DESIGN OF FINITE IMPULSE RESPONSE FILTER USING CHAOTIC DIFFERENTIAL EVOLUTION ALGORITHM

A Dissertation submitted in fulfillment of the requirements for the degree

of

MASTER OF ENGINEERING

in

Electronic Instrumentation and Control Engineering

Submitted by

HARLEEN KAUR

(801551003)

Under the guidance of

Mr. NIRBHOWJAP SINGH

Assistant Professor, EIED,

2017

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)

DECLARATION

I hereby declare that the work which has been presented in the dissertation entitled "**Design of Finite Impulse Response Filter using Chaotic Differential Evolution Algorithm**", in partial fulfilment of requirements for the award of degree of Master of Engineering in Electronic Instrumentation and Control Engineering submitted in the Department of Electrical and Instrumentation Engineering at Thapar University, Patiala is an authentic record of my own work carried out under the supervision of **Mr. Nirbhowjap Singh**. It refers others research work which is appropriately recorded in reference section. The matter contained in this dissertation has not been submitted, neither partially nor in full to some other degree to whatever other college and organization aside from as detail in content and references.

Harleen Kaur

Harleen Kaur

Regd. No: 801551003

Place: Patiala

Date: 27 July 2017

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

Nirbhowjap

Mr.Nirbhowjap Singh

Assistant Professor, EIED

Thapar University, Patiala

Date: 27 July 2017

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 supervisor **Mr. Nirbhowjap Singh** (assistant professor), Electrical and Instrumentation department for doing everything from the inception of the work idea to giving incredible suggestions at every step.

I am also thankful to Dr. Ravinder Aggarwal, 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 batch-mates 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.

Finally, I would like to express my sincere and deep gratitude to my parents for their love, encouragement, care and support. They have always wanted the best for me and I admire their determination and sacrifice.

Harleen Kaur
801551003

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF ABBREVIATIONS

ABC	Ant Colony Optimization
ACO	Artificial Bee Colony
BBO	Biogeography Based Optimization
BIBO	Bounded Input and Bounded Output
CDE	Chaotic Differential Evolution
CPSO	Chaotic Particle Swarm Optimization
CR	Crossover Rate
CSA	Cuckoo Search Algorithm
CSE	Common Sub Expression
CSO	Cat Swarm Optimization
DE	Differential Evolution
DEPSO	Differential Evolution Particle Swarm Optimization
DFT	Discrete Fourier Transform
EA	Evolution Algorithm
ECG	Electrocardiogram
EMI	Electromagnetic Interference
F	Mutation Scale Factor
FIR	Finite Impulse Response
GA	Genetic Algorithm
HC	Hill Climbing
HDE	Hybrid Differential Evolution
HGA	Hybrid Genetic Algorithm
HTGA	Hybrid Taguchi Genetic Algorithm
IDFT	Inverse Discrete Fourier Trnasform
IIR	Infinite Impulse Response
LSE	Least Square Error
OBL	Opposition Based Learning
OHS	Opposition based Harmony Search

PDP	Power Delay Product
PLI	Power Line Interference
PM	Park Mcclellan
PSD	Power Spectral Density
PSO	Particle Swarm Optimization
PSO-QI	Particle Swarm Optimization with Quantum Infusion
PSOCFIWA-WM	Paricle Swarm Optimization with Constriction Factor aand Inertia Weight Approach hybridized with Wavelet Mutation
QPSO	Quantum behaved Particle Swarm Optimization
RGA	Real Coded Genetic Algorithm
RPSO	Random Particle Swarm Optimization
SA	Simulated Annealing
SNR	Signal to Noise Ratio
SPT	Signed Power of Two Term
TIA	taguchi Immune Algorithm
TS	Tabu Search

ABSTRACT

In this dissertation work, an approach to design a finite impulse response filter having linear phase has been proposed. A digital filter design procedure has to obtain such a set of filter coefficients that expected frequency specification of the signal are met. The proposed approach relies on chaotic differential evolution evolution algorithm (CDE) to generate the filter coefficients. The chaotic differential evolution (CDE) algorithm is a modification of conventional differential evolution (DE) algorithm, which depends on chaotic sequence for its performance improvement. A uniformly distributed random number generation is very important for better performance of stochastic algorithm. Chaotic map is similar to uniform random number generating function with better statistical and dynamic properties. The chaotic sequence provides better balance between the exploration and exploitation capability of the algorithm. The performance of proposed algorithm is analysed by experimentation with generalized benchmark problems. Low pass and high pass FIR filters of 20^{th} and 30^{th} order have been designed using CDE approach. Least square error is used as an objective function to measure the deviation of frequency response of designed filter from the ideal response. The ripple constraint is introduced in order to reduce the magnitude of ripples present in pass band and stop band regions of filter. The ripple constraint is taken care of using penalty method. The simulation results shows that proposed CDE approach with Tent map outperforms conventional DE algorithm in providing maximum stop band attenuation and minimum stop band and pass band ripples in frequency response. The application of CDE approach for filtering the power line interference (PLI) effected electrocardiogram (ECG) signal is also included in this research work. The experimental results have shown that the proposed approach has ability to design a filter as per desired specifications.

CHAPTER 1

INTRODUCTION

Filtering is the most extensively used process in every digital signal processing systems. The filtering process is used to modify, reshape and reform the signal in order to obtain desired spectral characteristics in the output signal. Basically, digital filters perform the mathematical operations on discrete time input signals to enhance or reduce specific features of the signal. The filtering operations attenuate, amplify, isolate or reject the certain frequency components present in the signal. The digital filters offer various advantages over the analog filters like high accuracy, ease of implementation, small in size, component drift free, component tolerance independent and high reliability [1]. The very main advantage provided by digital filters is that, in order to achieve any certain characteristic in the output signal, there is simply the change in the values of contents stored in registers [2]. Digital filters are highly adaptable due to the use of programmable processor, multiplexers and can access both real time and stored digital signals. The important fields of application of digital filters are image processing, speech processing, data communication, radar and optical communication *etc.* The digital filters found wide application in filtering the noise corrupted biomedical signals like electrocardiogram (ECG), electromyography (EMG), electroencephalogram (EEG) *etc.* [3–5]

The digital filters can be sorted into two categories namely infinite impulse response (IIR) and finite impulse response (FIR) filters [6]. FIR are widely accepted over IIR filters. The major merits offered by FIR filters are summarized as: FIR filters have finite impulse response, they are non-recursive in nature, have guarantee bounded input and bounded output (BIBO) stability *i.e* they are inherently stable filters, they have linear phase and are extensively used in applications where no phase distortion is required, unlike IIR, FIR filters are insensitive to finite word length effect [7]. BIBO stability and linear phase characteristics of FIR filter make it's realization process easy.

Traditional methods involve in designing FIR filters are window techniques, frequency sampling, equiripple design techniques *etc.* [6]. In window based designing process, finite length window is convolved with the impulse response of ideal filter. Windowing technique has advantage of fast and simplicity but does not provide exact control of critical frequencies. Frequency sampling method involves sampling the frequency response into equally spaced point using discrete fourier transform (DFT) and then obtaining the impulse response applying the inverse DFT algorithm. The frequency sampling approach provide better control of critical frequencies but the estimated error is zero only at the sampling frequencies [7]. Recently, the filter design problems are solved using search algorithms [8, 9]. The search algorithms are broadly classified into gradient search, direct search and random search methods. Conventional and gradient based methods used for the designing procedure generally result in suboptimal solution and become computationally complex when used for multi-objective and constrained optimization problems [10]. The failure of conventional methods to perform expeditiously have attracted many researchers to employ meta heuristic approaches in multi objective, non-differentiable and constrained engineering design problems. The main advantages offered by heuristic techniques are flexibility, robustness and fast convergence rate [11]. The most commonly used population based algorithms used to design digital filters are genetic algorithm (GA) [12], differential evolution (DE) [2], particle swarm optimization (PSO) [13], cat swarm optimization (CSO) [14] and simulated annealing (SA) [15]. GA is the most popular evolutionary algorithm but it is complex in coding, time consuming and sometimes results in premature convergence [13]. PSO has faster convergence rate than GA but the main limitation is that it results in premature convergence when applied on high dimensional optimization functions [16]. DE is another heuristic approach developed in 1995 by Storn R. and Price K. [17]. The fundamental concept of DE is the addition of weighted difference of two stochastically chosen parameter vectors to third vector. The created vector called mutated vector, goes through the process of improvement by using genetic base operations like crossover and selection. In the comparison study performed by Storn R. and Price K. [18], DE performed efficiently as compared to SA and GA. DE algorithm has been successfully employed to design FIR filters [19–21]. DE algorithm is found to be simpler, faster and requires lesser parameters than other evolutionary algorithms (EA).

Nevertheless similar to other EAs, DE also possess some limitations such as premature convergence and not assuring the global optimal solution. The performance of DE has been improved by introducing hybridization of DE with other algorithms [22], adaptive strategies with DE [23] and incorporating fuzzy logic with DE [24] *etc.* In the last few years, the application of non-linear dynamics has received intense interest in the field of optimization. The important application employs the chaos map sequence to determine the algorithm's control parameters [25]. In literature, chaotic sequence has been successfully employed to tune the control parameters of GA [26], PSO [27], ant colony optimization (ACO) [28], bat algorithm [29] *etc*. The application of chaotic sequence improves the algorithm's performance as it avoids stagnation in local minima and provides fast convergence. DE combined with the chaotic sequence has been successfully utilized in solving engineering optimization problems [30–32]. Experimental studies have shown that use of chaotic sequences provide better balance of exploration and exploitation capabilities of evolutionary algorithms.

In the light of above discussion, in the presented work chaotic differential evolution (CDE) algorithm is used to design 20^{th} and 30^{th} order linear phase low pass and high pass FIR filters. To investigate the performance of proposed approach, the results have been compared with other results available in literature. In order to accomplish this, the dissertation work is organized in 6 chapters. In the succeeding chapter a detailed literature review of various papers related to FIR filter designing using various optimization techniques has been presented. Chapter 3 describe FIR mathematical foundation of high pass and low pass FIR filter problem. Chapter 4 introduces the proposed CDE algorithm with sufficient mathematical background. The solution methodology is discussed in chapter 5. The comparison of results obtained is presented in chapter 6. Case study of designing the filter for noise removal from biomedical signal is discussed in chapter 7. Finally, Chapter 8 conclude the entire dissertation work.

CHAPTER 2

LITERATURE SURVEY

Digital filters are the essential part of every digital processing system. In order to, achieve the desired spectral specifications of the signal, number of digital filter design procedures have been discussed in literature. The conventional techniques like window technique,frequency sampling and equi-ripple design technique have been used for designing FIR filters [33]. The window technique is the simplest and most popular design technique. In the window technique based design procedure, the infinite response of ideal filter is delayed and truncated. Because of abrupt truncation of fourier series, the ripples are generated in response bands of the filter. The oscillatory behavior is due to the slow convergence of fourier series at the points of discontinuities and is known as Gibbs phenomenon [34, 35]. The effect of Gibbs phenomenon can be reduced by using different types of window functions such as hanning, hamming and kaiser [36]. The selection of window function highly depends on the filter requirements. The main limitation of window technique is that it does not permit precise control of critical frequencies [7]. Frequency sampling approach [37] involves the sampling of frequency response at equispaced points by using discrete fourier transform (DFT) and finally computing the filter coefficients by the means of inverse discrete fourier transform (IDFT). This approach provides better control of critical frequencies but the approximation error is zero only at the sampling frequencies [7]. Parks mcclellan (PM) method also called equiripple or minimax method, is based on chebyshev approximation problem [38]. This method gives ideal equiripple estimate of desired frequency response but at the cost of computational complexity. PM method does not allow exclusive selection of maximum magnitude of ripples in pass band and stop band. alternatively it only specify the ratio of pass band to stop band ripple [21].

Another approach for designing digital filter involves the use of L_1 norm in approximating the error function[39, 40]. Grossmann L.D. and Eldar Y.C [39] have used weighted L_1 norm with the modified newton method to achieve the optimal set of filter coefficients. Aggarwal *et al.* [41] have designed 34^{th} order high pass and band stop filter using L_1 norm as error criterion.The simulation results are compared with that obtained using minimax method. It is observed that the frequency response of L_1 filter has flatter pass band and stop band with maintained transition width.

On the other hand, gradient search methods are also used by many researchers to design digital filters [8, 9, 42]. Joaquim M.B and Lucietto C.A.S. [8] used the steepest descent optimization method in to design the digital filter. It is found that the algorithm can estimate any type of frequency response and it produces lower absolute and mean square error than produced by kaiser window and PM method of filter design. Although the steepest descent method is simple in implementation but computational time is large.

For the designing of the digital filter, precise control over various parameters of frequency spectrum is required. This makes the objective function to be highly non-uniform, non-differentiable, non linear and multi modal in nature [43]. The conventional design methods are proved unfit for designing as they generally achieve suboptimal solution and are sensitive to starting solutions. The gradient methods may not be suitable for designing because of two main reasons: firstly, there is the requirement of the objective function to be continuous and differentiable and secondly, they generally converge to local optimal solution [10].

Failure of conventional and gradient methods in attaining global optimum solutions has attracted many researchers to apply meta heuristic optimization techniques for designing problems. Meta heuristic optimization techniques are bio inspired algorithms that rely on natural process of evolution and selection [11]. The evolutionary algorithms (EA) are gaining important because of their flexible nature, robustness, fast convergence rate and minimum computation time. EA are widely accepted as selection procedure to solve many engineering design problems [44–46]. For the design of digital filter the most popular nature inspired algorithms employed are genetic algorithm (GA), differential evolution (DE), particle swarm optimization (PSO), artificial bee colony (ABC) and simulated annealing (SA) [18, 47–50] *etc.*.

Suckley D. *et al.* [12] reported the use of genetic algorithm to design low pass FIR filter. A comparison of performance between the filter designed using GA is made with that designed using the sequential algorithm of Wade *et al.* [51]. It is discovered that GA achieves the optimal solution with minimum computational complexity and is faster than sequential algorithm. Boudjelaba K. *et al.* [52] have proposed the hybrid genetic algorithm scheme to design FIR filter. The main limitation of standard GA is slow convergence. The author has presented the hybridization of standard genetic operation with the local search approaches. Four different variants of hybrid genetic algorithm (HGA) are generated. Each variant varies on basis of the type of local search procedure (tabu search (TS), hill climbing (HC) or SA) and chromosomes (reference or random) selected in the algorithm. HGA converges to optimal solution in least execution time. A comparison of simulation results with other traditional filter designing approaches is performed, thus it was concluded that filter designed using HGA has better frequency response than any other algorithms.

Ababneh J.I *et al.* [13] have designed finite impulse response filter employing PSO and GA algorithms. Design procedure involve two different design strategies. In first strategy, length of the filter, frequencies of passband and stop band and the ratio of the size of the passband and stop band ripples are considered. In the second one, along with the filter specifications mentioned in first strategy, the reasonable size of passband and stop band ripples are considered. PSO acquires the desired filter specifications both with finite and infinite word length filter coefficients.

Mandal S. *et al.* [1] presented the craziness based PSO called CRPSO algorithm for designing of high pass FIR filter. The craziness factor is introduced in PSO to remove the limitations of standard PSO. The craziness factor is used to describe the sudden change in the direction of fish or bird in fish schooling and bird flocking. It is used to maintain the diversity of the particles. High pass filter of 20^{th}, 30^{th} and 40^{th} order are realized using CRPSO and results are compared with that obtained using PM, RGA and PSO. It is found that the filter designed using CRPSO has highest stop band attenuation and lowest stop band ripples.

Fang W. *et al.* [53] discovered the quantum behaved PSO known as QPSO algorithm for the design of low pass and band pass FIR filers. The QPSO algorithm is designed with the combination of PSO with quantum modal. Each particle in QPSO is assumed to have quantum behavior. It is realized that QPSO outplays GA and standard PSO in designing filter. The QPSO found the global optimum solution in very less execution time and has faster convergence speed than GA and standard PSO.

Saha S. K *et al.* [54] proposed PSO based on inertia weight approach and constriction factor and hybridized with wavelet mutation (PSOCFIWA-WM) for designing the 20^{th} order low pass, band pass, high pass and band stop FIR filter. In PSOCFIWA-WM firstly, the velocity vector of standard PSO is modified by introducing the constriction factor and inertia weight factor to it. Secondly, the wavelet transform is used to mutate the velocity vector for generating the optimal solution. Comparison with PM, RGA, PSO, PSOCFIWA is performed and it is concluded that

PSOCFIWA-WM found the optimal solution in terms of frequency response and ripples in the bands of the filter. The solution is obtained in very less time than all other algorithms under consideration.

Luitel B. *et al.* [55] have used with quantum infusion based PSO (PSO-QI) to design 20^{th} order low pass IIR and FIR filter. The basic principle of PSO-QI is to strengthen the global best particle attained from standard PSO by performing tournament with the off springs generated by QPSO and to select the winner as the new global best particle in the algorithm. The simulation results are compared with standard PSO and QPSO. The PSO-QI took more time to converge to global optimum solution than PSO and QPSO but the algorithm achieves the lowest value of error fitness function of filter design.

Luitel B. and Venayagamoorthy G.K. [56] suggested the hybridization of PSO with differential evolution (DE) algorithm for designing the low pass filter. Hybrid of DE and PSO called DEPSO is used in two different ways. In first case, the error fitness function that is defined in PM algorithm, is used for filter design while in second case, fitness function is formulated as the mean squared mismatch between expected low pass filter and designed low pass filter's frequency response . From the results obtained, it is concluded that DEPSO converges to lower fitness value in less number of iterations than PSO.

