented the paper on study of collapse in bare bones particle swarm optimization (BBPSO). The author said that Collapse is possible swarm pathology: particles approach each other faster than the swarm as a whole approaches a local optimum which resulted convergence toward a limit point becomes exponentially slow and the swarm stagnates. The author formulated the dynamic update rule of particle swarm optimization as a second-order stochastic difference equation, general relations are derived for search focus, search spread, and swarm stability at stagnation and this relation is applied to BBPSO to derived no-collapse condition. The author concluded that BBPSO situated at the edge of collapse is comparable to other PSOs, and that performance can be still further improved with the use of an adaptive distribution and overall no collapse condition is derived

On basis of literature survey the conclusion drawn are summarized as follows:

The meta-heuristic algorithm such as tabu search, simulated annealing (SA), GA, DE, PSO when used on filter design problem gave satisfactory results. As in the case of SA, it is obvious from [78] that it has strong ability of jumping out a local optimum solution, but its whole search ability is not strong. GA itself to be more efficient in terms of obtaining local optimum, but GA reported so far only deals with low to medium order FIR filters [79]. In order to overcome this disadvantage of the GA in numeric optimization problems, a differential evolution (DE) algorithm has been implemented to design of digital filters by various authors [44], [47] . The disadvantages of DE algorithm is extremely slow [47], involved large number of control parameters and depended much upon the initial population members . The other optimization algorithm known as PSO is implemented by several researchers to design the FIR filters due to its simple algorithm structure and can be controlled by little parameters But the limitation of PSO is that it may be influenced by premature convergence and stagnation problem. Due to this several modifications in conventional PSO are proposed and implemented in FIR filter design by various researchers [63], [62] .

This literature survey shows that there are many optimization techniques are used in digital FIR design problem. To the best of my knowledge, however, there is no filter design in literature which coefficients are optimized by bare bones particle swarm optimization (BBPSO). The BBPSO algorithm is proposed by the Kennedy as the simplest version of PSO by eliminating the velocity formula. BBPSO algorithm has advantages of almost parameter free tuning, best convergence speed, less execution time and implemented by few lines of computer code which make this algorithm simple and versatile.

Chapter3

PROBLEM DEFINITION

In digital filter design problem, the main objective is to develop a realizable approximation of a given magnitude response specification. Since impulse response of an ideal filter is non-casual and of infinite length, these ideal filters are not realizable. The magnitude response of the FIR filters are acquired by truncating the impulse response of ideal filter does not have a sharp band transition from pass-band to stop-band but, rather, exhibits a gradual roll-off [80].

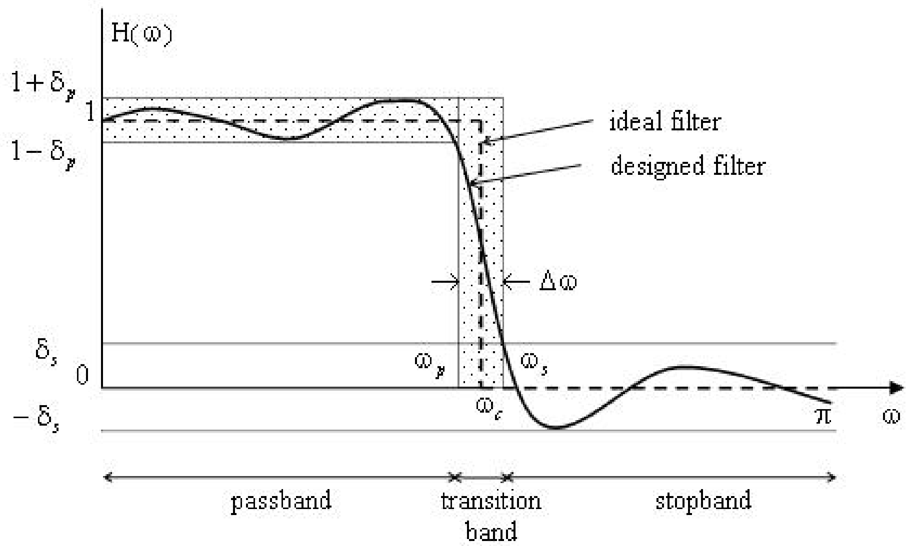


Figure 3.1: Typical magnitude specifications of digital lowpass filter

Thus , in case of digital filter design problem, the magnitude specification of digital filter pertains some accepting tolerance for pass band and stop band, also a band of transition is defined to drop off the magnitude smoothly between the pass-band and stop-band. The frequencies ω_p and ω_s are called pass-band edge frequency and stop band edge frequency respectively. The limit of tolerance used here are δ_p and δ_s which are called as the peak ripple values of pass band and stop band respectively. Another important parameter of digital filter design is minimum stop-band attenuation α_s and peak pass-band ripple α_p which are represented in dB. These are the loss specifications of the digital filters and represented by:

$$\alpha_p = -20\log_{10}(1-\delta_p)dB \tag{3.1}$$

$$\alpha_s = -20\log_{10}(\delta_s)dB \tag{3.2}$$

Filters design with exact linear phase is always possible in case of digital FIR filter case [80]. The FIR filter design is based on a direct approximation of the specified magnitude response, since it is always known the phase response of linear-phase FIR filter. An FIR filter is characterized by its impulse response $h(n)$ and this value of impulse response gave the type of filter. A casual FIR filter of degree N transfer function $H(z)$ is given by:

$$H(z) = \sum_{n=0}^{N} h(n)z^{-n}; \qquad n = 0, 1, \cdots, N \tag{3.3}$$

where, N is the order of the filter and $h(n)$ is the filter impulse response.

The frequency response of the FIR digital filter can be calculated as:

$$H(e^{j\omega_k}) = \sum_{n=0}^{N} h(n)e^{-j\omega_k n} \tag{3.4}$$

where, $\omega_k = \frac{2\pi k}{n}$ and $H(e^{j\omega_k})$ is the Fourier transform complex vector. The frequency is sampled in $[0, \pi]$ with N sampling points.

The least mean squared (LMS) error is used as an equation criteria fitness function in design of FIR filter problem. The algorithm process attempt to minimize this fitness error function (3.5) and thus optimizes performance of designed filter. The least mean squared error expression is given as:

$$E(\omega) = min \sum_{k=1}^{K} |H_i(e^{j\omega_k}) - H_d(e^{j\omega_k})|^2 (3.5)$$

where, $E(\omega)$ is the error sequence difference H_i denotes the ideal filter magnitude response which is expressed as:

$$H_i(e^{j\omega_k}) = \begin{cases} 1 & \text{for } 0 \leq \omega \leq \omega_c \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

$H_d(e^{j\omega_k})$is the desired magnitude response of designing filter,$H(e^{j\omega_k})$ is the filter magnitude response with a given set of filter coefficients and K is the sample number used for error caculation.

Chapter4

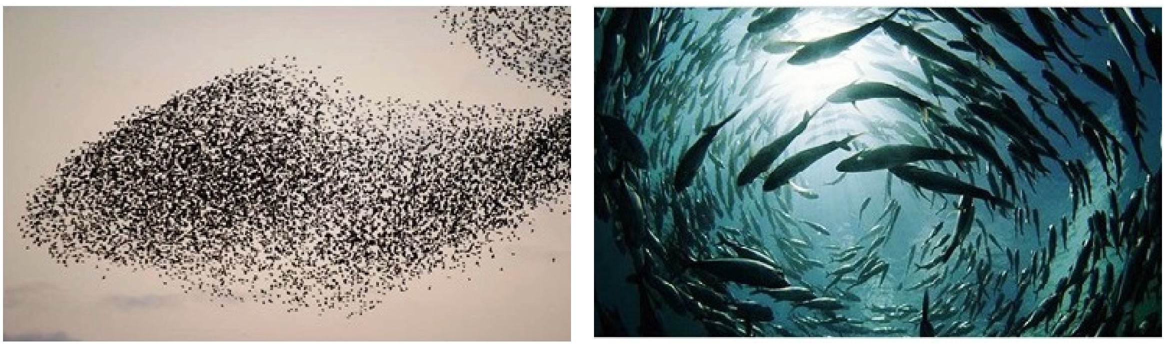
BBPSO ALGORITHM

Evolutionary algorithms and swarm intelligence based algorithms are meta-heuristics, which are stochastic, adaptive and results in an improved optimization approach. These algorithm approaches have capability to adjust itself according to changing environment and give better result by the help of some previously acquired information. In this section a swarm intelligence based algorithm known as bare bones particle swarm optimization (BBPSO) is discussed.

