4-13-2017

Linear Phase FIR Digital Filter Design Using Differential Evolution Algorithms

Wei Zhong
University of Windsor

Linear Phase FIR Digital Filter Design Using Differential Evolution Algorithms

by

Wei Zhong

A Thesis

Submitted to the Faculty of Graduate Studies

through Electrical and Computer Engineering

in Partial Fulfillment of the Requirements for

the Degree of Master of Applied Science at the

University of Windsor

Windsor, Ontario, Canada

2016

Linear Phase FIR Digital Filter Design Using Differential Evolution Algorithms

by

Wei Zhong

APPROVED BY:

Dr. Z. Kobti

School of Computer Science

Dr. Huapeng Wu

Department of Electrical & Computer Engineering

Dr. H. K. Kwan, Advisor

Department of Electrical and Computer Engineering

December 22, 2016

Declaration of Originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies Office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

Digital filter plays a vital part in digital signal processing field. It has been used in control systems, aerospace, telecommunications, medical applications, speech processing and so on. Digital filters can be divided into infinite impulse response filter (IIF) and finite impulse response filter (FIR). The advantage of FIR is that it can be linear phase using symmetric or anti-symmetry coefficients. Besides traditional methods like windowing function and frequency sampling, optimization methods can be used to design FIR filters. A common method for FIR filter design is to use the Parks-McClellan algorithm. Meanwhile, evolutional algorithm such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO) [2], and Differential Evolution (DE) have shown successes in solving multi-parameters optimization problems.

This thesis reports a comparison work on the use of PSO, DE, and two modified DE algorithms from [18] and [19] for designing six types of linear phase FIR filters, consisting of type1 lowpass, highpass, bandpass, and bandstop filters, and type2 lowpass and bandpass filters. Although PSO has been applied in this field for some years, the results of some of the designs, especially for high-dimensional filters, are not good enough when comparing with those of the Parks-McClellan algorithm. DE algorithms use parallel search techniques to explore optimal solutions in a global range. What's more, when facing higher dimensional filter design problems, through combining the knowledge acquired during the searching process, the DE algorithm shows obvious advantage in both frequency response and computational time.

Acknowledgements

I would like to express my sincere gratitude and appreciation to Dr. H. K. Kwan, who introduced me the subject of evolutionary optimization and various optimization algorithms for designing digital filters through an excellent set of graduate courses, assignments, and projects, for his invaluable guidance, helps, and suggestions throughout the thesis work. Also special thanks to Dr. Huapeng Wu and Dr. Z. Kotbi, for their expert guidance and advices in my research. What's more, I would also like to thank my family for their constant and endless love and my friends for their helps and friendships.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ACO	Ant Colony Optimization
CM	Crossover and Mutation
CMAES	Covariance Matrix Adaptation Evolutional Strategy
DE	Differential Evolution
DSA	Differential Search Algorithm
EA	Evolutionary Algorithm
ES	Evolutionary Strategies
GA	Genetic Algorithm
LPF	Low Pass Filter
MATLAB	Matrix Laboratory
MNA	Modified Nodal Analysis
PSO	Particle Swarm Optimization
TSP	Travelling Sales Problem
MPMSIDE	Multi- Population and Multi- Strategy Differential Evolution Algorithm

Chapter 1 Introduction to Linear Phase Digital FIR Filter Design

1.1 Introduction

A digital filter is a system that alters an incoming signal in the desired way to extract useful information and discard undesirable components. Digital filters are used pervasively in wide ranging products. Digital filter plays a vital part in digital signal processing field. It is used in control systems, aerospace, telecommunications, medical applications, speech processing and so on. Comparing with analog filters, digital filters have various advantages like:

- Digital filters do not suffer from components tolerances, and their response is invariant to temperature and time.
- Digital filters can be programmed easily on digital hardware
- Digital filters are insensitive to electrical noise to a great extent
- Digital filters are very versatile in the desired responses they can produce

Normally, it is divided into infinite impulse response filter (IIF) and finite impulse response filter (FIR). The advantage of FIR is that it can complete linear phase while satisfying the condition of symmetric coefficients. Except for tradition ways like windowing function and frequency sampling, researchers have supplied many optimization methods to design FIR filters.

This section will introduce the digital FIR filter design problem and the concepts associated. The specification of the digital filter design based on DE optimization problem and the solution techniques will also be discussed. Lastly, the implementation of digital FIR filters will be presented.