Vasundhara *et al.* [43] presented the random particle swarm optimization (RPSO) algorithm hybrid with DE named random PSODE (RPSODE) to design FIR low pass and high pass filter. In order to overcome the shortcomings of standard PSO and DE like stagnation and sub-optimality, RPSODE is used. RPSO uses the weighted particle for randomly adapting the PSO. Weighted vector helps to maintain balance between the exploration and exploitation searches of algorithm. The simulation results show that RPSODE finds the best optimal filter coefficients in lesser time as compared to PSO, DE and PSODE.

Zhao Z. *et al.* [57] proposed the chaotic theory with standard PSO. The logistic map is used as the chaotic sequence generator. The proposed algorithm is called as chaotic particle swarm optimization (CPSO). The low pass and band pass FIR filters are designed using CPSO and results are compared with PSO and QPSO. It is observed that CPSO outperforms other algorithms in term of convergence speed and time.

Karaboga N. and Cetinkaya B. [2] have applied the DE for designing FIR filter. The low pass FIR filters of 8^{th}, 14^{th} and 20^{th} order are designed. The simulation results are compared with results obtained using GA and least square method (LSQ). It is found that DE has faster convergence, requires few control parameters and has simpler structure than GA. DE found the optimal filter coefficients much faster than GA. Karaboga N. and Cetinkaya B. [19] proposed the DE to the design the digital FIR filter with fixed point coefficients. The fixed point representation of coefficients improves the performance of filter by providing larger space for coefficients for same wordlengths. 20^{th} order low pass filter is designed and performance is compared with GA. The frequency response of filter designed with DE using fixed point coefficients shows the minimum deviation from desired response.

Liu G. *et al.* [58] proposed the modified DE algorithm based on reserved genes to design FIR filter. The eclectric differential evolution (eDE) scheme is used to generate the mutated vector. Two array vectors named elite and loser are used to store the best and worst individual of each generation respectively. Reserved genes of individuals of these two arrays produce new individual vectors by replacing with genes of other individuals in generation. This method increases the diversity of population. The author concluded that low pass filter designed with new eDE method has better frequency response than eDE and GA.

Sharma S. *et al.* [20] suggested the design of linear phase FIR filter using DE algorithm with ripple constraint. The ripple constraint handling approach is introduced in order to decrease the ripples magnitude in the frequency response of filter. Penalty function is used to introduce ripple constraint in the objective function. Typical weight vectors are assigned with the maximum pass band and stop band ripples value in the objective function. The simulation results are compared with the DE without ripple constraint approach and it is observed that filter designed with ripple constraint using DE has lesser ripples magnitude in stop band. Karaboga N. [59] proposed the DE algorithm for designing IIR filter. The DE performance is firstly evaluated on generalized benchmark functions. Secondly, DE is employed to design low order and high order IIR filters and it is observed that DE has faster convergence speed and require lesser computational time than GA.

Singh B. *et al.* [60] have proposed the hybrid DE algorithm in order to design infinite impulse response (IIR) filter. Ten different mutant variants are evaluated for designing IIR filter. Hybrid DE uses exploratory search method in order to perform global and local search together. Opposition based learning (OBL) technique is used, in which the opposite solution is checked in order to enhance the possibility of starting with better solution. Hybrid DE is used to design low-pass, high-pass, band-pass and band-stop IIR filters. The results of simulation are compared with genetic based algorithms *i.e.* hierarchical genetic algorithms (HGA), hybrid taguchi genetic algorithm (HTGA) and taguchi immune algorithm (TIA). It is found that hybrid DE satisfies the desired specifications in amplitude response consistently and is found superior to GA based algorithms.

Albataineh Z. *et al.* [21] presented hybrid differential evolution (H-DE) algorithm to design low pass FIR filter. H-DE algorithm uses the local search along with adaptive crossover method to improve the convergence rate of standard DE. The filter is designed for finite word length filter coefficients. The performance comparison of H-DE with GA and PSO shows that H-DE outperforms PSO by providing fast convergence.

Saha S. K. *et al.* [61] suggested the opposition based harmony search (OHS) algorithm to design linear phase FIR filters. In OHS algorithm, opposition learning approach is incorporated into standard harmony search (HS) algorithm. Opposition based learning (OBL) approach improves the convergence rate of HS by examining the opposite solution so that algorithm can start with the better solution. The low pass, band pass, high pass, and band reject filters of order 20 are designed using OHS and comparison is performed between PM, RGA, DE and PSO. Simulation results depicts that filter designed with OHS has better magnitude response with minimum magnitude of ripple in stop band and higher stop band attenuation. OHS retain better convergence in minimum execution time among all other algorithms under consideration.

Reddy K.S. and Sahoo S.K. [62] proposed the DE algorithm to find optimum set of filter coefficients for hardware efficient digital FIR filter design. The DE algorithm is used to find the set of coefficients with reduced number of signed power of two term (SPT). The hardware cost for adders is estimated using common sub expression (CSE) elimination algorithm. The filter is designed for various wordlengths and the performance is compared with other algorithms available literature in term of area, power, delay and power-delay-product (PDP). The proposed technique shows the PDP gain of 29%, which is maximum value attained than other algorithms based on remez algorithm.

Saha S.K *et al.* [14] presented the cat swarm optimization (CSO) algorithm to determine the optimal coefficients for FIR filters. CSO is applied to design 20^{th} order low pass, high pass, band pass and band reject FIR filters. The results of simulation are compared with the RGA, DE and

PSO. It is observed that CSO provides better convergence in less execution time than RGA, DE and PSO. The frequency response of filters designed with CSO meets the desired specifications by providing minimum magnitude of the ripple and higher magnitude of attenuation in stop band. Aggarwal A *et al.* [63] investigate the effectiveness of population based and swarm intelligence based algorithms in finding optimum FIR filter coefficients. The 20^{th} order low pass and band stop filters are designed using PSO, RCGA and cuckoo search algorithm (CSA). The simulation results shows that filter designed using CSA has flatter response in passband and higher stop band attenuation. CSA attains the best optimum coefficients in minimum execution time.

Panda R. and Pati U.C. [3] proposed the application of digital FIR filter for the removal of artifacts from ECG signals. The 100^{th} order low pass and high pass filters are designed using rectangular, hamming, hanning and balckman window. The peak signal to noise ratio is considered as the base for comparison. The observations made shows that the filter designed using rectangular window provide better attenuation in stop band and higher PSNR value than other windows. Mittal A. and Rege A. [64] presented the window based filtering technique for eliminating the PLI noise from ECG signal. The FIR filter is designed using kaiser, hamming, and chebyshev windowing methods. The SNR is calculated for noisy and filtered signal and it is observed that filtered signal obtained from filter designed using chebyshev window method has higher SNR value than any other window method.

Biswas U. and Maniruzzaman M. [65] presented the use of adaptive filter and notch filter for removing the power line interference (PLI) noise from ECG signals. The two types of adaptive filter's performance is compared with notch filter. The adaptive filters used are normalized least-mean-square (NLMS) and recursive-least-square (RLS) filters. The performance comparison parameters used are signal to noise ratio (SNR), mean square error (MSE), power spectral density (PSD) and percentage root mean square difference (%PRD). It is found that the filtered signal obtained using adaptive NLMS filter has higher SNR and lower MSE, %PRD values than other filters.

GA introduced by Goldberg D.E. and Holland J.H. in 1988 [47] is the most popular optimization technique. GA's principle is based upon the evolution of genes in living systems. The main steps require to generate the optimum solution are: initialization of population, selection of chromosome with lower fitness value, crossover with other chromosomes and mutation to produce diversity in search space. The GA is considered good optimizer for searching the favorable solution regions in search space. The major disadvantages of GA are slow convergence, trapping in local optimum solution when the number of parameters is large, complex in coding and requires more execution time [13, 66]. PSO was discovered by Kennedy and Eberhart in 1995 [48]. PSO algorithm is developed after inspiring from the social behavior of certain animals in the team like fish schooling, bird flocking *etc.* PSO has faster convergence speed than GA [13]. Although PSO is found to be efficient in searching the optimum solution for non-linear, non-differentiable and multi modal problems yet it suffers from the problem of premature convergence, stagnation and requires large time for computation as compared to other mathematical approaches [16].

Storn and Price [17] in 1995 introduced a well established meta-heuristic approach called differential evolution (DE) algorithm. Basic principle of DE states that the weighted difference of stochastically chosen two individual vectors is added to the third vector. The generated vector goes through the process of improvement using crossover and selection operations. In the comparison study performed by Storn [18], DE was found efficient than SA and GA. DE algorithm is found to be simpler, faster and requires lesser parameters than other EAs. A comparative study of performance of different optimization algorithms carried by Dong X.L. *et al.* [67] concluded DE outperforms GA in case of efficiency, robustness and convergence. In power systems, DE has been used by Arya L.D. *et al.* [68] in planning, operation and distribution. In constrained optimization problem [69], experimented on benchmark functions, DE performance is compared with PSO and it is observed that DE performs well in terms of repeatability and quality of solutions obtained. Deb A. [70] designed the circularly polarized microstrip antenna using EAs. The performance comparison is made between the standard PSO, RCGA, DE along with their advanced variants and it is discovered that differential evolution with global and local neighborhood search (DEGL)'s performance is superior than PSO, GA and their advanced variants.

Similar to other population based algorithms, DE also suffers from the problem of premature convergence and stagnation. The state of inactivity occurs when algorithm is not able to improve its solution for the long term generations. The problem of stagnation in case of DE, occurs when decision space is large and aggravates as the size of dimension increases [71]. To eliminate the DE deficiencies, improvement strategies have been proposed by many researchers. DE with trigonometric mutation [72], incorporation of fuzzy logic with DE [24], hybridization with PSO [22] and biogeography-based optimization (BBO) [73] have been proposed for the performance improvement Self adaptive DE is proposed by [74] for short term hydro scheduling problem. In order to make proper balance between exploration and exploitation capabilities of algorithm, chaotic map sequences have been introduced in mutation strategy of DE in number of engineering design problems [30–32] and the performance of chaotic based DE is found to be superior than other population based algorithms.

CHAPTER 3

PROBLEM DEFINATION

The impulse response of a system defines it's behavior when impulse signal is applied at the input. The impulse function is used to estimate the system behavior for all the frequencies. The impulse response of ideal digital filter is non causal in nature. The system is called non causal when system's output not only depends on the present or past values but also on the future values of input. This non causal nature of ideal filter makes the practical realization impossible. For any frequency selective filter causality is the main matter of concern. The equation for linear time invariant systems that are causal and can be practically realized, is mathematically written as:

$$y(n) = -\sum_{k=1}^{N} a_k y(n-1) + \sum_{k=0}^{M-1} b_k x(n-1) \tag{3.1}$$

where M and N are the lengths of the input and output discrete sampled signals. a_k and b_k are real and constant parameters. The frequency response of LTI systems is given by:

$$H(\omega) = \frac{\sum_{k=0}^{M-1} b_k e^{-j\omega k}}{1 + \sum_{k=1}^{N} a_k e^{-j\omega k}} \tag{3.2}$$

From the Eq. 3.2 it is clear than frequency response of any system depends on the value of its coefficients *i.e* a_k and b_k.

The designing process of any physically realizable digital filter is to approximate the ideal frequency response characteristics by proper selection of the filter coefficients. The magnitude response of physically realizable low pass digital filter is shown in Fig 3-1. The ω_p, ω_s and ω_c are the pass band edge frequency, stop band edge frequency and cut off frequency of the digital low pass filter respectively. The region which defines the transition of response from pass band to stop band regions is called transition band ($\triangle\omega$) of the filter. δ_p is the magnitude of ripple in pass band and δ_s is the magnitude of ripple present in stop band region of filter. The maximum stop band

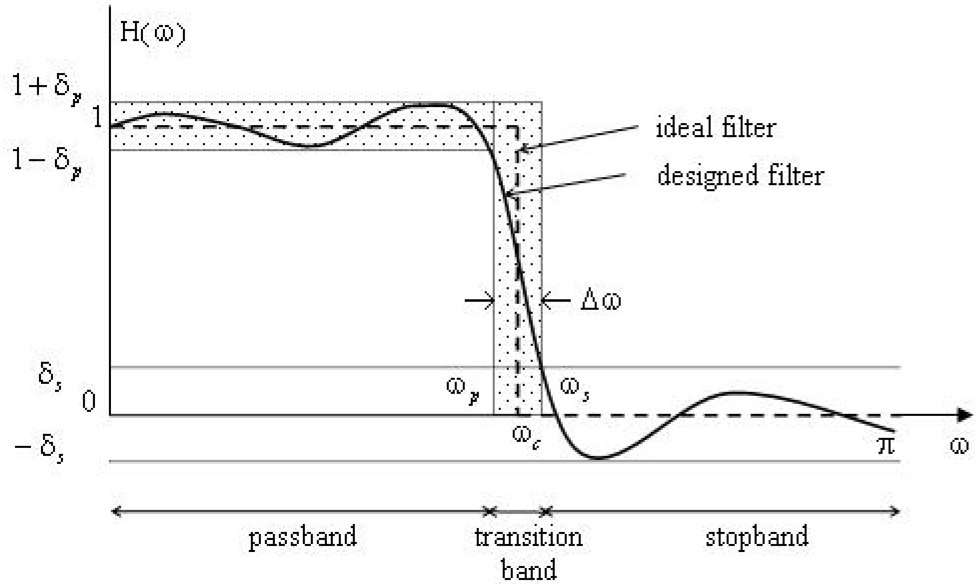


Figure 3-1: Amplitude vs frequency plot of practical low pass filter.

attenuation α_s and maximum pass band ripple α_p on decibel scale is defined as:

$$\alpha_s = -20log_{10}(\delta_s) \tag{3.3}$$

$$\alpha_p = -20log_{10}(\delta_p) \tag{3.4}$$

The digital FIR filter is mathematically represented as:

$$H(z) = \sum_{n=0}^{N} h(n)z^{-n} \tag{3.5}$$

where $h(n)$ is the finite impulse response, N is the order of the filter having length $(N+1)$.

The frequency response of the filter is obtained by the discrete fourier transformation (DFT) of the filter impulse response as:

$$H(e^{j\omega}) = \sum_{n=0}^{N} h(n)e^{-j\omega n} \tag{3.6}$$

For ideal low pass filter the frequency response is defined as:

$$D(\omega) = \begin{cases} 1 & for\,\omega \leq \omega_c \\ 0 & otherwise \end{cases} \tag{3.7}$$

For ideal high pass filter the frequency response is defined as:

$$D(\omega) = \begin{cases} 1 & for\, \omega \geq \omega_c \\ 0 & otherwise \end{cases} \tag{3.8}$$

where ω_c is the cut off frequency of filter. For designing an optimum filter using optimization algorithms, the error function plays a vital role. The algorithm optimizes the error function in order to obtain optimum filter coefficients. In this work, least square error (LSE) is used as error function. Mathematically, least square error $L(\omega)$ is defined as:

$$L(\omega) = min\left[\left|D_I(e^{j\omega})\right| - \left|H_d(e^{j\omega})\right|\right]^2 \tag{3.9}$$

where $D_I(e^{j\omega})$ is ideal frequency response and $H_d(e^{j\omega})$ is the frequency response of designed filter.

3.1 Ripple constraint handling

In order to obtain higher stop band attenuation, lower magnitude of ripples in stop band and pass band region, ripple constraint is introduced. A traditional penalty function based constraint handling technique is used to handle the ripple constraint [20]. The penalty function penalizes the infeasible ripple values that appear in the pass band and stop band of the filter. The maximum ripple $j_p(\omega)$ in pass band is defined as:

$$j_p(\omega) = max|\ 1 - H(\omega)\ | \qquad for\, \omega\ \in\ passband \tag{3.10}$$

In stop band maximum ripple $j_s(\omega)$ is defined as:

$$j_s(\omega) = max|\ H(\omega)\ | \qquad for\, \omega\ \in\ stopband \tag{3.11}$$

The modified objective function for filter design generated by considering the LSE along with ripple constraint is described as:

$$E(\omega) = c_r L(\omega) + c_p j_p(\omega) + c_s j_s(\omega) \tag{3.12}$$

where c_r, c_p, c_s are suitable weight vectors associated with least square error $L(\omega)$, maximum pass band ripple $j_p(\omega)$ and maximum stop band ripple $j_s(\omega)$ respectively. The coefficients $h(n)$ determines the filter type such as low pass, band pass, high pass and band reject filter. The objective of filter design problem is to obtain $h(n)$ so that expected specifications can be met. On the basis of length and symmetry of filter, FIR filters are classified as: type 1, type 2, type 3 and type 4. In this study, type 1 FIR filter having odd length and even symmetry *i.e.* $h(n) = h(N-n)$ for $0 \leqslant n \leqslant N$ is designed. The symmetry property of FIR filter reduces the computational task because instead of optimizing N only $\frac{N}{2}+1$ filter coefficients are required.