4.1 History

Particle swarm optimization (PSO) algorithm is introduced by Eberhart and Kennedy in 1995, is based on mimicking of some natural swarm social behavior which exist in nature such as: flocking behiviour of birds, schooling behavior of fish [19]. Here each particles are moved in search space to update their own best position and collectively find the swarm best position. After finding the position of global best position of swarm each individual particle in swarm update their own position to approach this best position and in whole process swarm is modified.

In the standard PSO model, each individual is defined as the volume-less particle



(a) Bird flocking **(b)** Fish schooling

Figure 4.1: Swarm behavior of birds and fish

(point) in the d-dimensional search space. The position of particles in the search process are the possible solutions and represented as $X_i = (X_{i1}, X_{i2} \cdots X_{id})$. The previous best position with best fitness value is called X_{lbest} of that particle and represented as $X_{lbest_i} = (X_{lbest_{i1}}, X_{lbest_{i2}} \cdots X_{lbest_{id}})$, after that, the swarm best position called as *gbest* is obtained, whose fitness value is the best among all individual fitness value of particles and represented as $X_{gbest} = (X_{gbest_1}, X_{gbest_2} \cdots X_{gbest_d})$. The particles in search space change their position in each iteration to achieve best position by some velocity and this velocity term is represented as $V_i = (V_{i1}, V_{i2} \cdots V_{id})$. So, in this algorithm each particle update their individual best position (X_{lbest}) and collectively update global best position (X_{gbest}) with the help of some previously defined fitness function, to provide the solution of optimization problem.

Mathematically, the velocity and position of particles are manipulated according to the following equations:

$$V_{ij}^{k+1} = \omega * V_{ij}^k + C_1 * r_1 * (X_{lbest_{ij}}^k - X_{ij}^k) + C_2 * r_2 * (X_{gbest_j}^k - X_{ij}^k) \tag{4.1}$$

$$X_{ij}^{k+1} = X_{ij}^k + V_{ij}^{k+1} \tag{4.2}$$

where $\omega \in [0.4, 0.9]$ is the inertia weight.C_1 and C_2 are two positive constants. r_1 and r_2 are two random numbers in range of $[0, 1]$. k denotes the current iteration.

The particle velocities must be in the range of $[V_{max}, V_{min}]$ on each dimension for any problem [81].

The main steps of the particle swarm optimization are summarized as follows:

1. Initialize each particle vector of the swarm by assigning random positions and velocities in d-dimensional problem search space.

2. Calculate fitness value of each particle according to the given objective function.

3. Calculate fitness value of each invidual and respective postions is set as X_{lbest}. If the current position fitness value is better than X_{lbest} fitness, then update the current position as X_{lbest}

4. Set the X_{gbest} position of swarm by taking best fitness function position among all fitness of particles in search space.

5. Update velocity and position of particle according to equation 4.1 and 4.2 respectively.

6. Repeat steps (2) to (5) until the maximum iteration or minimum error criteria is attained.

Since, PSO algorithm is likely converge around their weighted mean position of local best and global best position [23], it is not possible to obtain global optimum accurately with fast convergence speed. Therefore, a new approach is searched which can overcome these problems and provide solutions in multi dimensional search space with better convergence. These all conditions are satisfied by using the swarm based algorithm known as bare bones particle swarm optimization (BBPSO).

4.2 Mathematical modeling

Bare bones particle swarm optimization (BBPSO) was proposed by Kennedy in 2003, a Gaussian distribution with the information of local best and global best positions is used for updating the particle's positions [82].

Compared to the traditional PSO, BBPSO is probably the simplest version PSO since it does not required the velocity vector as well as inertia weight. BBPSO shows the property of self adapting in optimization process and because of few control parameter, it becomes simple in implementation. It has shown good performance in solving optimization problems in different areas such as integer programming problems [83], unsupervised image classification [84] and the static multi-objective economic dispatch problem [85].

BBPSO is originally derived to study the distribution of particles in conventional PSO model. After studied the model of standard PSO, it shows that particle is converged towards its personal best (X_{lbest}) and global best positions (X_{gbest}). But the work proposed by [86], [23] indicates the particle in the search space are more likely to converged towards weighted average of its local and global best positions as shown in equation 4.3.

$$\lim_{d \to \infty} X_{id} = \frac{C_1 * X_{lbest_{id}} + C_2 * X_{gbestd}}{C_1 + C_2} \tag{4.3}$$

The equation 4.3 shows that particles are structured in cyclic manner in search space and position of center is the, average of particles local best (X_{lbest}) and global best (X_{gbest}) positions. Considering this converging pattern of particles, Kennedy introduced a new algorithm named as BBPSO, by removing velocity vector. Here Gaussian random number is used in place of normal random number to overcome the above PSO model converging problem and position of particles are updated with the help of this Gaussian random number.

The BBPSO algorithm simplified the position update formula, and position update is performed by using Gaussian distribution of the mean $\frac{X_{gbest}+X_{lbest}}{2}$ and standard deviation of $|X_{gbest} - X_{lbest}|$. Mathematically, BBPSO update the position according to following equations:

$$X_{ij}^{k+1} = N(\mu, \sigma) \tag{4.4}$$

with,

$$\mu = 0.5(X_{gbest_j} + X_{lbest_{ij}}) \tag{4.5}$$

$$\sigma = |X_{gbest_j} - X_{lbest_{ij}}| \tag{4.6}$$

Where, $N(\mu, \sigma)$ is a Gaussian random number with mean of μ and deviation of σ.

The bare bones particle swarm optimization (BBPSO) steps is summarized as follows:

1. Set the number of particle n and maximum iteration count k.

2. Initializes particles X_i, $i = 1, 2, \cdots, n$.

3. For $i = 1, 2, \cdots, n$, evaluate X_i and set $X_{lbest_i} = X_i$.

4. Set X_{gbest}= best of X_i and set $k = 1$.

5. For $i = 1, 2, \cdots, n$ do steps 6 to 11.

6. For $j = 1, 2, \cdots, d$ do steps 7 to 8.

7. Compute μ and σ using equation (4.5) and (4.6)
   i.e $\mu = 0.5(X_{gbest} + X_{lbest}), \sigma = |X_{gbest} - X_{lbest}|$

8. Evaluate particle positions X_{ij} by equation (4.4)
   i.e. $X_{ij} = N(\mu, \sigma)$
9. If $f(X_{ij}) < f(X_{lbest_{ij}})$ then $X_{lbest_{ij}} = X_{ij}$
10. If $f(X_{ij}) < f(X_{gbest_j})$ then $X_{gbest_j} = X_{ij}$.
11. Set iteration count $k = k + 1$.
12. Repeat steps 2 to 11 until the minimum fitness error or maximum iteration is achieved.
13. Best solution output X_{gbest} is obtained.