CHAPTER 4

CHAOTIC DIFFERENTIAL EVOLUTION ALGORITHM

Differential evolution (DE) algorithm is heuristic approach introduced by Storn R. and Price K. in 1995 [17]. It is the population based algorithm and works in similar fashion as other genetic based algorithms do. To construct better solutions, DE differs from GA algorithm in the sense that it highly depends on the mutation operation while the other depends on crossover operation [2]. DE is the simple, fast and inherently parallel search method. The DE has better convergence than other EAs and also requires few control parameters. The basic idea of DE is to generate new individual vector by performing addition of weighted difference between two randomly chosen individual vectors with third randomly chosen vector. Generated new individual vector is called mutant vector. The mutant vector goes through the process of crossover and selection to obtain the optimal solution for any problem. In DE, mutation is the search mechanism while crossover and selection operations directs the search towards the region having set of better solutions. The control parameters of DE are population size (N_p), mutation scale factor (F) and crossover rate (CR). The basic steps involved in the DE algorithm are :

1. **Initialization:** The initial step involve in algorithm is the initialization of population. The initial population of size N_P and D dimension is randomly generated. With the increase in the size of dimension, the population size must be taken large enough so that the algorithm should have large search space to find best solution out of it. The size of population taken must be between ten times or five times of the dimension [18]. In D dimensional search space an i^{th} individual vector of population at k^{th} generation is denoted as x_{ij}^k; $(i = 1,2,...,N_P, j = 1,2,...,D)$. The value of individual vectors must lie within certain range. Each individual vector is thus defined as:

$$x_{ij}^k = x_j^{min} + rand(x_j^{max} - x_j^{min}) \tag{4.1}$$

   where x_j^{max} and x_j^{min} is the maximum and the minimum limit of individual vector respectively and $rand \in [0,1]$ is uniformly distributed random number.

2. **Mutation:** For every individual vector of population, a mutant vector V_{ij}^k is generated by

the process of mutation. Among different mutation strategies existing in DE, most popular mutation strategies are defined below:

$$DE/rand/1 : V_{ij}^k = x_{r_1 j}^k + F(x_{r_2 j}^k - x_{r_3 j}^k) \tag{4.2}$$

$$DE/best/1 : V_{ij}^k = x_{best}^k + F(x_{r_1 j}^k - x_{r_2 j}^k) \tag{4.3}$$

$$DE/current-to-best/1 : \quad V_{ij}^k = x_{ij}^k + F(x_{best}^k - x_{ij}^k) + F(x_{r_1 j}^k - x_{r_2 j}^k) \tag{4.4}$$

$$DE/best/2 : V_{ij}^k = x_{best}^k + F(x_{r_1 j}^k - x_{r_2 j}^k) + F(x_{r_3 j}^k - x_{r_4 j}^k) \tag{4.5}$$

$$DE/rand/2 : V_{ij}^k = x_{r_1 j}^k + F(x_{r_2 j}^k - x_{r_3 j}^k) + F(x_{r_4 j}^k - x_{r_5 j}^k) \tag{4.6}$$

where x_{best}^k is the best individual vector that acquire best fitness value at generation k and indices $r1 \neq r2 \neq r3 \neq r4 \neq r5$ are randomly chosen index from the range $[1, N_p]$ and are different from base index i. The process of mutation is illustrated for DE/rand/1 scheme in the Fig 4-1.

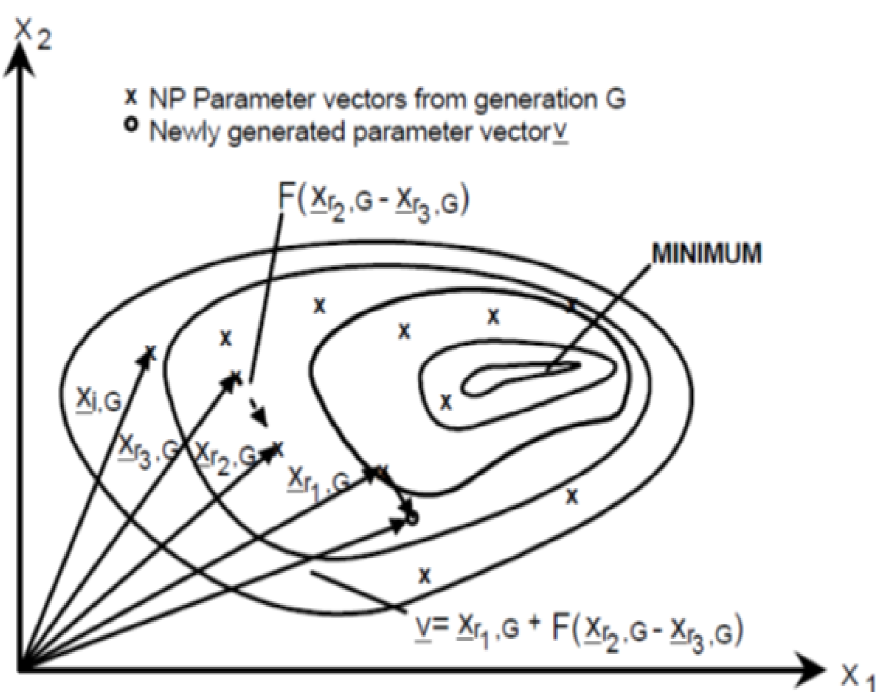


Figure 4-1: Mutation process for DE/rand/1 scheme.

3. **Crossover:** In order to maintain useful diversity in population, crossover operation is performed. Crossover operation involves mixing of parameters of mutant vector with that of individual vector and generate new vector called trial vector. In the DE family of algorithms, there are two commonly used crossover methods namely exponential and binomial [75]. In binomial crossover, the crossover is performed when value of random number is less than crossover rate (CR). Mathematically, the generation of trial vector U_{ij} is represented as:

$$U_{ij}^{k} = \begin{cases} V_{ij}^{k} & if(rand_j \leq CR\, or\, j = j_{rand}) \\ x_{ij}^{k} & otherwise \end{cases}$$

$$(i = 1,2,\ldots,N_P, j = 1,2,\ldots,D) \tag{4.7}$$

where $rand_j \in [0,1]$ is the j^{th} evaluation of a uniform random number generator and $j_{rand} \in [1,D]$ is the randomly chosen index.

4. **Selection:** In selection procedure, better performing individual is selected for the generation. It determines whether to select trial or individual vector as member of $(k+1)^{th}$ generation. Mathematically, it is written as:

$$x_{ij}^{k+1} = \begin{cases} U_{ij}^{k} & iff(U_{ij}^{k+1}) < f(x_{ij}^{k}) \\ x_{ij}^{k} & otherwise \end{cases}$$

$$(i = 1,2,\ldots,N_P, j = 1,2,\ldots,D) \tag{4.8}$$

where $f()$ is the objective function defined for the problem to be minimized.

If the trial vector yields the lower fitness value for the objective function then it replaces the individual vector in the next generation and otherwise individual vector remains same in next generation.

The previously mentioned steps are repeated for every individual vector until the best optimum solution is obtained. The algorithm terminates when maximum number of generations, specified by the user, have been executed. The iterative search process of DE is defined in algorithm 1.

Algorithm 1 Differential evolution algorithm

Initialize individual's population array using random process

1: **while** stopping criteria unmet **do**
2: Evaluate the fitness value of each individual based on objective function
3: Select three individual's randomly for performing mutation operation and generate mutant vector.
4: Perform crossover operation to generate trial vector
5: Select an individual for next generation out of trial and target vector based on lower fitness value of objective function
6: **end while**

4.1 Chaotic Differential Evolution Algorithm

The chaotic differential evolution (CDE) algorithm is a modification of conventional DE algorithm, which relies on chaotic sequence for its performance improvement. The very basic difference between standard DE and CDE is that in DE the random numbers are generated by using uniformly distributed random number generator whereas in CDE, chaotic map sequences replace the standard random number. Chaotic map is random number generation function. Chaotic map exhibits the similar properties of randomness with better statistical and dynamical properties. Tent map and Gauss map are used for generating randomization in algorithm's search process. The tent map based chaotic sequence generated is defined as [29] :

$$\rho_{n+1} = \begin{cases} \frac{\rho_n}{0.7} & if\rho_n < 0.7 \\ \frac{10}{3}(1-\rho_n) & otherwise \end{cases} \tag{4.9}$$

The equation for gauss map is defined as follow [29] :

$$\rho_{n+1} = \begin{cases} 0 & if\rho_n = 0 \\ \frac{1}{\rho_n} mod\, 1 & otherwise \end{cases} \tag{4.10}$$

where the value ρ_n lies between 0 and 1.

The basic steps involve in CDE approach are given below:

- **Initialization:** The chaotic map sequence is used to randomly initialize the population (x_{ij}^{k}) in the problem specific limits. In the present work $\rho(0,1)$ is used to represent the chaotic sequence. The population individuals are initialized by scaling the chaotic sequence generated random numbers in the specified limits.

- **Mutation:** In order to perform mutation operation on individual vectors, different mutation strategies as mentioned in Eq. 4.2-4.6 can be used. In this algorithm the index numbers, used for selecting random individuals to perform mutation, are generated by chaotic map sequence.

- **Crossover:** A trial vector (U_{ij}^{k}) is generated by performing the crossover operation between mutated (V_{ij}^{th}) and individual vector (x_{ij}^{k}). In chaotic crossover operation, the random number and randomly chosen index value are generated using chaotic maps. The value of random number generated is compared with crossover rate (CR) in order to ensure whether to perform the crossover operation or not.

- **Selection:** The objective function is evaluated for both trial and individual vector to select the next generation individual vector (x_{ij}^{k+1}). If the trial vector produces lower fitness value than individual vector, then it replaces that individual vector in next generation. The vector having lowest fitness value is selected as the best (x_{best}) solution for the next generation.

The process of mutation, crossover and selection is performed at each generation until the stopping criteria is satisfied. The stopping criteria used in this work is the maximum number of generations. The pseudo-code for CDE algorithm is presented in Algorithm 2.

Algorithm 2 Psudocode for CDE algorithm

Input: Population size N_p, Dimensions D, mutation scale factor F, maximum limit x^{max}, minimum limit x^{min}, , crossover rate CR, maximum generation IT^{max}

Generate a random initial population
for $i = 1 \rightarrow\ N_p$ **do**
  for $j = 1 \rightarrow\ D$ **do**
    $x_{ij} = x^{min} + \rho(0,1)(x^{max} - x^{min})$
  end for
  Evaluate fitness $f(x_{ij})$
end for
x_{best}=x_{ij} with $min[f(x_{ij})]$
for $k = 1 \rightarrow\ IT^{max}$ **do**
  for $i = 1 \rightarrow\ N_p$ **do**
    Select the individuals to perform mutation operation such that $\rho_1 \neq \rho_2 \neq \rho_3$
    $V^k_{ij} = x^k_{\rho_1 j} + F(x^k_{\rho_2 j} - x^k_{\rho_3 j})$
    Crossover operation
    if $\rho_j \leqslant CR$ or $\rho_j = j$ **then**
      $U^k_{ij} = V^k_{ij}$
    else
      x^k_{ij}
    end if
    Selection operation
    if $f(U^k_i) < f(x^k_i)$ **then**
      x^{k+1}_i =U^k_i
    end if
  end for
  if f(x^{k+1}_i)<f(x_{best}) **then**
    $x_{best} = x^{k+1}_i$
  end if
end for

Output: x_{best}

CHAPTER 5

SOLUTION APPROACH

The objective of finte impulse response (FIR) filter design process is to obtain a set of filter coefficients to achieve desired frequency response specifications. The solution methodology used in this work involves the use of chaotic differential evolution (CDE) algorithm to solve the FIR filter design problem. In this process, suitable set of filter coefficients are searched using the CDE approach for the desired specification of the filter.

5.1 Solution representation

In order to design the FIR filter, the filter coefficients are the search variables. In this optimization approach, dimension size (D) is the number of filter coefficients to be searched. For designing the type 1 FIR filter, the number of coefficients reduced to $\frac{N}{2}+1$. An i^{th} individual h_i represents a set of coefficients such that $h_i = [h_1, h_2, ...h_D]$. Considering N_p as population size, the filter coefficients at k^{th} generation are represented as:

$$h_i^k = [h_{i1}^k, h_{i2}^k, ...h_{iD}^k] \qquad (i = 1, 2, \ldots, N_p) \tag{5.1}$$

The control parameters used in algorithm are mutation scale factor (F) and crossover rate (CR). The value of F and CR lies between 0 and 1.

5.2 Population initialization

The initial population representing the set of filter coefficients is generated randomly using chaotic map sequence. The maximum (h^{max}) and minimum (h^{min}) limits of the filter coefficients are specified as 1 and -1 respectively. Mathematically, the initialization of population is described as:

$$h_{ij} = h_j^{min} + \rho(0,1)(h_j^{max} - h_j^{min})$$
$$(i = 1, 2..., N_p,\ j = 1, 2, ..., D) \tag{5.2}$$

where $\rho(0,1)$ is the random number generated by chaotic map.

5.3 Solution evaluation

The objective function $E(\omega)$ for designing the FIR filter is mentioned below:

$$E(\omega) = c_r L(\omega) + c_p j_p(\omega) + c_s j_s(\omega) \tag{5.3}$$

where c_r, c_p, c_s are suitable weight vectors associated with least square error $L(\omega)$, maximum pass band ripple $j_p(\omega)$ and maximum stop band ripple $j_s(\omega)$ respectively. After performing experimental trials, the value of c_r, c_p, c_s are selected as 1, 1000 and 2000 respectively. The value of c_s is taken more than c_p because the magnitude of ripples is generally more in stop band than in pass band region of filter. The objective function is evaluated for each individual vector of population. The vector having lowest fitness value is represented as h_{best}.

5.4 Solution updation and termination

An i^{th} individual of the population is upgraded using the process of mutation, crossover and selection. At each generation k an i^{th} individual is used to produce the mutant vector V_{ij}^k. The chaotic crossover operation is performed to create new vector called the trial vector U_{ij}^k. The selection operation selects the individual x_{ij}^{k+1} for the next generation out of U_{ij}^k and x_{ij}^k. The selection is performed on the basis of objective function value specified by Eq. 5.3. In addition, the population best represented by h_{best} is upgraded based upon better fitness value by comparing with x_{ij}^{k+1}. The process of updation continues until the maximum number of generation (IT^{max}) has been executed. After completion of iterative process, the h_{best} is selected as a solution of FIR filter design problem. The complete CDE iterative process to design FIR filter is represented as flow chart in Fig 5-1.

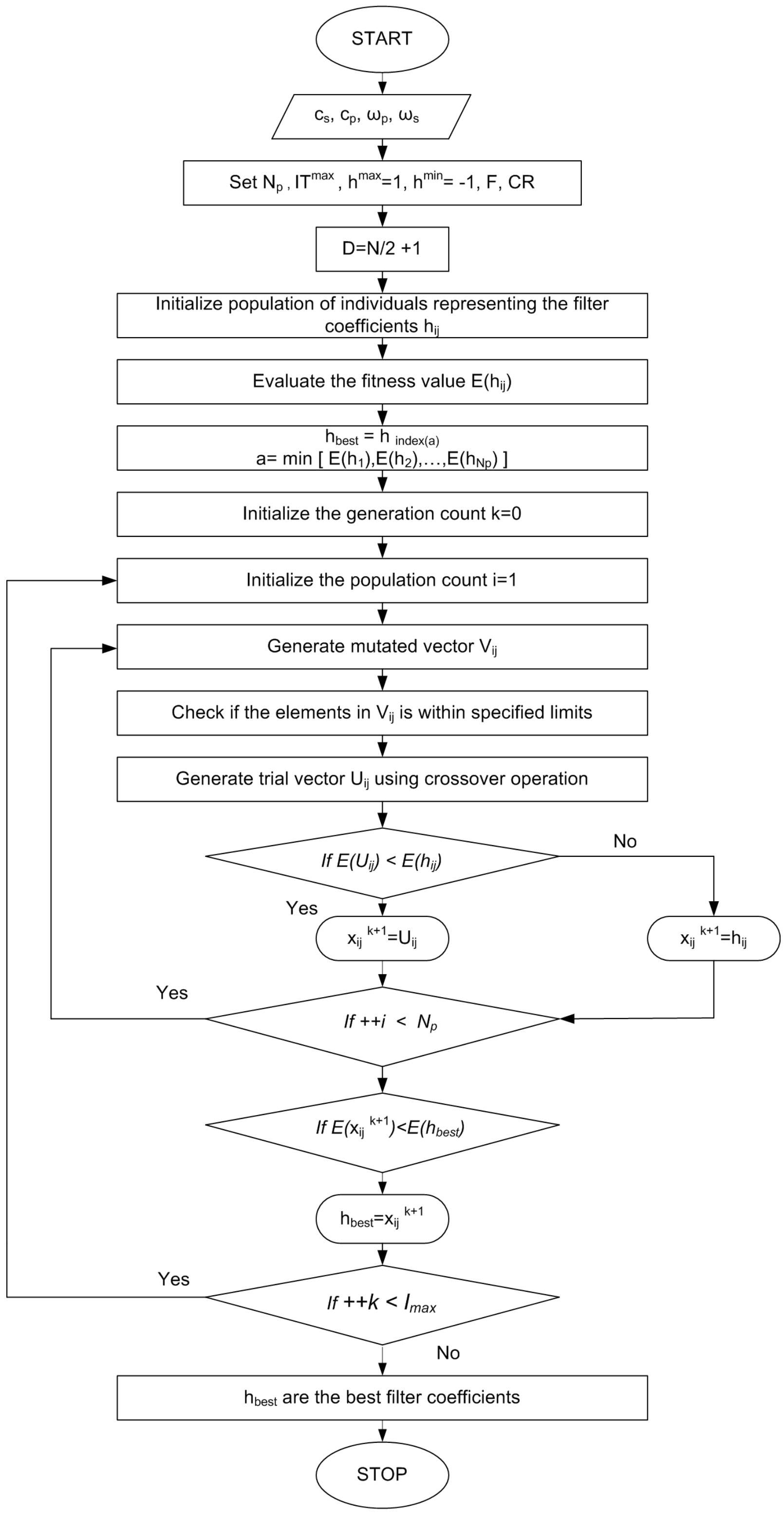


Figure 5-1: Flow chart of CDE approach for FIR filter design

CHAPTER 6

RESULTS AND DISCUSSION

6.1 Performance anlysis for generalised test functions

The performance of DE and CDE algorithm is evaluated by using test functions. The generalized benchmark functions [76, 77] used in this work for the evaluation purpose are tabulated in Table 6.1. Table 6.1 describes the description, range and global minima of the test functions under consideration. The algorithm presented in flowchart is implemented in MATLAB 8.3 version using