Chapter5

SOLUTION APPROACH

In order to investigate the performance of BBPSO algorithm, it is applied on some standard test functions. Two types of test functions are used i.e. unimodal and multi-modal. To optimize these problems effectively, the dimension size, swarm size and number of iteration are kept fixed and their respective values are given in table 5.5. The unimodal and multi-modal function with its initialization range are specified in table 5.1, 5.2, 5.3. The objective is to formulate the both algorithms on test functions and find global corresponding global optimum solutions.

This work describes an algorithmic process named as BBPSO, to design the linear phase low-pass FIR filter. In this algorithm filter with positive symmetric coefficient $h(n)$ is optimized by an iterative process, and each iteration filter coefficients $h(n)$ is updated according to fitness of previous coefficients. Based on error fitness function the result is iteratively improved until the minimum error criteria or maximum iteration is obtained. The property of digital FIR filter ensures the symmetry property in even order FIR filter i.e. if the filter order N is taken as even, the obtained coefficients $h(n)$ is always symmetrical about central coefficients. This symmetry property reduces the dimensions of filter design problem from dimension N to $\frac{N}{2} + 1$. These $\frac{N}{2}+1$ coefficients are optimized and after that final coefficients are obtained by flipping these coefficients and then concatenated to obtain final coefficients.

The 20^{th} and 30^{th} order low pass filters are designed and optimized its coefficients using BBPSO algorithm. By using the symmetry property of coefficients in even order FIR filter, the dimension of design problem is reduced to 11^{th} and 16^{th} order and after the optimizing these reduced order filter coefficients, the original coefficients are obtained by performing flipping and concatenation operation in these reduced coefficients. The flow chart of FIR filter design using BBPSO algorithm is shown in

Fig. 5.1. The desired low-pass FIR filter parameters are specified in Table 5.4. Here the PSO algorithm is used only for comparison purpose. BBPSO and PSO algorithm Control parameters value of PSO and BBPSO algorithm used in this work are specified in table 5.5.

Table 5.1: Unimodal benchmark test functions and its value range

Functions	Formula	Range	x_{max}	Global Optimum
f_1	$\sum_{i=1}^{n} x_i^2$	$[-100, 100]$	100	0
f_2	$\sum_{i=1}^{n} \left[100(x_{i+1} - x_i)^2 + (x_i - 1)^2\right]$	$[-10, 10]$	10	0
f_3	$exp\left(0.5 \sum_{i=1}^{n} x_i^2\right) - 1$	$[-1, 1]$	1	0
f_4	$\sum_{i=1}^{n} \mid x_i \mid + \prod_{i=1}^{n} \mid x_i \mid$	$[-100, 100]$	100	0
f_5	$\sum_{i=1}^{n} i x_i^2$	$[-100, 100]$	100	0

Table 5.2: Multimodal benchmark test functions

Function name	Mathematical expression
Ackley function	$-20exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos 2\pi x_i\right) + 20 + e$
Rastrigin function	$\frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
Griewank function	$\sum_{i=1}^{n}\left[x_i^2 - 10\cos(2\pi x_i) + 10\right]$

Table 5.3: Multimodal benchmark test functions and its value range

Function name	Range	x_{max}	Global optimum
Ackley function	$[-100, 100]$	100	0
Rastrigin function	$[-100, 100]$	100	0
Griewank function	$[-100, 100]$	100	0

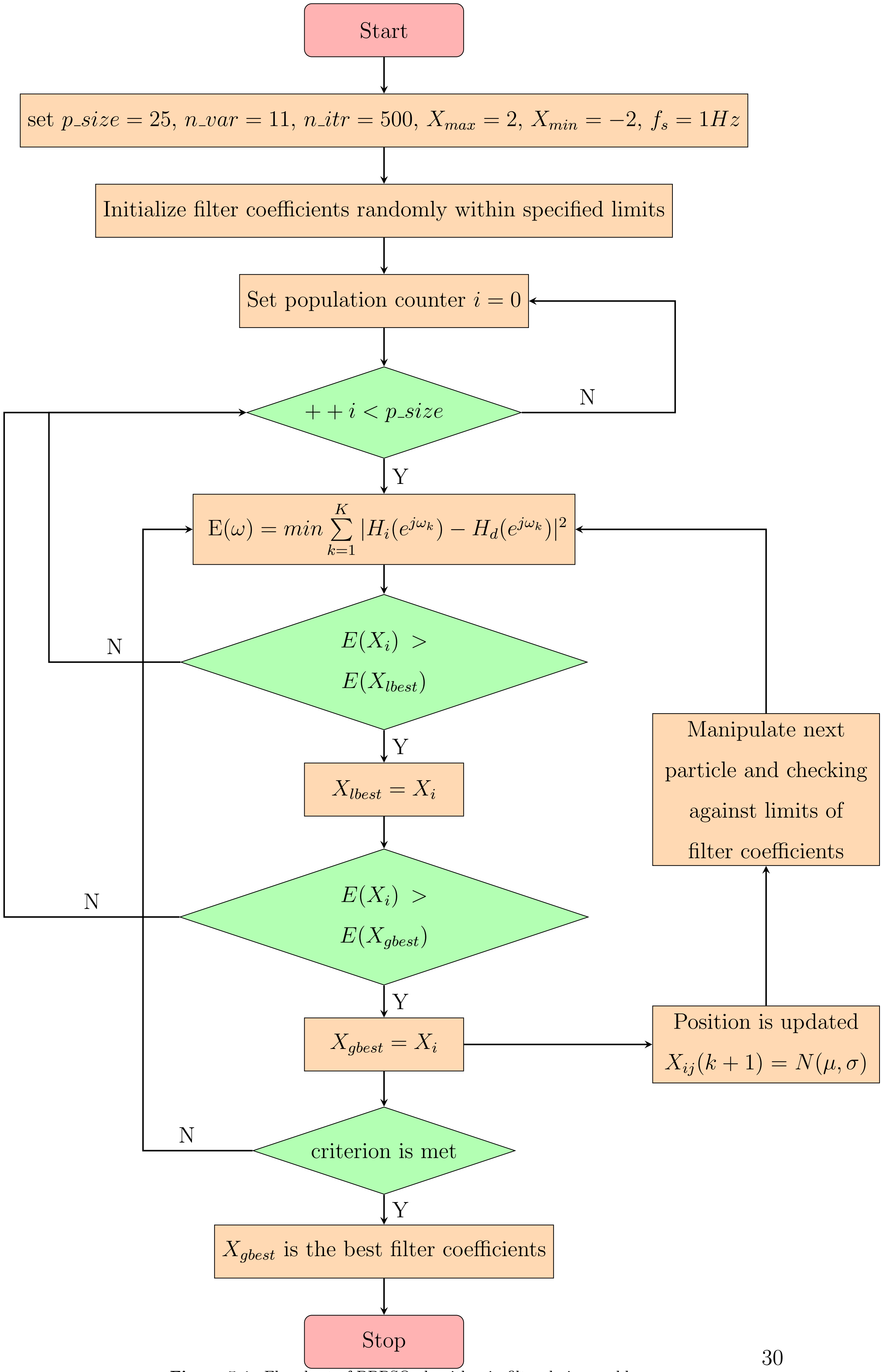


Figure 5.1: Flowchart of BBPSO algorithm in filter design problem

Table 5.4: FIR filter parameters

Filter parameters	Value
sampling frequency f_s	$1Hz$
number of frequency samples	128
pass band(normalized) edge frequency(ω_p)	0.45
stop band(normalized) edge frequency(ω_s)	0.55

Table 5.5: PSO and BBPSO parameters

Parameter	BBPSO	PSO
inertia weight (ω)	–	$[0.4, 0.9]$
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

Chapter6

RESULTS AND DISCUSSION

6.1 Test function optimization analysis

The BBPSO algorithm efficiency is tested by some standard benchmark test functions, which is defined in table 5.1, 5.2. With same dimension and iteration value BBPSO algorithm is simulated first on unimodal test functions and result are drawn. The result in table 6.1 is the best fitness value obtained after multiple trials, which reveal that BBPSO result outperforms to that of PSO in unimodal type testing environment and it is clear that BBPSO gave the final solution much accurately. Table 5.2 depicts multi-modal test function, these functions contain several local minimum, and leads to stagnation at local minimum. In this case also the BBPSO gave better result than PSO with same control parameter, its optimum value is recorded in table 6.2. Its convergence speed towards global optimum is high and gave result in much less time.