Table 6.1: Generalized test functions range and global optimum

Function	Description	Expression	Range	Global optimum
f_1	Sphere function	$\sum_{i=1}^{D} x_i^2$	[-5.12, 5.12]	0
f_2	Rosenbrock's function	$\sum_{i=1}^{D-1} 100[(x_i^2 - x_{i+1})^2 + (x_i - 1)^2]$	[-30, 30]	0
f_3	Schwefel 2.22 function	$\sum_{i=1}^{D} \lvert x_i^2\rvert + \prod_{i=1}^{D} \lvert x_i\rvert$	[-100, 100]	0
f_4	Rastrigin function	$10.D + \sum_{i=1}^{D} x_i^2 - 10.\cos(2\pi x_i)$	[-5.12, 5.12]	0
f_5	Schwefel function	$418.98.D - \sum_{i=1}^{D} x_i^2 \sin\sqrt{x_i}$	[-500, 500]	0
f_6	Ackley function	$-20e^{-0.02\sqrt{\frac{\sum_{i=1}^{D} x_i^2}{D}}} - e^{\frac{1}{D}\sum_{i=1}^{D}\cos 2\pi x_i} + 20 + e$	[-35, 35]	0
f_7	Griewank function	$\frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[-100, 100]	0
f_8	Step function	$\sum_{i=1}^{D} \lfloor \lvert x_i\rvert \rfloor$	[-100, 100]	0
f_9	Step2 function	$\sum_{i=1}^{D} \lfloor x_i + 0.5 \rfloor^2$	[-100, 100]	0

Intel core i3, 1.90GHz with installed memory of 4.00 GB. The values of control parameters of DE and CDE used to evaluate the performance of various benchmark test functions are described in Table 6.2

First of all, DE algorithm is implemented by incorporating different mutation strategies on benchmark test functions. The comparison between different mutation scheme is made on the basis of convergence rate. Thirty independent trials are performed for each mutation scheme based algorithm and the convergence plot of best solution out of thirty trials are shown in Fig. 6-1 to 6-9.

Table 6.2: Parameters values of DE and CDE for optimizing the test functions

Parameters	DE / CDE
Population size (N_p)	100
Dimension (D)	10
Mutation scale factor (F)	0.85
Crossover rate (CR)	0.25
Number of iteration (IT^{max})	2000

Fitness value vs iteration plot is shown for first 100 iterations only.

Figure 6-1 shows the convergence profile of different mutation scheme for Sphere function. It is observed from the convergence plot that DE/current-to-best/1 and DE/best/1 have faster convergence rate than other schemes. Figure 6-2 shows the convergence profile of different mutation scheme for Rosenbrock function. In this case, DE/current-to-best/1 performs better than other schemes.

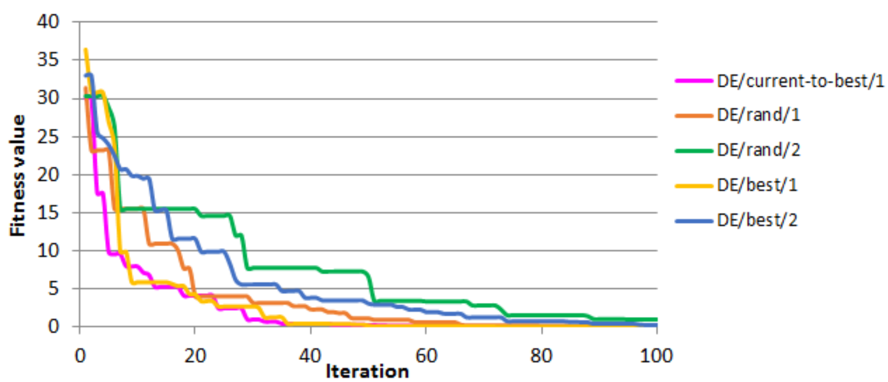


Figure 6-1: Sphere function

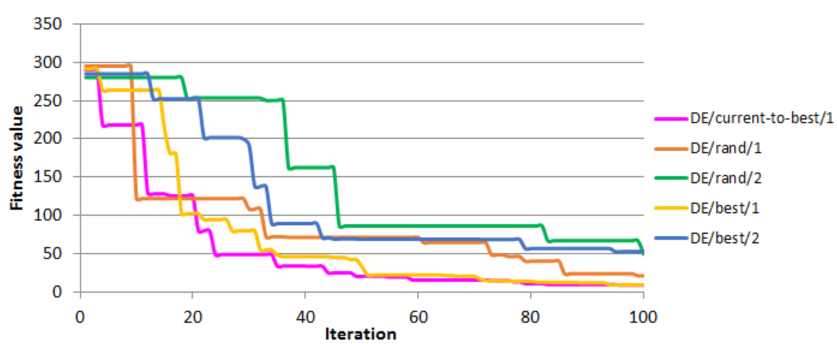


Figure 6-2: Rosenbrock function

Figure 6-3 and Fig. 6-4 shows the convergence profile for Schwefel 2.22 function and Rastrigin function respectively. In these figures DE/current-to-best/1 and DE/best/1 have similar performance and better convergence than other methods.

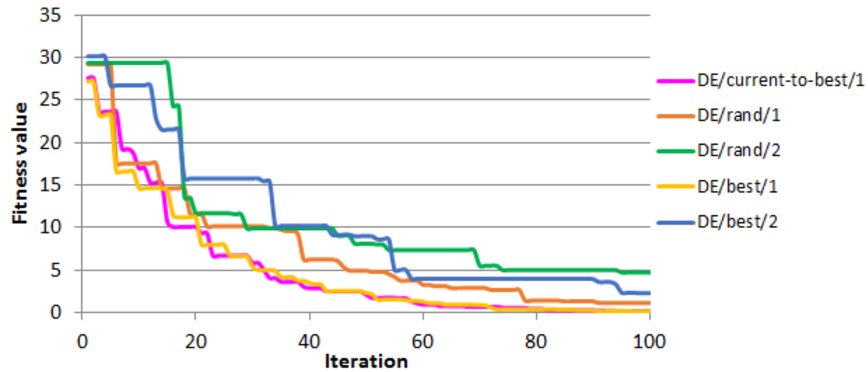


Figure 6-3: Schwefel 2.22 function

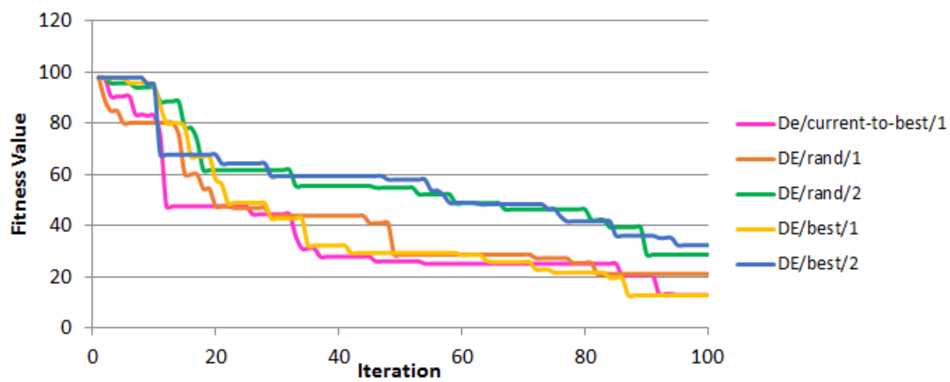


Figure 6-4: Rastrigin function

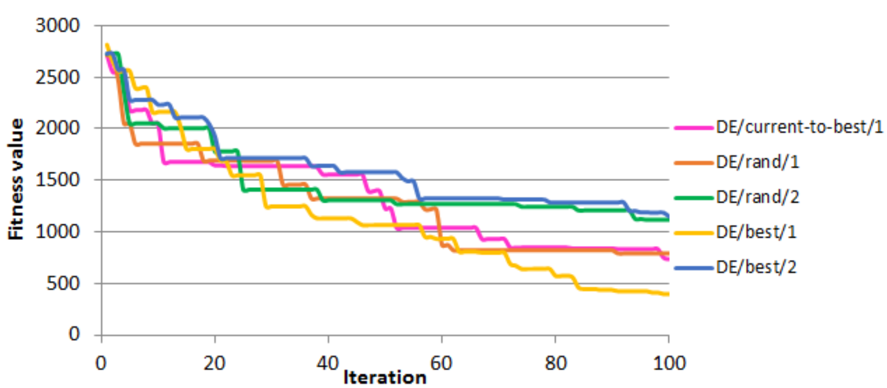


Figure 6-5: Schwefel function

In case of Schwefel function the convergence profile shown in Fig. 6-5 demonstrates that DE/best/1 converges faster than all other strategies. Figure 6-6, Fig. 6-8, Fig. 6-7 and Fig. 6-9 shows the convergence profile for Ackley, Griewank, Step and Step2 function respectively.

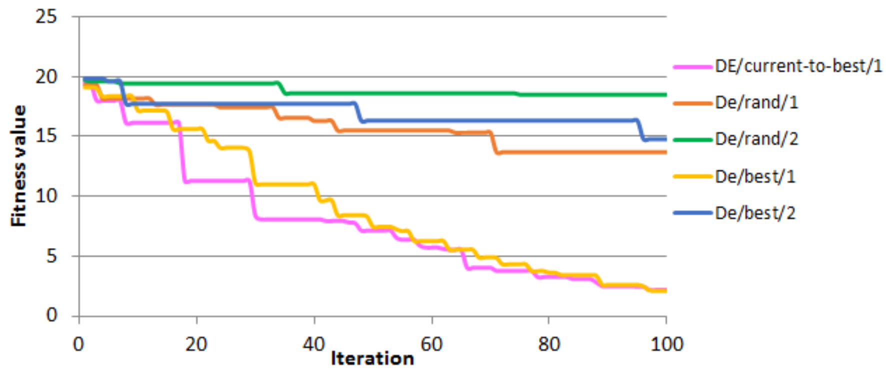


Figure 6-6: Ackley function

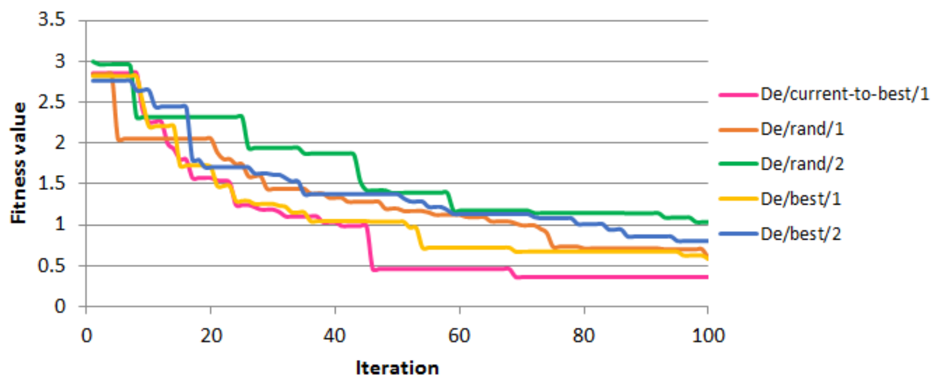


Figure 6-7: Griewank function

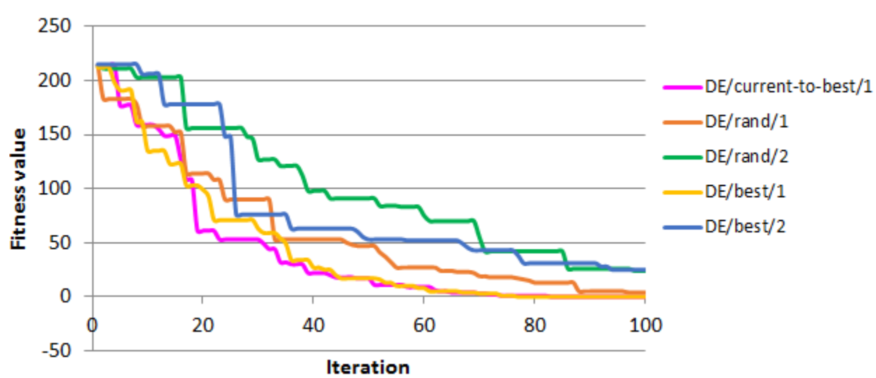


Figure 6-8: Step function

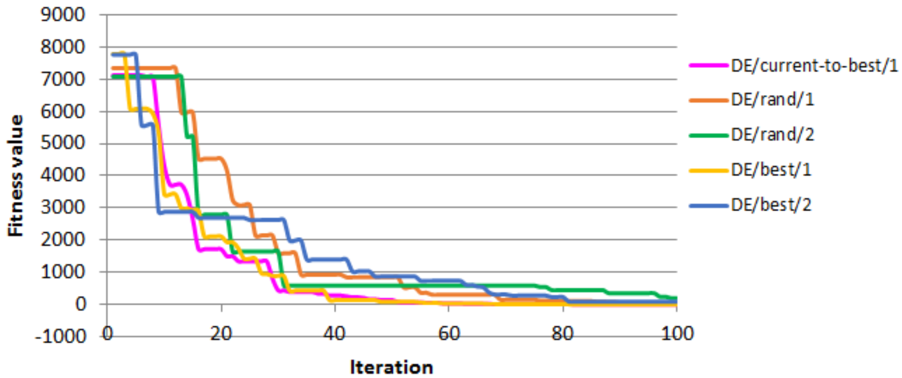
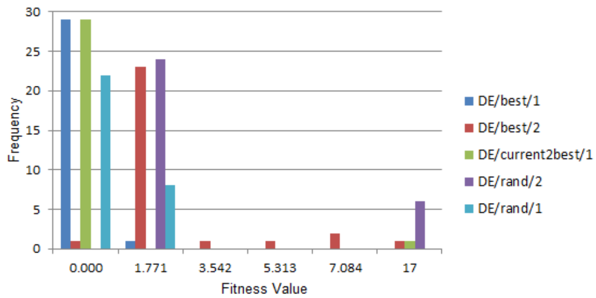


Figure 6-9: Step2 function

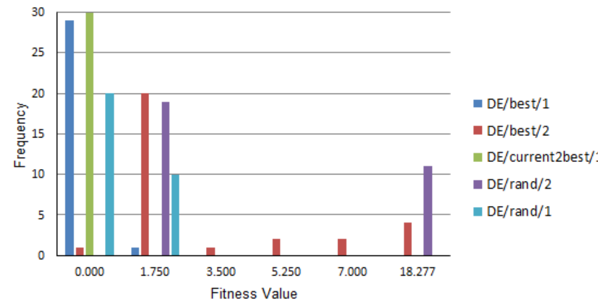
It is well observed in all these convergence plots that DE/current-to-best/1 has better convergence rate than all other mutation schemes. After evaluating the all test functions using different mutation schemes of DE, the DE/current-to-best/1 mutation strategy's performance is found superior in each case.

The CDE algorithm's performance is compared using Tent map and Gauss map as random number generator function. The CDE algorithm using Tent map sequence is termed as CDE-1 and for using Gauss map sequence is termed as CDE-2. The five mutation scheme used in CDE algorithm are termed as CDE/x/y, where x and y defines the mutation scheme. The algorithm is made to run for 30 times. The number of iterations has been taken 200. The minimum fitness values attained using different mutation strategies is plotted against the frequency of occurrence.

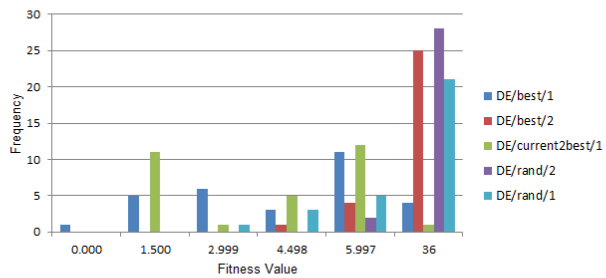
Figure 6-10, 6-11 and 6-12 shows the frequency vs fitness plot for test functions under consideration using CDE-1 and CDE-2 algorithms respectively. For Sphere function Fig. 6-10a shows the convergence for CDE-1 and Fig. 6-10b is for using CDE-2. It is observed that for both CDE-1 and CDE-2, CDE/current-to-best/1 mutation strategy converges to minimum value for almost all 30 runs. Fig. 6-10c and Fig. 6-10d shows the frequency plot for Rosenbrock function using CDE-1 and CDE-2 respectively. The CDE/best/1 mutation scheme in CDE-2 algorithm attains minimum value for more number of times than in CDE-1. CDE/current-to-best/1 has comparable performance in both the cases. For Schwefel 2.22 function Fig. 6-10e and Fig. 6-10f shows frequency plot using CDE-1 and CDE-2 respectively. It is found that both CDE/current-to-best/1 and CDE/best/1 approaches to minimum value for more number of times than any other mutation scheme in both CDE-1 and CDE-2 algorithms.



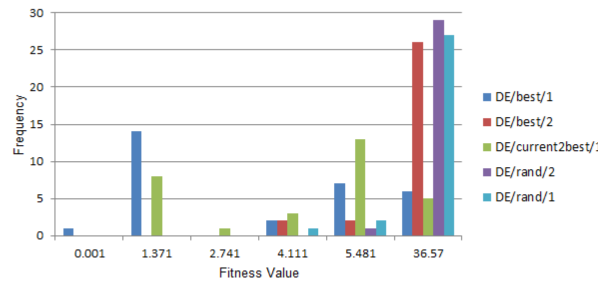
(a) Frequency plot for Sphere function using CDE-1



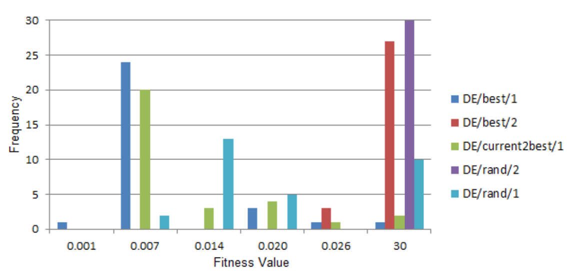
(b) Frequency plot for Sphere function using CDE-2



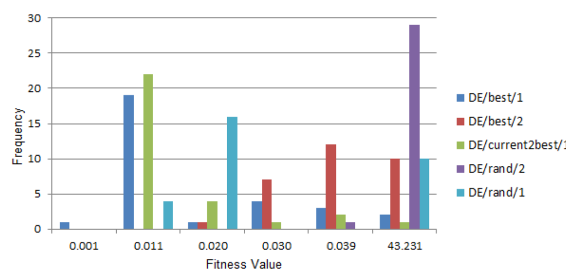
(c) Frequency plot for Rosenbrock function using CDE-1



(d) Frequency plot for Rosenbrock function using CDE-2



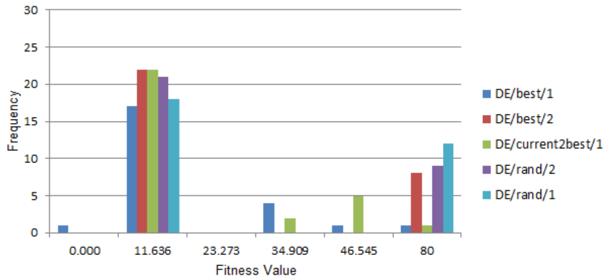
(e) Frequency plot for Schwefel's 2.22 function using CDE-1



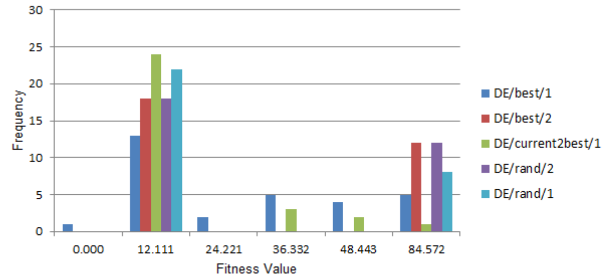
(f) Frequency plot for Schwefel's 2.22 function using CDE-2

Figure 6-10: Generalized test functions convergence comparison using Tent map and Gauss map

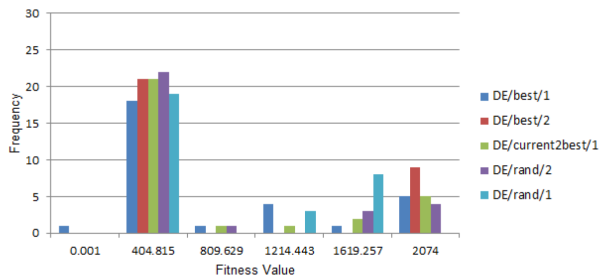
Considering Fig. 6-11, frequency plot for Rastrigin function using CDE-1 and CDE-2 is depicted in Fig. 6-11a and Fig. 6-11b respectively. In both these techniques, CDE/current-to-best/1 performs in the similar manner. For Schwefel function Fig. 6-11c and Fig. 6-11d shows frequency plot using CDE-1 and CDE-2 respectively. CDE/current-to-best/1 attain minimum value for more number of times for using CDE-1 than using CDE-2 approach. The frequency plot for Ackley function using CDE-1 and CDE-2 approach is shown in Fig. 6-11e and Fig. 6-11frespectively. Fig. 6-11e shows that CDE/current-to-best/1 performs better in case of CDE-1 algorithm.



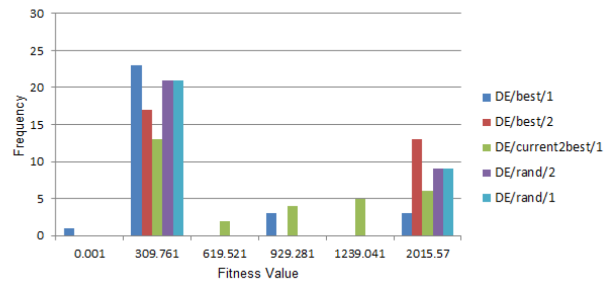
(a) Frequency plot for Rastrigin function using CDE-1



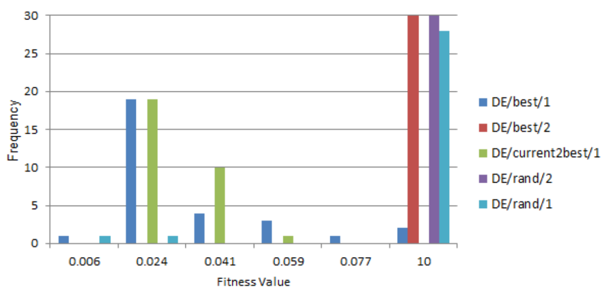
(b) Frequency plot for Rastrigin function using CDE-1



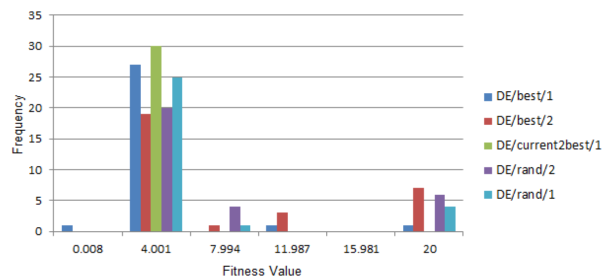
(c) Frequency plot for Schwefel function using CDE–1



(d) Frequency plot for Schwefel function using CDE-2



(e) Frequency plot for Ackley function using CDE-1

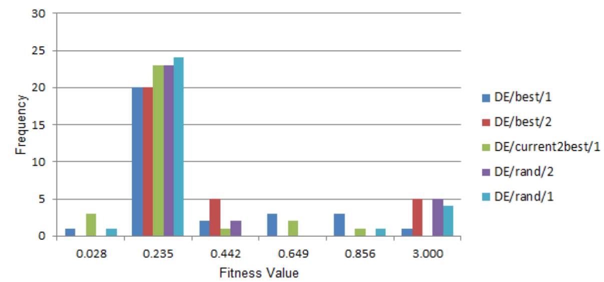


(f) Frequency plot for Ackley function using CDE-2

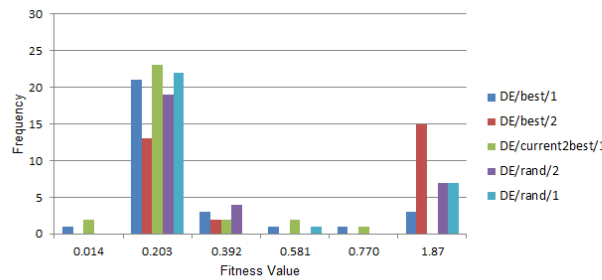
Figure 6-11: Generalized test functions convergence comparison using Tent map and Gauss map

Considering Fig. 6-12, frequency plot for Griewank function using CDE-1 and CDE-2 is depicted in Fig. 6-12a and Fig. 6-12b respectively. The plot shows that CDE/current-to-best/1 has almost comparable performance in evaluating the function for using either CDE-1 or CDE-2 technique. In case of Step function Fig. 6-12c and Fig. 6-12d shows that CDE/current-to-best/1 and CDE/best/1 attain minimum value for all 30 runs for both the cases. CDE/current-to-best/1 is found to attain the minimum value for almost all runs (Fig. 6-12e and Fig. 6-12f) in case of using either CDE-1 and CDE-2 for evaluating the step2 function.

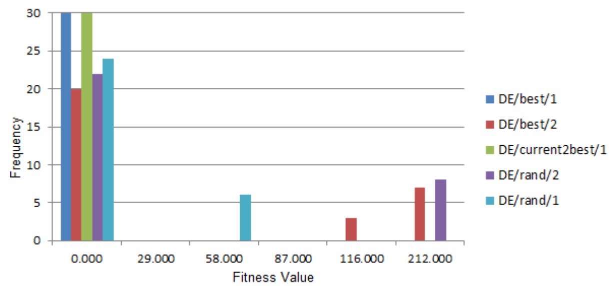
From the above discussion, it is observed that CDE/current-to-best/1 mutation scheme generally outperforms all other mutation schemes and the performance of CDE-1 is generally found better than CDE-2 algorithm. For comparing the performance of DE and CDE in evaluating the test



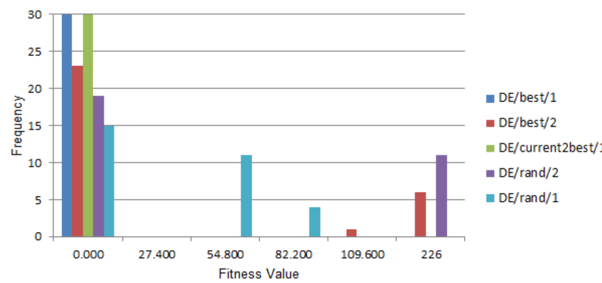
(a) Frequency plot for Griewank function using CDE-1



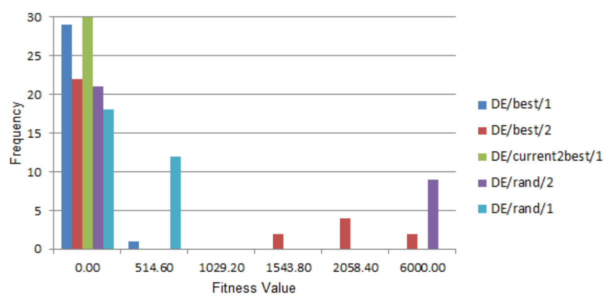
(b) Frequency plot for Griewank function using CDE-2



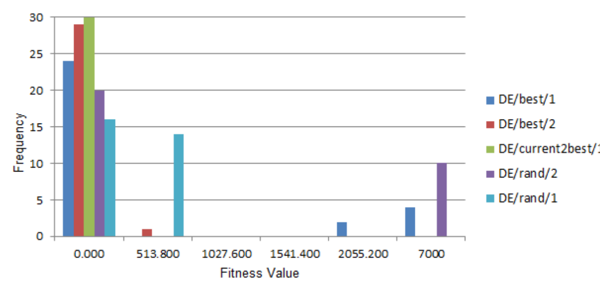
(c) Frequency plot for Step function using CDE-1



(d) Frequency plot for Step function using CDE-2



(e) Frequency plot for Step2 function using CDE-1



(f) Frequency plot for Step2 function using CDE-2

Figure 6-12: Generalized test functions convergence comparison using Tent map and Gauss map

functions, current-to-best/1 mutation strategy is used in both the cases. In case of CDE algorithm, for randomization tent map chaotic sequence is selected. The algorithms are made to run for 30 times. Number of iterations taken as 2000. The comparison of best fitness values attained for test functions using DE and CDE algorithm is tabulated in Table 6.3. It is well observed that CDE attains the lower fitness value for all the functions under consideration than the DE.

6.2 Performance analysis of proposed approach for filter design problem

The performance of proposed algorithm is investigated for solving real time engineering design problems. FIR filters are designed using CDE algorithm and results are compared with that ob-

Table 6.3: Comparison of best fitness values attained

Function	DE	CDE
f_1	1.01×10^{-79}	1.10×10^{-81}
f_2	2.58	4.51×10^{-17}
f_3	9.82×10^{-44}	3.21×10^{-44}
f_4	1.12×10^{-8}	5.30×10^{-9}
f_5	1.18×10^{2}	1.27×10^{-4}
f_6	7.999×10^{-15}	4.44×10^{-15}
f_7	0	0
f_8	0	0

tained using classical DE algorithm. The algorithm searches for the best set of filter coefficients to meet the desired frequency spectral specifications. In this work, the low pass and high pass filters of 20^{th} and 30^{th} order are designed using DE and CDE algorithm. The filter specifications are presented in Table 6.4. To design 20^{th} order filter, the population size is taken as 50 and number of iterations chosen as 500. For 30^{th} order filter designing, the population size has been taken 100 and number of iteration chosen as 1000. Values of F and CR taken are 0.8 and 0.9 respectively.

Table 6.4: FIR filter specifications

	Low pass filter	High pass filter
Sampling frequency (f_s)	$1Hz$	$1Hz$
Number of frequency samples	128	128
Pass band edge frequency (normalized) (ω_p)	0.45	0.5
Stop band edge frequency (normalized) (ω_s)	0.55	0.45
Transition width ($\triangle\omega$)	0.1	0.1

6.2.1 20^{th} order low pass filter

The magnitude response and normalized plot for 20^{th} order low pass filter is demonstrated in Fig. 6-13 and Fig. 6-14 respectively. The frequency response characteristics of filter designed using

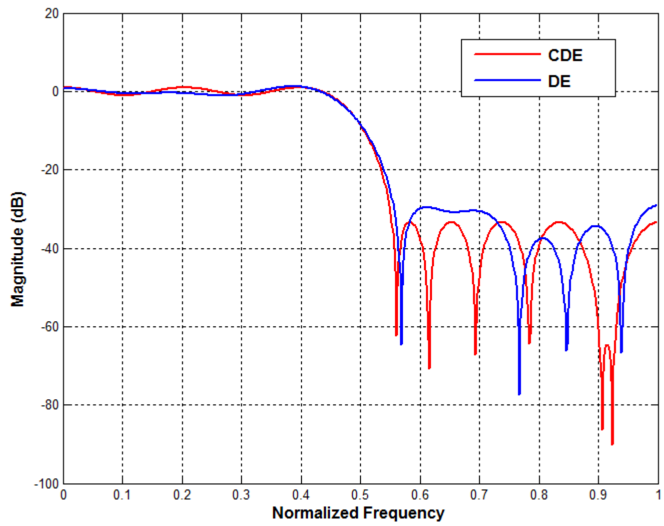


Figure 6-13: Magnitude response in dB for 20^{th} order low pass filter

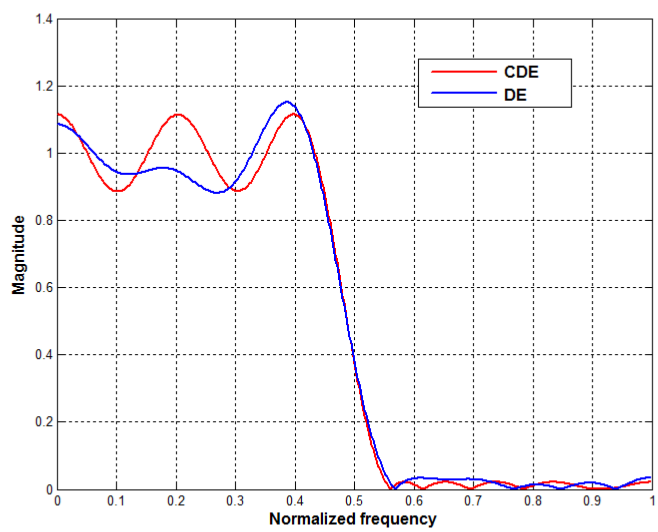


Figure 6-14: Normalized plot for 20^{th} order low pass filter

DE and CDE are compared and discussed in Table 6.5. It is observed that filter designed using CDE provide higher attenuation in stop band *i.e.* 33.699 dB while using DE stop attenuation achieved is 27.0568 dB. The filter coefficients obtained using DE and CDE approach for designing 20^{th} order low pass filter are tabulated in Table 6.6.

Table 6.5: Parameters comparison for 20^{th} order low pass filter

Parameter	DE	CDE
Maximum pass band ripple (normalized)	0.150329	0.114594
Maximum stop band ripple (normalized)	0.044377	0.021454
Maximum stop band attenuation (dB)	27.0568	33.3699

Table 6.6: Optimal filter coefficients for 20^{th} order low pass filter

$h(n)$	DE	CDE	$h(n)$
$h(1)$	-0.021680251	-0.042110069	$h(21)$
$h(2)$	-0.017237346	-0.042110069	$h(20)$
$h(3)$	0.013026892	0.004484677	$h(19)$
$h(4)$	0.037789223	0.03776406	$h(18)$
$h(5)$	-0.017238376	-0.006437069	$h(17)$
$h(6)$	-0.075986539	-0.057235066	$h(16)$
$h(7)$	-0.007667183	0.006649022	$h(15)$
$h(8)$	0.104124264	0.103519305	$h(14)$
$h(9)$	-0.005524652	-0.009426993	$h(13)$
$h(10)$	-0.311296449	-0.315284447	$h(12)$
$h(11)$	-0.481737037	-0.493638326	$h(11)$

6.2.2 30^{th} order low pass filter

For 30^{th} order low pass filter the magnitude and normalized plot is shown in Fig. 6-15 and Fig. 6-16. The comparison of frequency response characteristics for 30^{th} order low pass filter is discussed in Table 6.7. The magnitude of ripple in pass band and stop band of filter designed using CDE has been significantly decreased. The CDE yields maximum stop band attenuation of 34.6792 in comparison to DE (24.2650 dB). The set of filter coefficients obtained for 30^{th} order low pass filter are presented in Table 6.8.