Table 6.1: Performance comparison of unimodal test functions

Function	PSO	BBPSO
f_1	$5.673E-9$	$3.091E-33$
f_2	3.04	0.0231
f_3	$9.920E-15$	$2.2238E-16$
f_4	$1.667E-3$	$8.600E-21$
f_5	$13.454E-9$	$9.763E-33$

Table 6.2: Performance comparison of multimodal test functions

Function name	PSO	BBPSO
Ackley function	20.7041	$7.9936E-15$
Rastrigin function	7.8612	1.5430
Griewank function	2.3125	0.9902

After testing the proposed algorithm on several benchmark test functions it is conformed that BBPSO algorithm has ability to solve unimodal as well as multi modal problems in multi-dimensional search space with equally efficiency and produce best result with accuracy and converge quickly towards true global optimum.

6.2 Magnitude response analysis of designed filter

In this work, BBPSO algorithm is used to optimize the low pass FIR filter coefficients of 20^{th} and 30^{th} order. The algorithm control parameter is set according to table 5.5 and desired filter specification is described in Table 5.4. The desired linear phase low pass FIR filter is designed using proposed algorithm and its various parameters are discussed also a comparison with that of PSO is made.

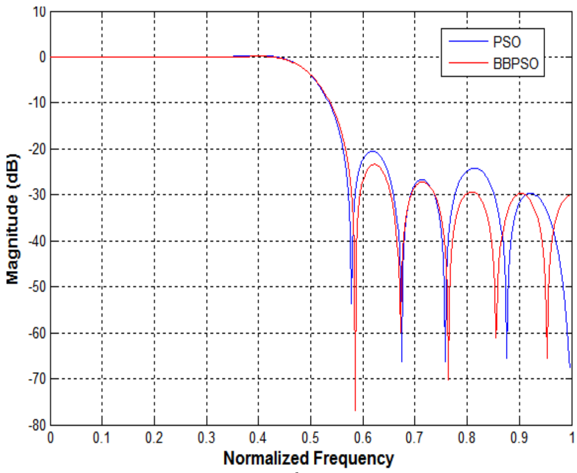


Figure 6.1: dB plot of 20^{th} order low pass FIR filter.

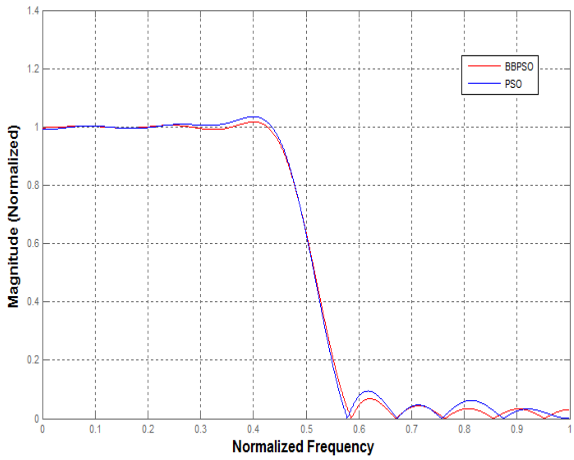


Figure 6.2: Magnitude plot of 20^{th} order low pass FIR filter.

Fig. 6.1 shows the gain plot in dB of 20^{th} order low-pass FIR filter using PSO and BBPSO algorithms individually and it is sure that maximum attenuation in stop band is achieved in case of BBPSO. Fig. 6.2 shows the normalized magnitude plot of 20^{th} order low pass FIR filter. Fig. 6.3 and Fig. 6.4 shows the pass band and stop band ripples present in designed filter respectively using PSO and BBPSO. As seen from the fig. 6.1 and 6.2 it is clear that filter designed by the BBPSO algorithm have sharper transition band response, whereas Fig. 6.2 depicts the ripple information in both frequency band. It is further verified by fig. 6.3 and Fig. 6.4 which shows the normalized pass band and stop ripples of 20^{th} order low pass filter, which lead the very low ripples in pass-band and also low-ripples in stop band is achieved using BBPSO and this value more less than that of achieved by PSO.

The best coefficients of obtained from the filter of 20^{th} order design have been

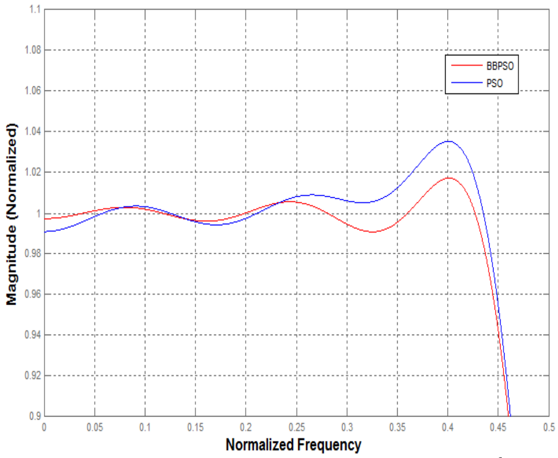


Figure 6.3: Normalized pass band ripple plot of 20^{th} order low pass FIR filter.

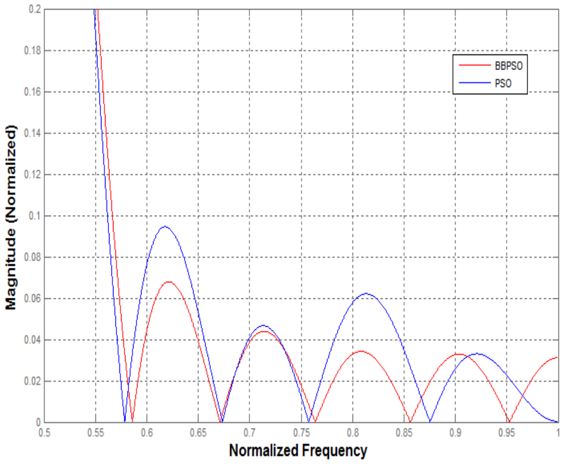


Figure 6.4: Normalize stop band ripple plot of 20^{th} order low pass FIR filter.

Table 6.3: Optimized coefficients of FIR filter of order 20

$h(N)$	PSO	BBPSO
$h(1) = h(21)$	-0.014236594744891	-0.023617425437406
$h(2) = h(20)$	0.019385491291270	0.033527644540311
$h(3) = h(19)$	0.011744420796837	0.013616454475978
$h(4) = h(18)$	-0.032406457857607	-0.046644680202339
$h(5) = h(17)$	-0.012281907140211	-0.007403889722164
$h(6) = h(16)$	0.054287624855255	0.065157332070242
$h(7) = h(15)$	0.012514590017345	0.003763283449318
$h(8) = h(14)$	-0.100317066828962	-0.107333883786525
$h(9) = h(13)$	-0.012774394728642	-0.001615562492581
$h(10) = h(12)$	0.316181719233667	0.318948014614085
$h(11)$	0.513194574284170	0.500584271517688

Table 6.4: Designed 20^{th} order filter ripple and attenuation value comparison

Filter specification	PSO	BBPSO
Maximum pass band ripple (normalized)	0.03533	0.01713
Maximum stop band ripple (normalized)	0.0942	0.0671
Peak pass band ripple (α_p)	$0.3124dB$	$0.1500dB$
Minimum stop band attenuation (α_s)	$20.5189dB$	$23.4655dB$
Maximum stop band attenuation	$66.39dB$	$76.97dB$

found by the two methods (PSO, BBPSO) and recorded in Table 6.3. The magnitude plot itself shows that BBPSO method of filter design gave better coefficient than that designed by PSO method. the filter specification obtained by these two methods are recorded in Table 6.4. This obtained specification shows that filter designed by BBPSO method reduces the ripples in stop band as well as pass band very efficiently and gives 23.4655dB and 76.97dB as minimum and maximum stop band attenuation in stop band. The pass band normalized ripples and peak pass band ripple of designed filter is also reduced to very low value as 0.01713 and 0.1500dB respectively.

Next, simulation is performed to realize the 30^{th} order FIR filter with same filter specifications and algorithms control parameters as described in Table 5.4 and Table 5.5.

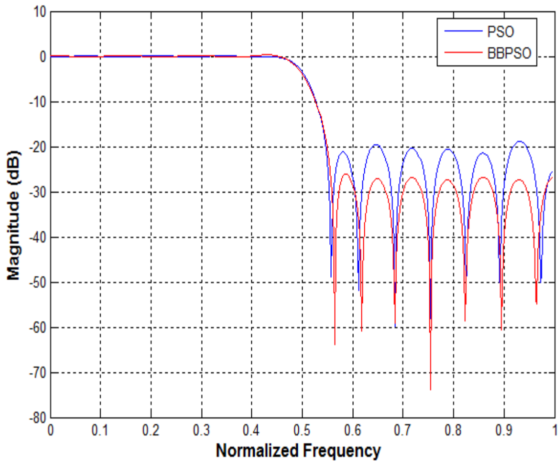


Figure 6.5: dB plot of 30^{th} order low pass FIR filter.

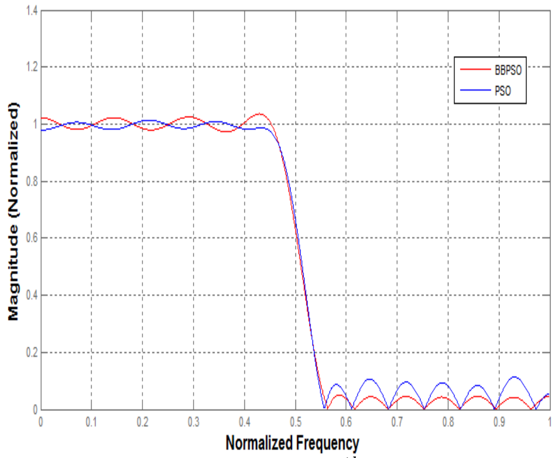


Figure 6.6: Magnitude plot of 30^{th} order low pass FIR filter.

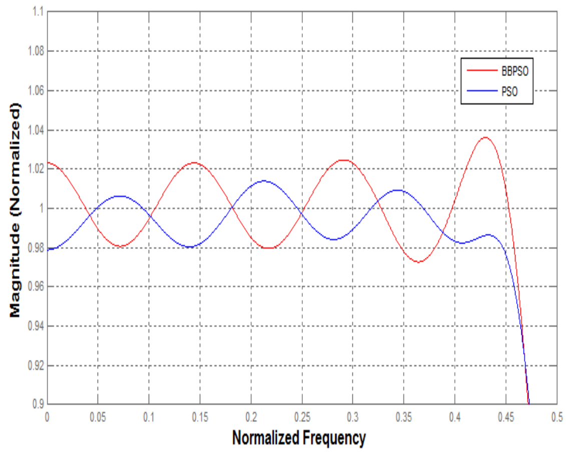


Figure 6.7: Normalized pass band ripple plot of 30^{th} order low pass FIR filter.

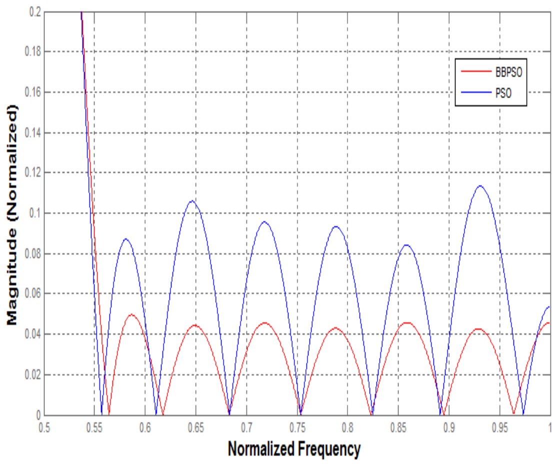


Figure 6.8: Normalize stop band ripple plot of 30^{th} order low pass FIR filter.

Here Fig.6.5 and Fig. 6.6 shows the corresponding dB plot and normalized magnitude plot of 30^{th} order low pass FIR filter respectively and these plots are obtained by optimizing the filter coefficients by each algorithm individually. Fig. 6.7 shows the normalized pass band ripple plot and Fig. 6.8 depicts the normalized stop band ripple plot of obtained FIR filter. As seen from Fig. 6.5 it is obvious that in stop band, BBPSO algorithm gives maximum attenuation, information from Fig. 6.8 conforms that the stop band peak ripple value is largely suppressed in BBPSO filter optimization process.

Table 6.5: Optimized coefficients of FIR filter of order 30

$h(N)$	PSO	BBPSO
$h(1) = h(31)$	0.007319545812418	0.004901365376514
$h(2) = h(30)$	-0.030151455581647	-0.00862704846293
$h(3) = h(29)$	0.021287518452502	0.020010486169365
$h(4) = h(28)$	0.007841194567636	0.0063528745525145
$h(5) = h(27)$	-0.021304988760221	-0.019403494999438
$h(6) = h(26)$	-0.006711285925364	-0.0076027918236145
$h(7) = h(25)$	0.026558297766530	0.026141519587660
$h(8) = h(24)$	0.008363909210574	0.0086897844640497
$h(9) = h(23)$	-0.037788016829386	-0.037747164859281
$h(10) = h(22)$	-0.007366341002840	-0.0095255022985486
$h(11) = h(21)$	0.054388994942789	0.057873496883405
$h(12) = h(20)$	0.011021208434931	0.01014985216479
$h(13) = h(19)$	-0.104271652095753	-0.102539657255485
$h(14) = h(18)$	-0.007689489291611	-0.0104634502225572
$h(15) = h(17)$	0.311903975140424	0.3180121673954515
$h(16)$	0.512055633038889	0.510686831827831

Table 6.6: Designed 30^{th} order filter ripple and attenuation value comparison

Filter specification	PSO	BBPSO
Maximum pass band ripple (normalized)	0.01923	0.02310
Maximum stop band ripple (normalized)	0.1124	0.04872
Peak pass band ripple (α_p)	$0.1686dB$	$0.2029dB$
Minimum stop band attenuation (α_s)	$18.9846dB$	$26.2458dB$
Maximum stop band attenuation	$51.38dB$	$73.79dB$

The best coefficients obtained from the 30^{th} order FIR low pass filter design by using both the algorithms are illustrated in Table 6.5. The obtained property of this filter is recorded in Table 6.6, which shows both algorithm produces the similar responses. But, in the stop band case the BBPSO, produces minimum and maximum stop band attenuation as 26.2458dB and 73.79dB respectively which is much lower than PSO.

from the Table 6.4 and Table 6.6,it is clear that the performance of BBPSO is better than PSO in FIR filter design problem irrespective of filter order and optimize the coefficients in efficient way.

6.3 Convergence profiles of BBPSO and PSO

To compare the effectiveness of BBPSO and PSO algorithms in terms of the error fitness value, the minimum fitness error value is plotted against iteration value. Fig. 6.9 and Fig. 6.10 shows the convergences of fitness error value plot which indicates the performance of both the algorithm, employed to optimize the coefficients of 20^{th} order and 30^{th} order low-pass FIR filter respectively.