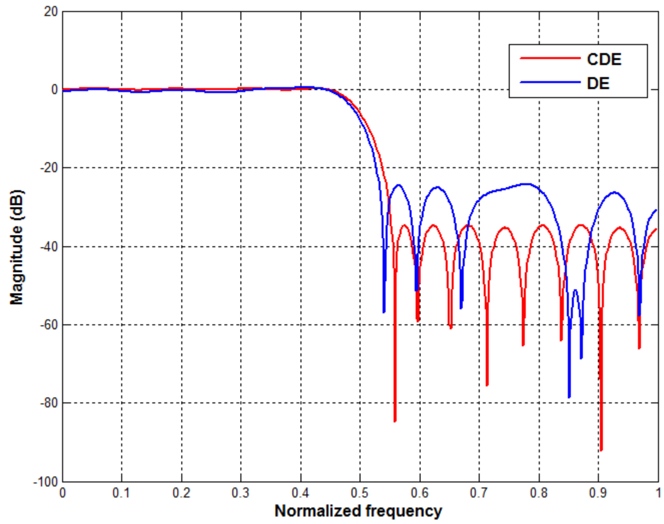


Figure 6-15: Magnitude response in dB for 30^{th} order low pass filter

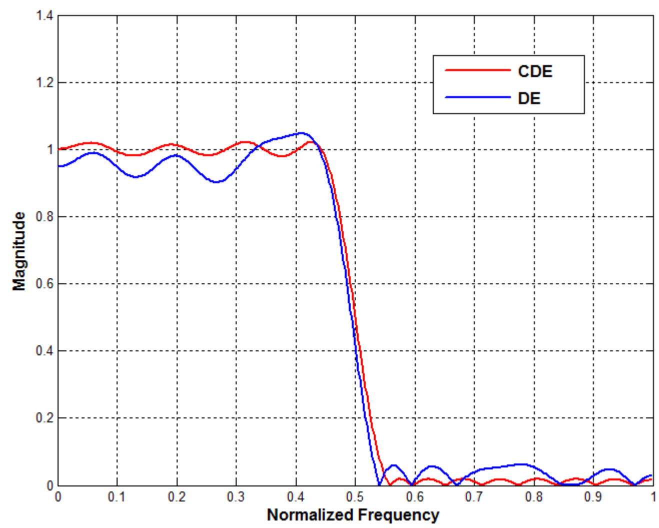


Figure 6-16: Normalized plot for 30^{th} order low pass filter

Table 6.7: Parameters comparison for 30^{th} order low pass filter

Parameter	DE	CDE
Maximum pass band ripple (normalized)	0.09801	0.021442
Maximum stop band ripple (normalized)	0.06121	0.018452
Maximum stop band attenuation (dB)	24.2650	34.6792

6.2.3 20^{th} order high pass filter

Figure 6-17 and Fig. 6-18 shows the magnitude and normalized plot respectively for 20^{th} order high pass filter. Comparison of DE and CDE based on frequency response characteristics is

Table 6.8: Optimal filter coefficients for 30^{th} order low pass filter

$h(n)$	DE	CDE	$h(n)$
$h(1)$	-0.015707031	0.013542642	$h(31)$
$h(2)$	-0.002797467	-0.000832524	$h(30)$
$h(3)$	0.008100741	-0.013486594	$h(29)$
$h(4)$	0.002162997	-0.000510386	$h(28)$
$h(5)$	-0.020491495	0.017998923	$h(27)$
$h(6)$	0.011786882	0.00005630	$h(26)$
$h(7)$	0.031503889	-0.026686573	$h(25)$
$h(8)$	-0.01508075	-0.000846902	$h(24)$
$h(9)$	-0.04164009	0.036906698	$h(23)$
$h(10)$	0.012260194	-0.000695051	$h(22)$
$h(11)$	0.061312113	-0.059034272	$h(21)$
$h(12)$	0.00381118	-0.001105003	$h(20)$
$h(13)$	-0.105254853	0.102251935	$h(19)$
$h(14)$	-0.003504403	-0.000102292	$h(18)$
$h(15)$	0.311822989	-0.317422061	$h(17)$
$h(16)$	0.471041904	-0.500466051	$h(16)$

discussed in Table 6.9. From the table, it is observed that filter designed using CDE provides maximum stop band attenuation of $32.9994dB$ while DE yields maximum attenuation in stop band of $27.7594dB$.

Table 6.9: Parameters comparison for 20^{th} order high pass filter

Parameter	DE	CDE
Maximum pass band ripple (normalized)	0.151796	0.111813
Maximum stop band ripple (normalized)	0.040929	0.022389
Maximum stop band attenuation (dB)	27.7594	32.9994

The filter coefficients obtained for 20^{th} order high pass filter are tabulated in Table 6.10.

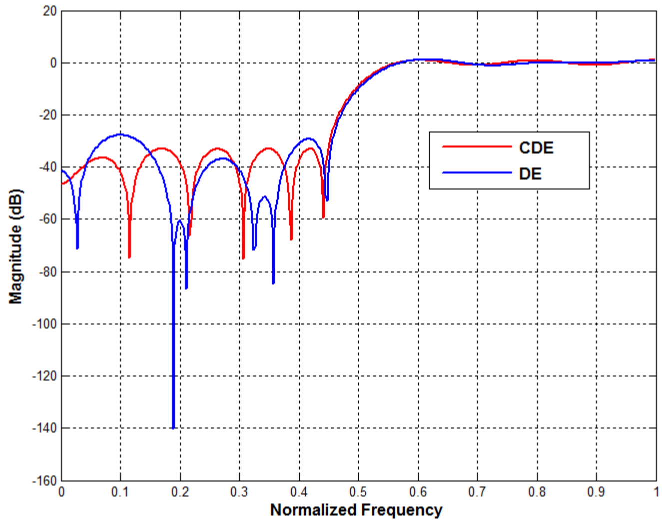


Figure 6-17: Magnitude response in dB for 20^{th} order high pass filter

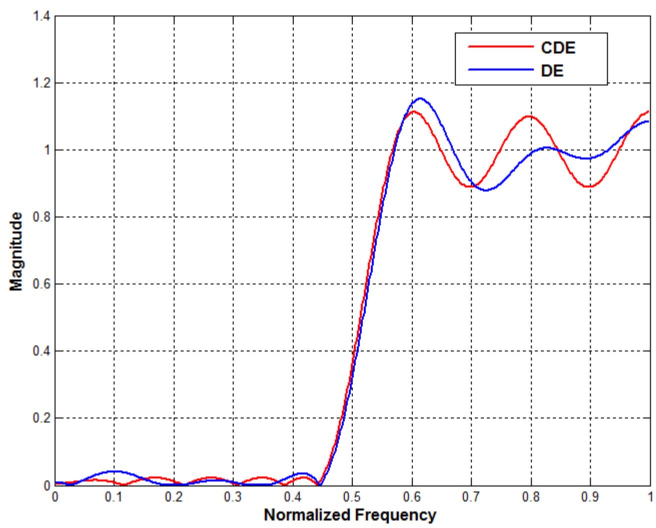


Figure 6-18: Normalized plot for 20^{th} order high pass filter

6.2.4 30^{th} order high pass filter

The magnitude response and normalized plot for 30^{th} order high pass filter is depicted in Fig. 6-19 and Fig. 6-20 respectively. The frequency response characteristics comparison for 30^{th} order high pass filter using DE and CDE algorithm is presented in Table 6.11. It is observed that filter designed using CDE provide higher attenuation in stop band *i.e.* 34.7571 dB while using DE stop attenuation achieved is 29.8366 dB. The filter coefficients obtained using DE and CDE approach for 30^{th} order high pass filter are tabulated in Table 6.12.

Table 6.10: Optimal filter coefficients for 20^{th} order high pass filter

$h(n)$	DE	CDE	$h(n)$
$h(1)$	0.024725224	-0.031857732	$h(21)$
$h(2)$	-0.019494602	0.040630277	$h(20)$
$h(3)$	-0.020092232	0.00693843	$h(19)$
$h(4)$	0.043659919	-0.036323933	$h(18)$
$h(5)$	0.020334707	-0.00753273	$h(17)$
$h(6)$	-0.071302036	0.059572815	$h(16)$
$h(7)$	-0.002870702	0.008696113	$h(15)$
$h(8)$	0.098085673	-0.100960694	$h(14)$
$h(9)$	0.011444699	-0.008729189	$h(13)$
$h(10)$	-0.319805077	0.316267781	$h(12)$
$h(11)$	0.479077303	-0.488635319	$h(11)$

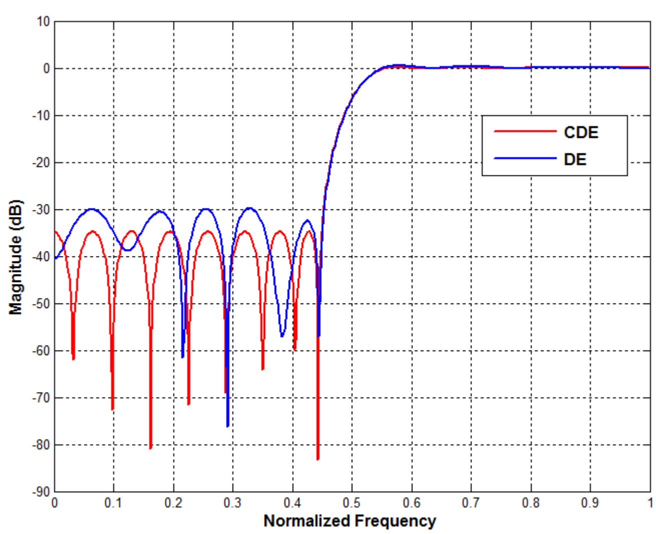


Figure 6-19: Magnitude response in dB for 30^{th} order high pass filter

Table 6.11: Parameters comparison for 30^{th} order high pass filter

Parameter	DE	CDE
Maximum pass band ripple (normalized)	0.059429	0.022317
Maximum stop band ripple (normalized)	0.032223	0.018287
Maximum stop band attenuation (dB)	29.8366	34.7571

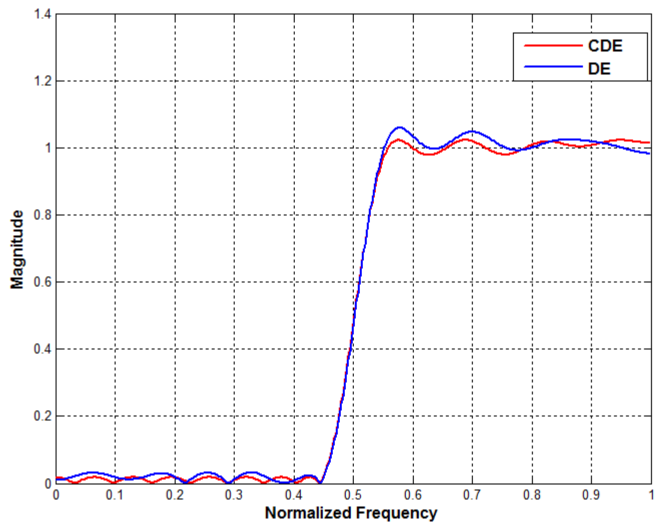


Figure 6-20: Normalized plot for 30^{th} order high pass filter

From all of the above discussion, it is well known that the frequency response reveals the superiority of the proposed CDE technique over the conventional DE technique for both the cases of high pass and low pass filter designing. The frequency response characteristics comparison for both high pass and low pass filter of 20^{th} and 30^{th} order shows that the filter designed using CDE approach provide better control over the ripples in stop band and pass band and provides more stop band attenuation than that designed using DE technique.

The comparison of both the algorithms engaged to optimize the coefficients for 20^{th} order low pass and high pass filters, in terms of error fitness value is shown in Fig. 6-21 and Fig. 6-22 respectively. The figure is plotted between the error fitness values versus the iteration values. Similar plots are obtained for 30^{th} order low pass and high pass filter. The convergence profile shows that CDE converges to lower fitness value in few number of iteration than DE algorithm. Therefore, it can be observed that CDE algorithm takes less computation time than DE algorithm for execution.

Table 6.12: Optimal filter coefficients for 30^{th} order high pass filter

$h(n)$	DE	CDE	$h(n)$
$h(1)$	0.009613338	0.012350149	$h(31)$
$h(2)$	0.00570878	0.003525865	$h(30)$
$h(3)$	-0.01566682	-0.014442359	$h(29)$
$h(4)$	-0.005224997	-0.000197562	$h(28)$
$h(5)$	0.019274438	0.018065758	$h(27)$
$h(6)$	-0.000350483	0.000866718	$h(26)$
$h(7)$	-0.022783397	-0.025521657	$h(25)$
$h(8)$	-0.001244285	-0.00186593	$h(24)$
$h(9)$	0.042130565	0.037962059	$h(23)$
$h(10)$	0.001627102	0.001968987	$h(22)$
$h(11)$	-0.061541154	-0.058921616	$h(21)$
$h(12)$	-0.005113899	-0.000489518	$h(20)$
$h(13)$	0.106029949	0.100987639	$h(19)$
$h(14)$	-0.001785102	0.003860517	$h(18)$
$h(15)$	-0.325051703	-0.31945887	$h(17)$
$h(16)$	0.499426605	0.500540819	$h(16)$

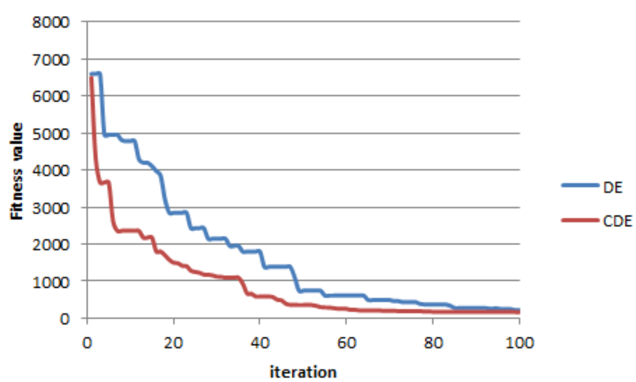


Figure 6-21: Convergence profile for 20^{th} order low pass filter

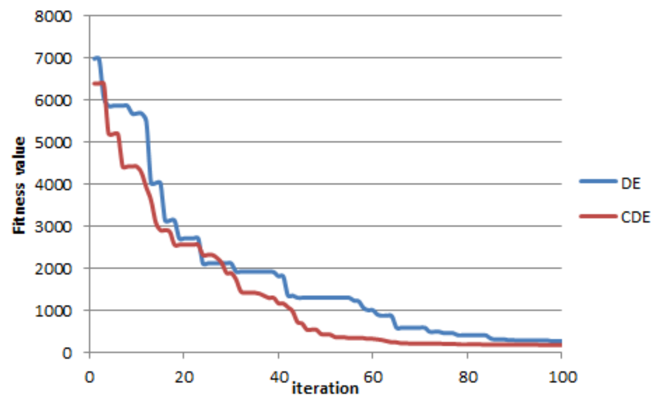


Figure 6-22: Convergence profile for 20^{th} order high pass filter

The proposed CDE algorithm performance is also compared with the results obtained by other researchers. Table 6.13 shows the comparison of results obtained using CDE algorithm with other results reported in literature.

Table 6.13: Comparison of proposed method results with other reported results

Algorithm	Parameters				
	Filter type	Order	Max. pass band ripple (normalized)	Max. stop band ripple (normalized)	Max. stop band attenuation (dB)
DE [2]	Low pass	20	>0.08	>0.09	-
PSO [13]	Low pass	30	0.15	0.031	<30
DEPSO [56]	Low pass	20	0.291	0.270	<27
eDE [58]	Low pass	20	0.04	>0.07	-
CRPSO [1]	High pass	20	0.126	0.0722	22.83
	High pass	30	0.139	0.0363	28.78
PSOCFIWA-WM [54]	High pass	20	0.150	0.0260	31.69
CSA [63]	High pass	20	0.020	0.0363	28.80
Proposed (CDE)	Low pass	20	0.115	0.0214	33.36
	High pass	20	0.112	0.0224	32.99
	Low pass	30	0.021	0.0185	34.67
	High pass	30	0.020	0.0192	34.32

From the table, it is observed that, the low pass filter of 20^{th} order designed using proposed CDE approach has lower pass band ripple than DEPSO [56] and lower stop band ripple than DE [2], DEPSO [56] algorithms. 30^{th} order low pass filter designed using CDE has higher stop band attenuation than PSO [13] algorithm. High pass filter of 20^{th} order designed using CDE has lower pass band ripples than CRPSO [1] and PSOCFIWA-WM [54] algorithms. Maximum stop band attenuation achieved is higher than CRPSO [1], PSOCFIWA-WM [54] and CSA [63] techniques. The 30^{th} order high pass filter using CDE has higher stop band attenuation and lower pass band and stop band ripples than CRPSO [1]. The comparison results reveals that proposed CDE algorithm provide maximum stop band attenuation and lower magnitude ripples in pass band and stop regions of filter.