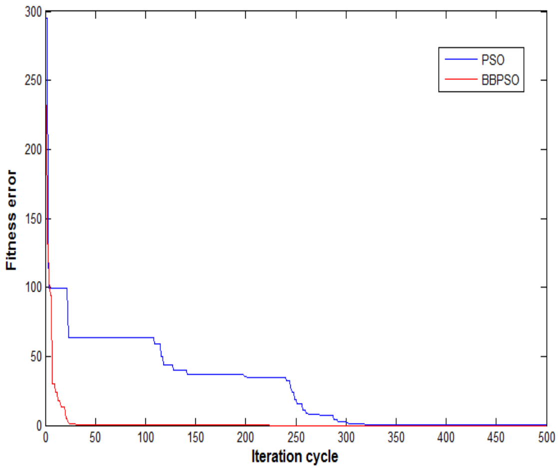


Figure 6.9: Convergence profile of PSO and BBPSO in case of 20^{th} order low-pass FIR filter design.

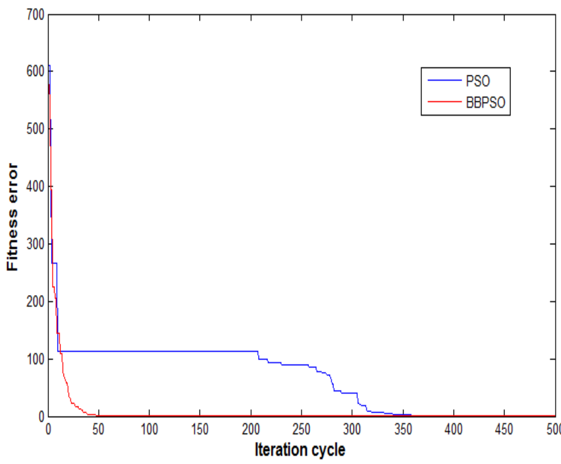


Figure 6.10: Convergence profile of PSO and BBPSO in case of 30^{th} order low-pass FIR filter design

It can be observed that both PSO and BBPSO converge to the same fitness but BBPSO converges to a much lower fitness in lesser number of iterations. From FIG and FIG it is observed that BBPSO algorithm converge quicky in both the cases but in case of PSO, convergence profile is affected and take more iteration to converge also PSO stagnation problem is revealed in both the figures. The PSO converges to the minimum error fitness value of 0 in 5 seconds whereas the BBPSO converges to the minimum error fitness value of 0 in 3 seconds in 20^{th} order filter design case and converges to 0 in 5 second in 30^{th} order case. For the prosed low-pass filter design problem, the BBPSO is converged the least minimum error fitness value in finding the optimum filter coefficients with the lesser number of iteration cycles and execution time. With a view to the above fact, the use of BBPSO over PSO is thus justified by the results of the experiment run in lesser number of iterations, where BBPSO seems to show better convergence and consistency than PSO. It is seen that when filter order is increased the PSO convergence rate becomes slow but there is almost no effect on the convergence behavior of BBPSO. This implies that BBPSO algorithm is more fit to design the filters with higher orders.

It may finally be inferred that the performance of BBPSO algorithm in FIR filter design, is better as compared to PSO. All optimization programs are written in MATLAB 7.10 version on core (TM) i3 processor, 2.10 GHz with 2 GB RAM.

Chapter7

CONCLUSION AND FUTURE SCOPE

In an FIR filter design problem the objective is to obtain filter coefficients as per desired specifications. In this work design of linear phase low pass FIR filter is presented. The BBPSO is modified form of canonical PSO. To investigate the behavior of BBPSO based filter design approach, experiments are performed on standard benchmark test functions and filter design problem.

Two cases of filter design with filter order 20^{th} and 30^{th} have been realized using BBPSO based solution approach. In order to compare the performance, the PSO algorithm is also simulated. The comparison of result shows that BBPSO algorithm minimize the pass band and stop band ripples efficiently, and gives better magnitude response. it is observed that convergence speed of BBPSO is better and gives best optimized result in very less time irrespective the dimension of problem. The simulation result also show that when order of filter increases the PSO result are affected but at the same time BBPSO gives the best result. So, BBPSO is adequate enough for handling other related design problems.

The future scope of present work is to include the other design limits such as ripple constraint and finite word length effect. The consequence can be considered as an additional constraint to objective function. The other directions of future scope is to investigate the behavior of BBPSO variants for better solution quality.

Bibliography

[1] J. McClellan, T. Parks, and L. Rabiner, "A computer program for designing optimum fir linear phase digital filters," *IEEE Transactions on,Audio and Electroacoustics*, vol. 21, pp. 506–526, Dec 1973.

[2] A. Oppenheim, R. Schafer, and J. Buck, "The discrete fourier transform," *Discrete-time signal processing. Upper Saddle River, NJ: Prentice-Hall*, pp. 541–600, 1999.

[3] N. Karaboga and B. Cetinkaya, "Design of digital fir filters using differential evolution algorithm," *Circuits, Systems and Signal Processing*, vol. 25, no. 5, pp. 649–660, 2006.

[4] S. Mondal, D. Chakraborty, R. Kar, D. Mandal, and S. Ghoshal, "Novel particle swarm optimization for high pass fir filter design," in *2012 IEEE Symposium on,Humanities, Science and Engineering Research (SHUSER)*, pp. 413–418, June 2012.

[5] R. Kar, D. Mandal, S. Bardhan, and S. Ghoshal, "Optimization of linear phase fir band pass filter using particle swarm optimization with constriction factor and inertia weight approach," in *2011 IEEE Symposium on,Industrial Electronics and Applications (ISIEA),*, pp. 326–331, Sept 2011.

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

[7] B. Luitel and G. Venayagamoorthy, "Differential evolution particle swarm optimization for digital filter design," in *IEEE Congress on,Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence).*, pp. 3954–3961, June 2008.

[8] L. R. Rabiner, B. Gold, C. McGonegal, *et al.*, "An approach to the approximation problem for nonrecursive digital filters," *IEEE Transactions on,Audio and Electroacoustics*, vol. 18, no. 2, pp. 83–106, 1970.

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

[10] T. Parks and J. McClellan, "Chebyshev approximation for nonrecursive digital filters with linear phase," *IEEE Transactions on,Circuit Theory*, vol. 19, no. 2, pp. 189–194, 1972.

[11] H.-C. Lu and S.-T. Tzeng, "Design of arbitrary fir log filters by genetic algorithm approach," *Signal Processing*, vol. 80, no. 3, pp. 497–505, 2000.

[12] S. Ahmad and A. Antoniou, "Cascade-form multiplierless fir filter design using orthogonal genetic algorithm," in *2006 IEEE International Symposium on,Signal Processing and Information Technology,*, pp. 932–937, Aug 2006.

[13] S. Ahmad and A. Antoniou, "A multiobjective genetic algorithm for asymmetric fir filters," in *2007 IEEE International Symposium on,Signal Processing and Information Technology*, pp. 525–530, Dec 2007.

[14] H. A. Oliveira Jr, A. Petraglia, and M. R. Petraglia, "Frequency domain fir filter design using fuzzy adaptive simulated annealing," *Circuits, Systems and Signal Processing*, vol. 28, no. 6, pp. 899–911, 2009.

[15] H. Yi, "A new optimization algorithm based on improved simulated annealing and genetic algorithm," in *2011 International Conference on,E -Business and E -Government (ICEE)*, pp. 1–4, May 2011.

[16] D. Karaboga, D. Horrocks, N. Karaboga, and A. Kalinli, "Designing digital fir filters using tabu search algorithm," in *IEEE International Symposium on,Circuits and Systems, 1997. ISCAS '97., Proceedings of 1997*, vol. 4, pp. 2236–2239 vol.4, Jun 1997.

[17] K. Reddy, M. Bharath, S. Sahoo, S. Sinha, and J. Reddy, "Design of low power, high performance fir filter using modified differential evolution algorithm," in *2011 International Symposium on,Electronic System Design (ISED)*, pp. 62–66, Dec 2011.

[18] G. Samanta and A. Chandra, "A novel design strategy of low-pass fir filter using opposition-based differential evolution algorithm," in *2012 IEEE Students' Conference o,Electrical, Electronics and Computer Science (SCEECS),*, pp. 1–4, March 2012.

[19] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *IEEE International Conference on.,Neural Networks, 1995. Proceedings*, vol. 4, pp. 1942–1948 vol.4, Nov 1995.

[20] Z. You, W. Chen, G. He, and X. Nan, "Adaptive weight particle swarm optimization algorithm with constriction factor," in *2010 International Conference of,Information Science and Management Engineering (ISME)*, vol. 2, pp. 245–248, Aug 2010.

[21] W.-P. Huang, L.-f. Zhou, and J. xin Qian, "Fir filter design: frequency sampling filters by particle swarm optimization algorithm," in *2004. Proceedings of 2004 International Conference on,Machine Learning and Cybernetics*, vol. 4, pp. 2322–2327 vol.4, Aug 2004.

[22] D. Krusienski and W. Jenkins, "Particle swarm optimization for adaptive iir filter structures," in *Congress on,Evolutionary Computation, 2004. CEC2004.*, vol. 1, pp. 965–970 Vol.1, June 2004.

[23] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on,Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[24] Z.-H. Zhan, J. Zhang, Y. Li, and H.-H. Chung, "Adaptive particle swarm optimization," *IEEE Transactions on,Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, pp. 1362–1381, Dec 2009.

[25] D. Mandal, R. Kar, S. P. Ghoshal, *et al.*, "Digital fir filter design using fitness based hybrid adaptive differential evolution with particle swarm optimization," *Natural Computing*, vol. 13, no. 1, pp. 55–64, 2014.

[26] Z. Qigang, W. Wenqi, and Z. Shujing, "A new design of 2-d fir filter by mcclellan transformation," in *3rd International Conference on,Signal Processing, 1996.*, vol. 1, pp. 72–75 vol.1, Oct 1996.

[27] W. Yunlong, W. Shihu, and J. Rendong, "An extreme simple method for digital fir filter design," in *2011 Third International Conference on,Measuring Technology and Mechatronics Automation (ICMTMA)*, vol. 1, pp. 410–413, Jan 2011.

[28] X. Lai, "Projected least-squares algorithms for constrained fir filter design," *IEEE Transactions on,Circuits and Systems I: Regular Papers*, vol. 52, pp. 2436–2443, Nov 2005.

[29] T. Kaya and M. Ince, "The fir filter design by using window parameters calculated with ga," in *Decision and Control, 2009. ICSCCW 2009. Fifth International Conference on,Soft Computing, Computing with Words and Perceptions in System Analysis*, pp. 1–4, Sept 2009.

[30] W.-P. Huang, L.-f. Zhou, and J.-X. Qian, "Fir filter design: frequency sampling filters by particle swarm optimization algorithm," in *Proceedings of 2004 International Conference on,Machine Learning and Cybernetics, 2004.*, vol. 4, pp. 2322–2327, IEEE, 2004.

[31] H.-C. Lu and S.-T. Tzeng, "Design of arbitrary {FIR} log filters by genetic algorithm approach," *Signal Processing*, vol. 80, no. 3, pp. 497 – 505, 2000.

[32] D. Krusienski and W. Jenkins, "Adaptive filtering via particle swarm optimization," in *Conference Record of the Thirty-Seventh Asilomar Conference on,Signals, Systems and Computers, 2004.*, vol. 1, pp. 571–575 Vol.1, Nov 2003.

[33] R. Kar, D. Mandal, S. Bardhan, and S. Ghoshal, "Optimization of linear phase fir band pass filter using particle swarm optimization with constriction factor and inertia weight approach," in *2011 IEEE Symposium on,Industrial Electronics and Applications (ISIEA)*, pp. 326–331, Sept 2011.

[34] S. Mandal, P. Mallick, D. Mandal, R. Kar, and S. Ghoshal, "Optimal fir band pass filter design using novel particle swarm optimization algorithm," in *2012 IEEE Symposium on,Humanities, Science and Engineering Research (SHUSER)*, pp. 141–146, June 2012.

[35] S. Mondal, D. Chakraborty, R. Kar, D. Mandal, and S. Ghoshal, "Novel particle swarm optimization for high pass fir filter design," in *2012 IEEE Symposium on,Humanities, Science and Engineering Research (SHUSER)*, pp. 413–418, June 2012.

[36] S. Saha, A. Chaudhuri, D. Mandal, R. Kar, and S. Ghoshal, "Optimization of iir high pass filter using craziness based particle swarm optimization technique," in *2012 IEEE Symposium on,Humanities, Science and Engineering Research (SHUSER)*, pp. 401–406, June 2012.

[37] S. Saha, R. Kar, D. MANDAL, and S. Ghoshal, "Adaptive particle swarm optimization for low pass finite impulse response filter design," in *2013 International Conference on,Communications and Signal Processing (ICCSP)*, pp. 19–23, April 2013.

[38] A. Sarangi, R. K. Mahapatra, and S. P. Panigrahi, "Retracted: Depso and pso-qi in digital filter design," *Expert Systems with Applications*, vol. 38, no. 9, pp. 10966–10973, 2011.

[39] 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.

[40] A. Soleimani, "Combine particle swarm optimization algorithm and canonical sign digit to design finite impulse response filter," *Soft Computing*, vol. 19, no. 2, pp. 407–419, 2015.

[41] K. Boudjelaba, F. Ros, and D. Chikouche, "Potential of particle swarm optimization and genetic algorithms for fir filter design," *Circuits, Systems, and Signal Processing*, vol. 33, no. 10, pp. 3195–3222, 2014.

[42] D. Mandal, R. Kar, S. P. Ghoshal, *et al.*, "Digital fir filter design using fitness based hybrid adaptive differential evolution with particle swarm optimization," *Natural Computing*, vol. 13, no. 1, pp. 55–64, 2014.

[43] R. Kar, D. Mandal, S. Mondal, and S. P. Ghoshal, "Craziness based particle swarm optimization algorithm for fir band stop filter design," *Swarm and Evolutionary Computation*, vol. 7, pp. 58–64, 2012.

[44] N. Karaboga and B. Cetinkaya, "Design of digital fir filters using differential evolution algorithm," *Circuits, Systems and Signal Processing*, vol. 25, no. 5, pp. 649–660, 2006.

[45] A. Chandra and S. Chattopadhyay, "Differential evolution based design of multiplier-less fir filter using canonical signed digit representation," in *2012 International Conference on,Communications, Devices and Intelligent Systems (CODIS)*, pp. 425–428, Dec 2012.

[46] S. Chattopadhyay, S. Sanyal, and A. Chandra, "Design of fir pulse-shaping filter: Superiority of differential evolution optimization over convex optimization," in *2010 5th European Conference on,Circuits and Systems for Communications (ECCSC)*, pp. 189–192, Nov 2010.

[47] K. Reddy, M. Bharath, S. Sahoo, S. Sinha, and J. Reddy, "Design of low power, high performance fir filter using modified differential evolution algorithm," in *2011 International Symposium on,Electronic System Design (ISED)*, pp. 62–66, Dec 2011.