CHAPTER 7

CASE STUDY

In this chapter, the application of chaotic differential evolution (CDE) algorithm to design filter for filtering the noise corrupted biomedical signal is discussed. The possible sources of noise for the biomedical signal can be sensors or signal conditioning circuits. Addition of noise into the signal distorts the signal quality and results in misleading interpretation of data. In this case study, electrocardiogram (ECG) signal is taken into consideration. Power line interference (PLI) noise introduced in ECG signal is removed using notch filter designed using CDE algorithm.

7.1 ECG signal

Electrocardiogram (ECG) is used to measure and record the electrical activity of human heart. Graphical recording of the electrical activity of heart helps in diagnosing various heart related diseases and abnormalities present. The typical ECG waveform of healthy person is shown in Fig. 7-1. ECG is recorded by placing the surface electrodes on the limbs or chest of the patient. The amplitude of signal recorded is of small magnitude. A typical range is 0.0001 to 0.003 volt and frequency range lies between 0.05 to 100 Hz [78]. ECG signals have only one form of morphological characteristics (P-QRS-T). The visual analysis of any variations in this morphology, makes it possible to diagnose many cardiac diseases. The major noise sources present in ECG signal are described below [79] :

Baseline Wander: The sources of baseline wander noise added into ECG trace include respiration, perspiration, body movements and poor electrode contacts *etc.* The wander noise added in ECG increases the amplitude of QRS complex. The spectral content of this noise added is confined to frequency range of 0.15 to 0.3 Hz *i.e* below 1 Hz.

Power Line Interference: PLI is high frequency noise present in ECG signal. The PLI noise is centered at 50/60 Hz. The main source of generation of this noise is the improper grounding of the ECG equipments.

Electromyographic noise: EMG noise is generated either due to the contraction of muscles along

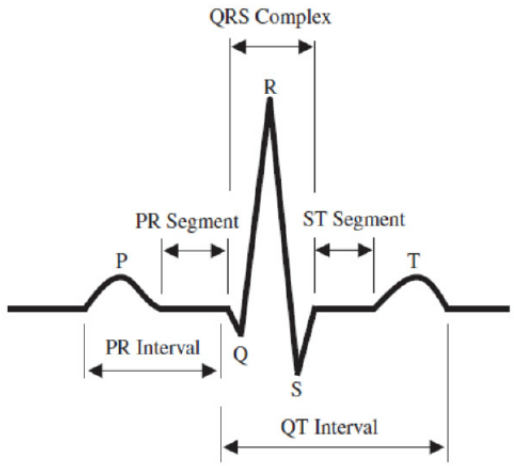


Figure 7-1: Typical ECG waveform

with the heart or by sudden movements of body. The spectral content of EMG noise signal is generally present in a frequency range from dc to 10 KHz. The frequency component of this noise overlaps with that of QRS complex of ECG and also extends into high frequencies.

7.2 Power line interference noise removal

The main source of noise in ECG is 50 or 60 Hz power line interference (PLI) noise [65]. It is called main source of noise because PLI noise lies in between the frequency band of ECG signal *i.e.* between 0.05 to 100 Hz. PLI noise effects the ST segment of ECG signal and creates problem for diagnosing the arrhythmia. Arrhythmia is any disturbance observed in the regular rhythmic activity of the heart. In order to remove the PLI noise from the ECG signal, the notch or band stop filter is the best solution. The notch filter filters out the 50 or 60 Hz frequency component from the noise corrupted signal.

7.3 Material and Method

The ECG signals used for testing are taken from MIT-BIH Arrhythmia Database [80]. The data base consists of 48 half hour sets of two channel ambulatory ECG recordings. Records were obtained from 47 subjects that includes 25 men with average age between 32 to 89 years and women with average age between 23 to 89 years. The ECG signal is obtained by placing the electrodes on the chest. In database records, upper signal is obtained from modified limb lead *II* and lower signal from modified lead V1, V2 or V5. The normal QRS complex are generally well known in upper signal. Each ECG signal record file was digitized by sampling frequency of 360Hz with 11-bit resolution over a 10 mV range [81]. In this case study, records of only four patients with lead *II* are considered for evaluation. Fig 7-2 shows the ECG signal for record no. 103 with lead *II*. The frequency spectrum of original ECG signal (record 103, lead *II*) is shown in Fig 7-3

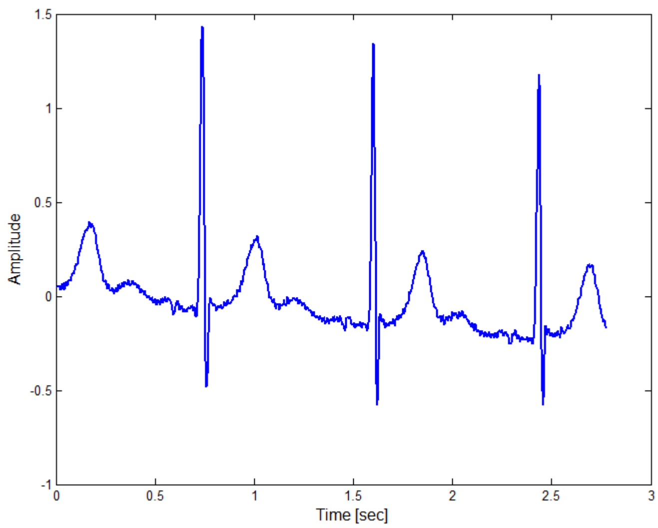


Figure 7-2: Original ECG signal in time domian

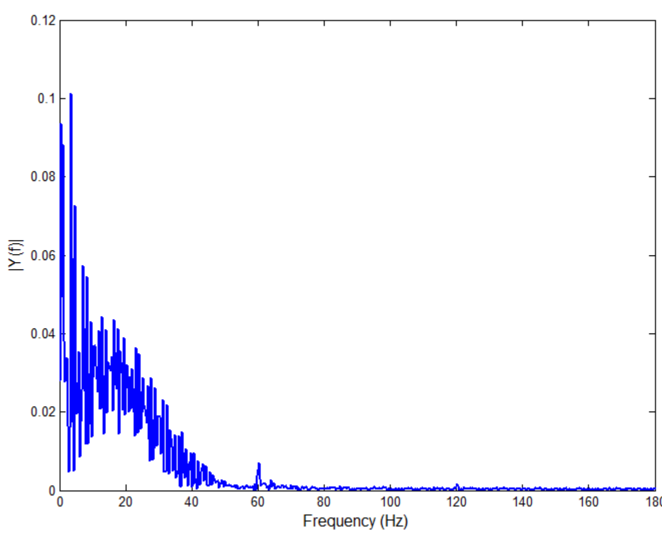


Figure 7-3: Original ECG signal in frequency domain

The PLI noise of 60Hz is generated using MATLAB. The peak to peak amplitude of noise level taken is 0.15mV. Mathematically, the 60Hz PLI noise $N(t)$ is represented as:

$$N(t) = A \sin 2\pi f t \tag{7.1}$$

where f is the frequency and A is the amplitude of PLI noise . The PLI noise corrupted signal is obtained by adding this PLI noise signal into the original ECG signal. The PLI noise corrupted ECG signal and it's frequency spectrum are shown in Fig 7-4 and 7-5 respectively. A spike is observed at 60 Hz frequency in frequency spectrum of noisy signal.

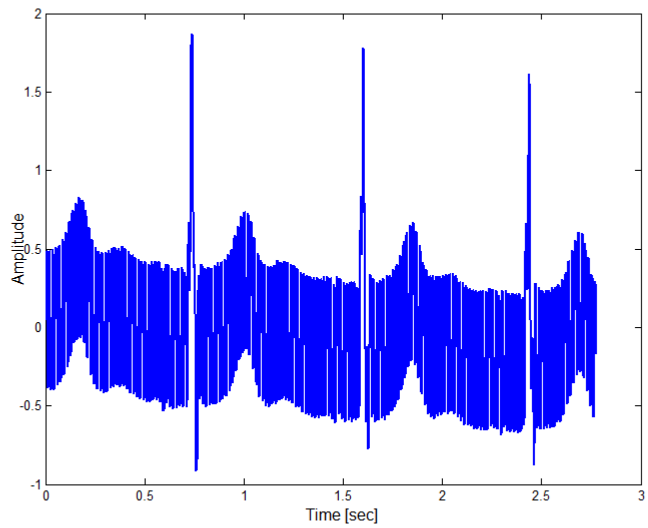


Figure 7-4: Noisy ECG signal in time domain

7.4 Design of Notch Filter

The chaotic differential evolution (CDE) algorithm and Differential evolution (DE) are used to design the notch filter with cut off frequency of 60 Hz. The sampling frequency has been taken 360 Hz. The parameters to design the band stop (notch) filter are: lower pass band edge frequency (normalized) (ω_{pl})=0.24 ; lower stop band edge frequency (normalized) (ω_{sl})=0.31 ; upper stop band edge frequency (normalized) (ω_{sl})=0.35 ; upper pass band edge frequency (normalized) (ω_{sl})=0.42. The 30^{th} order Band stop filter is designed using CDE. The values of control parameters *i.e* mutation rate (F) and crossover rate (CR) are taken 0.8 and 0.9 respectively. The algorithm is made to run for 2000 iterations and best optimal set of filter coefficients is selected out of 30

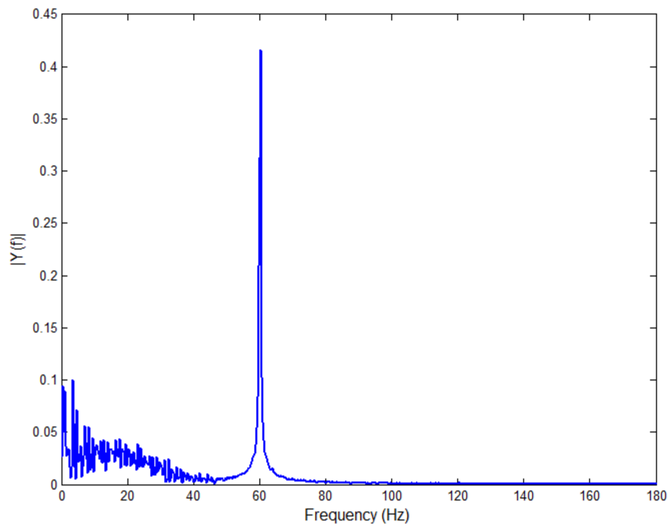


Figure 7-5: Noisy ECG signal in frequency domain

trial. The magnitude response in dB of 30^{th} order band stop filter designed using CDE and DE is shown in Fig 7-6.

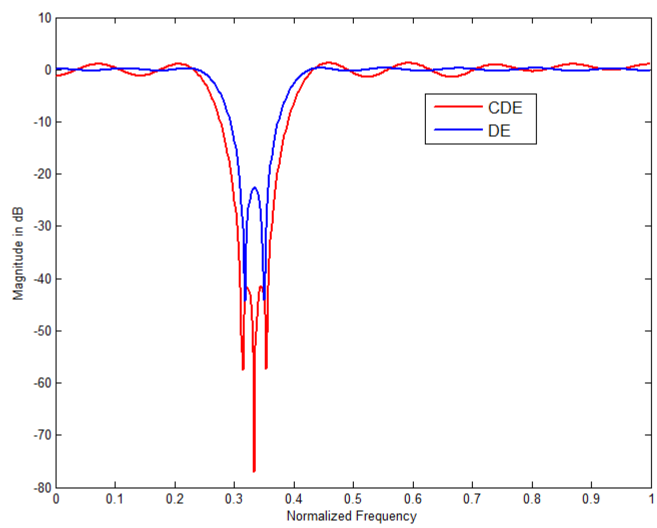


Figure 7-6: Magnitude response in dB for notch filter

7.5 Results and Discussion

The band stop also called notch filter designed using CDE and DE algorithm is applied to noise corrupted ECG signal. The filtered ECG signal waveform is shown in Fig 7-7. The frequency spectrum of filtered ECG signal is shown in Fig 7-8. It is observed that the amplitude of spike at

60Hz frequency in the frequency spectrum has been decreased. The PSD (Power spectral density)

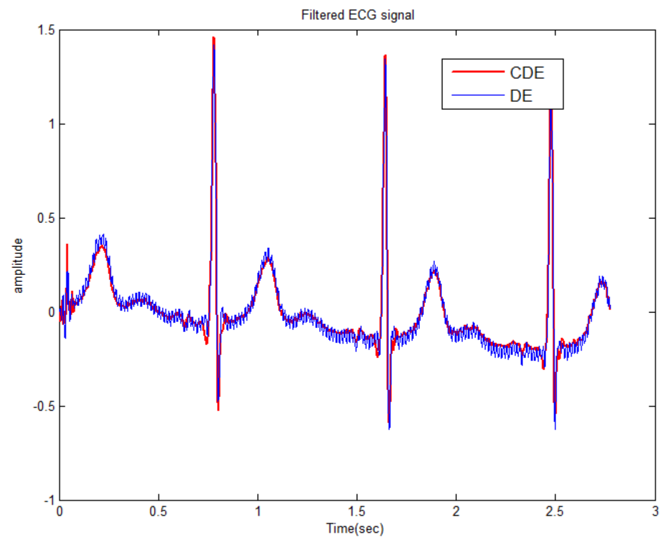


Figure 7-7: Filtered ECG signal in time domain

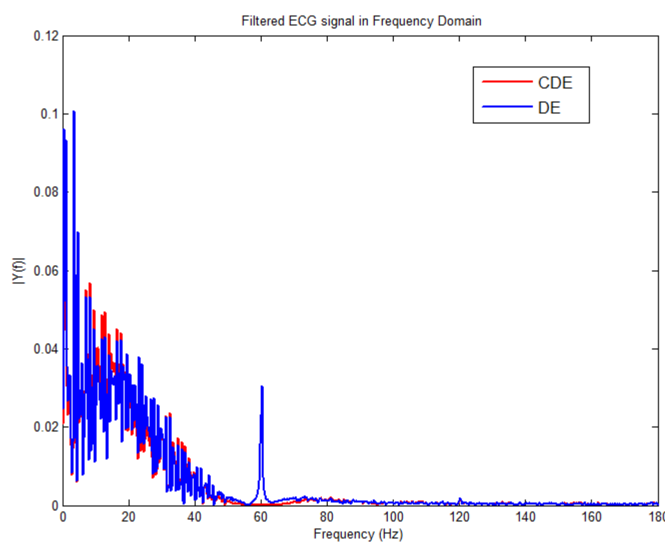


Figure 7-8: Filtered ECG signal in frequency domain

curves for noisy ECG signal and notch filtered signal is shown in Fig 7-9. The PSD curves shows that the power at 60 Hz frequency has been significantly decreased for filtered signal obtained using CDE.

The signal-to-noise ratio (SNR) is calculated to compare the performance of filtered signal with original noise free signal. The estimated values of performance parameters for noisy and filtered signal are shown in Table 7.1. It is well observed that SNR ratio increases for filtered signal

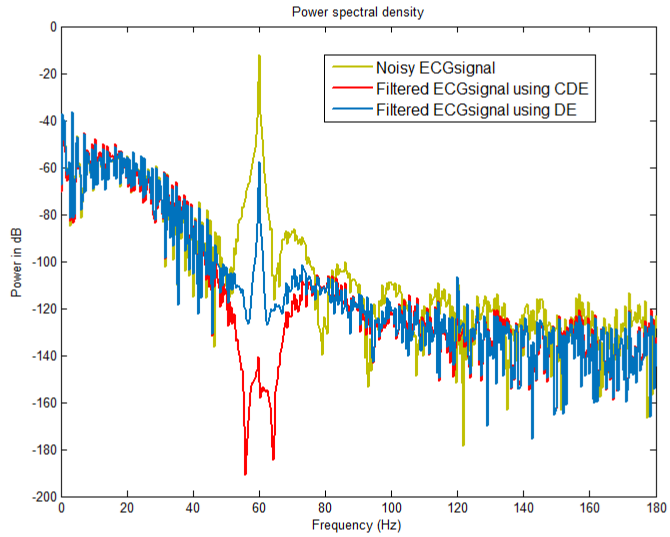


Figure 7-9: Power spectral density comparison between noisy and filtered ECG signal

obtained using filter designed using CDE algorithm and the values of MSE calculated between filtered and original ECG signal is very small. Also, the power density at 60 Hz frequency for filtered signal has been significantly decreased when filter is designed using CDE.