[48] G. Samanta and A. Chandra, "A novel design strategy of low-pass fir filter using opposition-based differential evolution algorithm," in *2012 IEEE Students' Conference on,Electrical, Electronics and Computer Science (SCEECS)*, pp. 1–4, March 2012.

[49] K. S. Reddy and S. K. Sahoo, "An approach for fir filter coefficient optimization using differential evolution algorithm," *AEU-International Journal of Electronics and Communications*, vol. 69, no. 1, pp. 101–108, 2015.

[50] P. Upadhyay, R. Kar, D. Mandal, and S. Ghoshal, "Iir system identification using differential evolution with wavelet mutation," *Engineering Science and Technology, an International Journal*, vol. 17, no. 1, pp. 8–24, 2014.

[51] G. Liu, Y. Li, and G. He, "Design of digital fir filters using differential evolution algorithm based on reserved genes," in *2010 IEEE Congress on,Evolutionary Computation (CEC)*, pp. 1–7, July 2010.

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

[53] A. Lee, M. Ahmadi, G. Jullien, W. Miller, and R. Lashkari, "Digital filter design using genetic algorithm," in *1998 IEEE Symposium on,Advances in Digital Filtering and Signal Processing*, pp. 34–38, Jun 1998.

[54] D. Xu and M. Daley, "Design of finite word length fir digital filter using a parallel genetic algorithm," in *Southeastcon '92, Proceedings., IEEE*, pp. 834–837 vol.2, Apr 1992.

[55] Z. An-Xin, L. Ping, and L. Jian-jun, "The design of fir filter based on apa genetic algorithms," in *2011 International Conference on,Mechatronic Science, Electric Engineering and Computer (MEC)*, pp. 1118–1121, Aug 2011.

[56] S. Ahmad and A. Antoniou, "Cascade-form multiplierless fir filter design using orthogonal genetic algorithm," in *2006 IEEE International Symposium on,Signal Processing and Information Technology*, pp. 932–937, Aug 2006.

[57] K. Khamei, A. Nabavi, and S. Hessabi, "Design of variable fractional delay fir filters using genetic algorithm," in *Proceedings of the 2003 10th IEEE International Conference on,Electronics, Circuits and Systems, 2003. ICECS 2003.*, vol. 1, pp. 48–51 Vol.1, Dec 2003.

[58] S. Ahmad and A. Antoniou, "A multiobjective genetic algorithm for asymmetric fir filters," in *2007 IEEE International Symposium on,Signal Processing and Information Technology*, pp. 525–530, Dec 2007.

[59] F. Wysocka-Schillak, "Approximation of fir by iir filters using hybrid genetic algorithm," in *Signal Processing Algorithms, Architectures, Arrangements, and Applications (SPA), 2008*, pp. 149–154, Sept 2008.

[60] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *The 1998 IEEE International Conference on,Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence.*, pp. 69–73, May 1998.

[61] Y. Shi and R. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings of the 1999 Congress on,Evolutionary Computation, 1999. CEC 99.*, vol. 3, pp. –1950 Vol. 3, 1999.

[62] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on,Evolutionary Computation*, vol. 6, pp. 58–73, Feb 2002.

[63] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: simpler, maybe better," *IEEE Transactions on,Evolutionary Computation*, vol. 8, pp. 204–210, June 2004.

[64] L. Jin-Yue and Z. Bao-Ling, "The application of particle swarm optimization algorithm in the extremum optimization of nonlinear function," in *2012 IEEE 12th International Conference on,Computer and Information Technology (CIT)*, pp. 286–289, Oct 2012.

[65] I. C. Trelea, "The particle swarm optimization algorithm: convergence analysis and parameter selection," *Information processing letters*, vol. 85, no. 6, pp. 317–325, 2003.

[66] D. Tan, W. Luo, and Q. Liu, "Multi-objective particle swarm optimization algorithm for engineering constrained optimization problems," in *IEEE International Conference on,Granular Computing, 2009, GRC '09.*, pp. 523–528, Aug 2009.

[67] Z. You, W. Chen, G. He, and X. Nan, "Adaptive weight particle swarm optimization algorithm with constriction factor," in *2010 International Conference of,Information Science and Management Engineering (ISME)*, vol. 2, pp. 245–248, Aug 2010.

[68] L. Jin-Yue and Z. Bao-Ling, "The application of particle swarm optimization algorithm in the extremum optimization of nonlinear function," in *2012 IEEE 12th International Conference on,Computer and Information Technology (CIT)*, pp. 286–289, Oct 2012.

[69] X. Jun and H. Chang, "The discrete binary version of the improved particle swarm optimization algorithm," in *International Conference on,Management and Service Science, 2009. MASS'09.*, pp. 1–6, IEEE, 2009.

[70] G. Chen, "Simplified particle swarm optimization algorithm based on particles classification," in *2010 Sixth International Conference on,Natural Computation (ICNC)*, vol. 5, pp. 2701–2705, IEEE, 2010.

[71] Z.-H. Zhan, J. Zhang, Y. Li, and H. S.-H. Chung, "Adaptive particle swarm optimization," *IEEE Transactions on,Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 6, pp. 1362–1381, 2009.

[72] F. Pan, X. Hu, R. Eberhart, and Y. Chen, "An analysis of bare bones particle swarm," in *Swarm Intelligence Symposium, 2008. SIS 2008. IEEE*, pp. 1–5, Sept 2008.

[73] T. J. Richer and T. M. Blackwell, "The lévy particle swarm," in *IEEE Congress on,Evolutionary Computation, 2006. CEC 2006.*, pp. 808–815, IEEE, 2006.

[74] R. Krohling, E. Mendel, *et al.*, "Bare bones particle swarm optimization with gaussian or cauchy jumps," in *IEEE Congress on,Evolutionary Computation, 2009. CEC'09.*, pp. 3285–3291, IEEE, 2009.

[75] M. G. Omran, A. P. Engelbrecht, and A. Salman, "Bare bones differential evolution," *European Journal of Operational Research*, vol. 196, no. 1, pp. 128–139, 2009.

[76] C.-H. Chen, "A revised bare bone particle swarm optimizer and its variant," in *2013 International Conference on,Fuzzy Theory and Its Applications (iFUZZY)*, pp. 488–493, IEEE, 2013.

[77] T. Blackwell, "A study of collapse in bare bones particle swarm optimization," *IEEE Transactions o,Evolutionary Computation*, vol. 16, no. 3, pp. 354–372, 2012.

[78] H. Yi, "A new optimization algorithm based on improved simulated annealing and genetic algorithm," in *E -Business and E -Government (ICEE), 2011 International Conference on*, pp. 1–4, May 2011.

[79] M. Oner and M. Askar, "Incremental design of high complexity fir filters by genetic algorithms," in *Proceedings of the Fifth International Symposium on,Signal Processing and Its Applications, 1999. ISSPA'99.*, vol. 2, pp. 1005–1008, IEEE, 1999.

[80] S. K. Mitra, *Digital Signal Processing: a computer-based approach*, vol. 2. McGraw-Hill New York, 2013.

[81] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *The 1998 IEEE International Conference on,Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence.,*, pp. 69–73, May 1998.

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

[83] M. G. Omran, A. Engelbrecht, and A. Salman, "Barebones particle swarm for integer programming problems," in *2007 IEEE Swarm Intelligence Symposium*, 2007.

[84] M. Omran and S. Al-Sharhan, "Barebones particle swarm methods for unsupervised image classification," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, pp. 3247–3252, IEEE, 2007.

[85] Y. Zhang, D.-W. Gong, and Z. Ding, "A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch," *Information sciences*, vol. 192, pp. 213–227, 2012.

[86] F. Van den Bergh and A. P. Engelbrecht, "A study of particle swarm optimization particle trajectories," *Information sciences*, vol. 176, no. 8, pp. 937–971, 2006.