Table 7.1: Estimated values of performance parameters

Record	SNR (dB) (Noisy)	SNR(dB) (filtered signal)		Power of noisy (dB) (at 60 Hz)	Power of filtered signal(dB) (at 60 Hz)	
		DE	CDE		DE	CDE
100	4.1803	23.4058	57.4589	-11.99	-57.4	-127.8
101	2.6960	18.9034	59.2942	-12.06	-57.47	-140.8
102	11.7452	34.0356	59.5768	-11.79	-57.1	-111.9
103	3.0682	20.2798	67.2637	-12.22	-57.62	-151.5

CHAPTER 8

CONCLUSION

In this research work, a modified version of classical differential evolution (DE) algorithm called chaotic differential evolution (CDE) algorithm has been proposed. The basic difference between classical DE and CDE is that in DE the random number are generated by using uniformly distributed random number generator. Whereas in CDE, chaotic map sequences have been employed for randomization. The performance of proposed CDE is analyzed using standard test functions and finite impulse response (FIR) filter design problem. The nine standard test functions are considered for performance analysis namely; Sphere function, Rosenbrock's function, Schwefel 2.22 function, Rastrigin function, Schwefel function, Ackley function, Griewank function, Step function and Step2 function. In order to analyze the performance, the five process mutation schemes of DE have been implemented and their search capabilities are compared on the basis of their convergence rate. From the comparison of results of standard test functions, it is observed that DE/current-to-best/1 mutation scheme has better convergence then all other schemes taken into consideration. Similarly, the performance of CDE algorithm is analyzed for standard test functions using two chaotic sequences *i.e.* Tent map and Gauss map. Thirty independent trials are performed to minimize the considered test functions. On the basis of obtained results, it has been observed that CDE/current-to-best/1 mutation scheme converges to minimum value for most number of trials. Also, the performance of CDE/current-to-best/1 with Tent map is found better than with Gauss map. In order to validate the capability of CDE to solve real world engineering design problem, a FIR filter design problem is considered. Low pass and high pass FIR filters of 20^{th} and 30^{th} order are designed. The ripple constraint is introduced in order to reduce the magnitude of ripples present in pass band and stop band regions of filter. The ripple constraint is taken care of using penalty method. The filters are designed using CDE algorithm with Tent map sequence and CDE/current-to-best/1 mutation scheme. In order to compare the search capabilities of CDE, the filters are also designed using DE algorithm with DE/current-to-best/1 mutation scheme. The comparison of frequency response of filter coefficients obtained by CDE clearly indicates that the designed filter has met the expected specifications better than DE. CDE outperforms DE by providing higher value of maximum stop band attenuation and lower stop band and pass band ripple magnitude in frequency response. The performance of CDE is also tested while designing the filter for noise corrupted biomedical signal. Power line interference (PLI) noise effected electrocardiogram (ECG) signal has been considered in this study. Notch filter is designed using CDE algorithm to filter out the 60Hz noise from ECG signal. The filtered ECG signal characteristics are compared with original noise free signal. From the comparison of results obtained, it is observed that signal to noise ratio (SNR) of filtered signal increases and magnitude of power spectral density reduces at 60 Hz frequency. Hence, it is concluded that CDE based filter design procedure has the ability to design a filter as per desired specifications.

PUBLICATIONS

Harleen Kaur and NirbhowJap Singh, *”Digital Finite Impulse Response Low Pass Filter Design Using Differential Evolution Algorithm”* Journal of Multi Disciplinary Engineering Technologies (JMDET), vol 11, no.1, 2017. [Accepted]

BIBLIOGRAPHY

[1] S. Mandal, S. P. Ghoshal, R. Kar, and D. Mandal, "Design of optimal linear phase FIR high pass filter using craziness based particle swarm optimization technique," *Journal of King Saud University-Computer and Information Sciences*, vol. 24, no. 1, pp. 83–92, 2012.

[2] 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.

[3] R. Panda and U. C. Pati, "Removal of artifacts from electrocardiogram using digital filter," in *IEEE Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*. IEEE, 2012, pp. 1–4.

[4] V. Zschorlich, "Digital filtering of EMG-signals," *Electromyogr. clin, Neurophysiol*, vol. 29, pp. 81–86, 1989.

[5] J. Principe, J. Smith, S. Balakrishnan, and A. Paige, "Microcomputer-based digital filters for EEG processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 6, pp. 697–705, 1979.

[6] A. V. Oppenheim, *Discrete-time signal processing*. Pearson Education India, 1999.

[7] J. G. Proakis and D. Manolakis, "Digital signal processing, algorithms and applications," *Prentice-Hall, New-York*, 1995.

[8] M. B. Joaquim and C. A. Lucietto, "A nearly optimum linear-phase digital FIR filters design," *Digital signal processing*, vol. 21, no. 6, pp. 690–693, 2011.

[9] T. Çilolu, "An efficient local search method guided by gradient information for discrete coefficient FIR filter design," *Signal Processing*, vol. 82, no. 10, pp. 1337–1350, 2002.

[10] R. Salomon, "Evolutionary algorithms and gradient search: similarities and differences," *IEEE Transactions on Evolutionary Computation*, vol. 2, no. 2, pp. 45–55, 1998.

[11] X.-S. Yang, *Nature-inspired optimization algorithms*. Elsevier, 2014.

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

[13] 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.

[14] S. K. Saha, S. P. Ghoshal, R. Kar, and D. Mandal, "Cat swarm optimization algorithm for optimal linear phase FIR filter design," *International Society of Automation transactions*, vol. 52, no. 6, pp. 781–794, 2013.

[15] R. Kacelenga, P. Graumann, and L. Turner, "Design of digital filters using simulated annealing," in *IEEE International Symposium on Circuits and Systems*. IEEE, 1990, pp. 642–645.

[16] N. J. Singh, J. Dhillon, and D. Kothari, "Synergic predator-prey optimization for economic thermal power dispatch problem," *Applied Soft Computing*, vol. 43, pp. 298–311, 2016.

[17] R.Storn and K.Price, "Differential evolution–a simple and efficient adaptive scheme for global optimization over continuous spaces: technical report TR-95-012," *International Computer Science, Berkeley, California*, 1995.

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

[19] N. Karaboğa and B. Çetinkaya, "Efficient design of fixed point digital FIR filters by using differential evolution algorithm," *Computational Intelligence and Bioinspired Systems*, pp. 812–819, 2005.

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

[21] Z. Albataineh, J. I. Ababneh, and F. Salem, "Linear phase FIR low pass filter design using hybrid-differential evolution," *International Journal of Research in Wireless Systems*, vol. 1, no. 2, 2012.

[22] W.-J. Zhang and X.-F. Xie, "DEPSO: hybrid particle swarm with differential evolution operator," in *IEEE Conf. Syst. Man Cybern.*, vol. 4, 2003, pp. 3816–3821.

[23] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, 2009.

[24] J. Liu and J. Lampinen, "A fuzzy adaptive differential evolution algorithm," *Soft Comput.*, vol. 9, no. 6, pp. 448–462, 2005.

[25] D. Yang, G. Li, and G. Cheng, "On the efficiency of chaos optimization algorithms for global optimization," *Chaos, Solitons & Fractals*, vol. 34, no. 4, pp. 1366–1375, 2007.

[26] X. F. Yan, D. Z. Chen, and S. X. Hu, "Chaos-genetic algorithms for optimizing the operating conditions based on RBF-PLS model," *Comput. Chem. Eng.*, vol. 27, no. 10, pp. 1393–1404, 2003.

[27] J. Cai, X. Ma, Q. Li, L. Li, and H. Peng, "A multi-objectivechaotic particle swarm optimization for environmental/economic dispatch," *Energy Convers. Manage.*, vol. 50, no. 5, pp. 1318–1325, 2009.

[28] W. Gong and S. Wang, "Chaos ant colony optimization and application," in *Fourth Int. Conf. Internet Comput. Sci. Eng. (ICICSE), 2009.* IEEE, 2009, pp. 301–303.

[29] A. H. Gandomi and X.-S. Yang, "Chaotic bat algorithm," *J. Comput. Sci. Eng.*, vol. 5, no. 2, pp. 224–232, 2014.

[30] L. dos Santos Coelho, J. G. Sauer, and M. Rudek, "Differential evolution optimization combined with chaotic sequences for image contrast enhancement," *Chaos Solitons Fractals*, vol. 42, no. 1, pp. 522–529, 2009.

[31] N. J. Singh, J. Dhillon, and D. Kothari, "Multi-objective thermal power load dispatch using chaotic differential evolutionary algorithm and powells method," *Soft Comput.*, pp. 1–16, 2017.

[32] Y. Lu, J. Zhou, H. Qin, Y. Wang, and Y. Zhang, “Chaotic differential evolution methods for dynamic economic dispatch with valve-point effects,” *Eng. Appl. Artif. Intell.*, vol. 24, no. 2, pp. 378–387, 2011.

[33] L. Rabiner, “Techniques for designing finite-duration impulse-response digital filters,” *IEEE transactions on communication technology*, vol. 19, no. 2, pp. 188–195, 1971.

[34] J. W. Gibbs, “Fouriers series,” *Nature*, vol. 59, no. 1522, p. 200, 1898.

[35] J.W.Gibbs, “Fouriers series,” *Nature*, vol. 59, no. 1539, p. 606, 1899.

[36] T. K. Roy and M. Morshed, “Performance analysis of low pass FIR filters design using kaiser, gaussian and tukey window function methods,” in *IEEE International Conference on Advances in Electrical Engineering*, 2013, pp. 1–6.

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

[38] 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.

[39] L. D. Grossmann and Y. C. Eldar, “An l_1-method for the design of linear-phase FIR digital filters,” *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5253–5266, 2007.

[40] A. Aggarwal, T. Rawat, M. Kumar, and D. Upadhyay, “Design of optimal band-stop FIR filter using l_1-norm based RCGA,” *Ain Shams Engineering Journal*, 2016.

[41] A. Aggarwal, M. Kumar, and T. K. Rawat, “L_1 error criterion based optimal FIR filters,” in *Annual IEEE India conference (INDICON)*. IEEE, 2014, pp. 1–6.

[42] J. Cadzow, “Recursive digital filter synthesis via gradient based algorithms,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 5, pp. 349–355, 1976.

[43] Vasundhara, D. Mandal, S. P. Ghoshal, and R. Kar, “Digital FIR filter design using hybrid random particle swarm optimization with differential evolution,” *International Journal of Computational Intelligence Systems*, vol. 6, no. 5, pp. 911–927, 2013.

[44] İ. Durgun and A. R. Yildiz, "Structural design optimization of vehicle components using cuckoo search algorithm," *Materials Testing*, vol. 54, no. 3, pp. 185–188, 2012.

[45] N. Noman and H. Iba, "Differential evolution for economic load dispatch problems," *Electric Power Systems Research*, vol. 78, no. 8, pp. 1322–1331, 2008.

[46] J. Robinson and Y. Rahmat-Samii, "Particle swarm optimization in electromagnetics," *IEEE transactions on antennas and propagation*, vol. 52, no. 2, pp. 397–407, 2004.

[47] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," *Machine learning*, vol. 3, no. 2, pp. 95–99, 1988.

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

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

[50] P. J. Van Laarhoven and E. H. Aarts, "Simulated annealing," in *Simulated Annealing: Theory and Applications*. Springer, 1987, pp. 7–15.

[51] G. Wade, P. Van-Eetvelt, and H. Darwen, "Synthesis of efficient low-order FIR filters from primitive sections," *IEE Proceedings G (Circuits, Devices and Systems)*, vol. 137, no. 5, pp. 367–372, 1990.

[52] K. Boudjelaba, F. Ros, and D. Chikouche, "An efficient hybrid genetic algorithm to design finite impulse response filters," *Expert Systems with Applications*, vol. 41, no. 13, pp. 5917–5937, 2014.

[53] W. Fang, J. Sun, W. Xu, and J. Liu, "FIR digital filters design based on quantum-behaved particle swarm optimization," in *Innovative Computing, Information and Control, 2006. ICICIC'06. First International Conference on*, vol. 1. IEEE, 2006, pp. 615–619.

[54] S. K. Saha, R. Kar, D. Mandal, and S. P. Ghoshal, "Optimal linear phase FIR filter design using particle swarm optimization with constriction factor and inertia weight approach with

wavelet mutation," *International Journal of Hybrid Intelligent Systems*, vol. 11, no. 2, pp. 81–96, 2014.

[55] B. Luitel and G. K. Venayagamoorthy, "Particle swarm optimization with quantum infusion for the design of digital filters," in *Swarm Intelligence Symposium, 2008. SIS 2008. IEEE*. IEEE, 2008, pp. 1–8.

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

[57] Z. Zhao, H. Gao, and Y. Liu, "Chaotic particle swarm optimization for FIR filter design," in *International Conference on Electrical and Control Engineering (ICECE)*. IEEE, 2011, pp. 2058–2061.

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

[59] N. Karaboga, "Digital IIR filter design using differential evolution algorithm," *EURASIP Journal on Applied Signal Processing*, vol. 2005, pp. 1269–1276, 2005.

[60] B. Singh, J. Dhillon, and Y. Brar, "A hybrid differential evolution method for the design of IIR digital filter," *ACEEE International Journal of Signal & Image Processing*, vol. 4, no. 1, pp. 1–10, 2013.

[61] S. Saha, R. Dutta, R. Choudhury, R. Kar, D. Mandal, and S. Ghoshal, "Efficient and accurate optimal linear phase FIR filter design using opposition-based harmony search algorithm," *The Scientific World Journal*, vol. 2013, 2013.

[62] 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.

[63] A. Aggarwal, T. K. Rawat, and D. K. Upadhyay, "Design of optimal digital FIR filters using evolutionary and swarm optimization techniques," *AEU-International Journal of Electronics and Communications*, vol. 70, no. 4, pp. 373–385, 2016.

[64] A. Mittal and A. Rege, "Design of digital FIR filter implemented with window techniques for reduction of power line interference from ECG signal," in *International Conference on Computer, Communication and Control (IC4)*. IEEE, 2015, pp. 1–4.

[65] U. Biswas and M. Maniruzzaman, "Removing power line interference from ECG signal using adaptive filter and notch filter," in *Electrical Engineering and Information & Communication Technology (ICEEICT), 2014 International Conference on*. IEEE, 2014, pp. 1–4.

[66] J.-M. Renders and S. P. Flasse, "Hybrid methods using genetic algorithms for global optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 2, pp. 243–258, 1996.

[67] X.-l. Dong, S.-q. Liu, T. Tao, S.-p. Li, and K.-l. Xin, "A comparative study of differential evolution and genetic algorithms for optimizing the design of water distribution systems," *Journal of Zhejiang University-Science A*, vol. 13, no. 9, pp. 674–686, 2012.

[68] L. Arya, S. Choube, and R. Arya, "Differential evolution applied for reliability optimization of radial distribution systems," *International Journal of Electrical Power & Energy Systems*, vol. 33, no. 2, pp. 271–277, 2011.

[69] M. Iwan, R. Akmeliawati, T. Faisal, and H. M. Al-Assadi, "Performance comparison of differential evolution and particle swarm optimization in constrained optimization," *Procedia Engineering*, vol. 41, pp. 1323–1328, 2012.

[70] A. Deb, J. S. Roy, and B. Gupta, "Performance comparison of differential evolution, particle swarm optimization and genetic algorithm in the design of circularly polarized microstrip antennas," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 8, pp. 3920–3928, 2014.

[71] F. Neri and V. Tirronen, "Recent advances in differential evolution: a survey and experimental analysis," *Artificial Intelligence Review*, vol. 33, no. 1-2, pp. 61–106, 2010.

[72] H.-Y. Fan and J. Lampinen, “A trigonometric mutation operation to differential evolution,” *J. Global Optim.*, vol. 27, no. 1, pp. 105–129, 2003.

[73] W. Gong, Z. Cai, and C. X. Ling, “ DE/BBO: a hybrid differential evolution with biogeography-based optimization for global numerical optimization,” *Soft Comput.*, vol. 15, no. 4, pp. 645–665, 2010.

[74] A. Glotić, A. Glotić, P. Kitak, J. Pihler, and I. Tičar, “Parallel self-adaptive differential evolution algorithm for solving short-term hydro scheduling problem,” *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2347–2358, 2014.

[75] S. Das, S. S. Mullick, and P. N. Suganthan, “Recent advances in differential evolution–an updated survey,” *Swarm and Evolutionary Computation*, vol. 27, pp. 1–30, 2016.

[76] M. Jamil and X.-S. Yang, “A literature survey of benchmark functions for global optimisation problems,” *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 4, no. 2, pp. 150–194, 2013.

[77] X.-S. Yang, “Appendix A: test problems in optimization,” *Engineering optimization*, pp. 261–266, 2010.

[78] M. Islam, G. Tangim, T. Ahammad, and M. Khondokar, “Study and analysis of ECG signal using matlab & labview as effective tools,” *International journal of Computer and Electrical engineering*, vol. 4, no. 3, p. 404, 2012.

[79] P. Shetty and S. Bhat, “Analysis of various filter configurations on noise reduction in ECG waveform,” *International Journal of Computing, Communications & Instrumentation Engg.(IJCCIE)*, vol. 1, no. 1, pp. 88–91, 2014.

[80] R. Mark and G. Moody, “MIT-BIH arrhythmia database directory,” *Cambridge: Massachusetts Institute of Technology*, 1988.

[81] https://physionet.org/physiobank/database/mitdb/, [accessed 03-04-2017].

Harleen_05-07-2017

ORIGINALITY REPORT

%13	%4	%13	%5
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Sangeeta Mandal. "Swarm intelligence based optimal linear fir high pass filter design using Particle Swarm Optimization with Constriction Factor and Inertia Weight Approach", 2011 IEEE Student Conference on Research and Development, 12/2011 Publication	%1
2	Submitted to Thapar University, Patiala Student Paper	%1
3	Lecture Notes in Computer Science, 2013. Publication	%1
4	Sharma, Shekhar, L. D. Arya, and Sumant Katiyal. "Design of linear-phase digital FIR filter using Differential Evolution optimization with ripple constraint", 2014 International Conference on Computing for Sustainable Global Development (INDIACom), 2014. Publication	<%1
5	Lecture Notes in Computer Science, 2015. Publication	<%1