1.2 FIR digital filter

FIR is short for finite impulse response and is also called feedforward or non-recursive, or transversal filter. This kind of digital filter exhibits a finite duration impulse response. For a FIR filter whose impulse response of length N=R+1, R being the order, is given by **h**= $[h_0\ h_1\ h_2 \ldots h_{N-1}]^T$.

Consider a FIR filter of length *M* (*M*=*R-1*). Its efficient can obtain the impulse response, with $x(n) = \delta(n)$.

$$h(n) = \sum_{k=0}^{M-1} b_k \delta(n-k) = b_n \tag{1.1}$$

Note that FIR filters only have zeros (no poles), so that it is also known as all-zero filters. [4]

The transfer function or the z-transform can be written as equation (1.2).

$$H(\mathrm{z}) = C(z) = \sum_{n=0}^{N} c(n) z^{-n} \tag{1.2}$$

Where $C(z)$ denotes a polynomial written in ascending powers ofz^{-1}. The coefficients $c(n)$ for $n \geq 0$ represent the impulse response values of the FIR digital filter. By substituting $z = e^{jwT}$ into equation (1.1), the frequency response of a FIR digital filter can be transferred as

$$H(w) = \sum_{n=0}^{N} c(n)\cos nwT - j\sum_{n=0}^{N} c(n)\sin nwT = |H(w)|e^{\theta(w)} \tag{1.3}$$

$|H(w)|$ is the magnitude response and $\theta(w)$ is the filter's phase response.

$$|H(w)| = \left\{\left[\sum_{n=0}^{N} c(n)\cos nwT\right]^2 + \left[\sum_{n=0}^{N} c(n)\sin nwT\right]^2\right\}^{1/2} \tag{1.4}$$

$$\theta(w) = -\tan^{-1}\left[\frac{\sum_{n=0}^{N} c(n)\sin nwT}{\sum_{n=0}^{N} c(n)\cos nwT}\right] \tag{1.5}$$

The group delay $\tau(w)$ of filter is defined as is defined in equation (1.6)

$$\tau(w) = -\frac{\partial\theta(w)}{\partial wT} \tag{1.6}$$

1.3 Symmetric Filters

Due to the symmetric characteristic of the linear phase filters, the first and last coefficients, the second and next to last can be rearranged to calculate the frequency response of the direct form FIR filter in this way:

$$\begin{aligned} H(\exp(j\Omega)) &= \sum_{k=0}^{M} b_k \exp(-jk\Omega) \\ &= b_0\exp(-j0) + b_M\exp(-jM\Omega) + b_1\exp(-j\Omega) \\ &\quad + b_{M-1}\exp(-j(M-1)\Omega) \end{aligned} \tag{1.7}$$

And then taking out a common factor $\exp(-jM\Omega/2)$

$$
\begin{aligned}
H(\exp(j\Omega)) = \exp(-jM\Omega/2) & \\
\times \{b_0\exp(jM\Omega/2) + b_M\exp(-jM\Omega/2) & \\
+ b_1\exp(-j(M-2)\Omega/2) + b_{M-1}\exp(-j(M-1)\Omega/2) + \cdots\} &
\end{aligned} \tag{1.8}
$$

If the filter length $M+1$ is odd, then the final term in the brackets is the only term of the middle one$b_{M/2}$, which is the center coefficient of the designed filter.

Symmetric impulse response

Assuming the coefficients$b_0 = b_M$, $b_1 = b_{M-1}$, etc. And $\exp(j\theta) + \exp(-j\theta) = 2\cos(\theta)$, then the frequency response should be

$$
\begin{aligned}
H(\exp(j\Omega)) = \exp(-jM\Omega/2) & \\
\times \{b_0\cos(M\Omega/2) + b_1\cos\big((M-2)\Omega/2\big) + \cdots\} &
\end{aligned} \tag{1.9}
$$

We could notice the equation above is a purely real function multiplied by a linear phase term. Therefore the response has linear phase, and the delay is $M/2$, which is the half of the filter length.

1.4 Linear phrase filters

The ability to have an exactly linear phase is one of the most important duties of FIR filter design.

Generally, a filter is not a linear phase, unless it satisfies the following condition.

$$h(n) = \pm h(M-1-n), \quad n = 0,1,\dots,M-1 \tag{1.10}$$

The following are four types of linear phase filters

Type 1 - M odd and even symmetry: the impulse response is $h(n)$,

$$h(n) = h(M-1-n)$$

The transfer function can be represented as

$$H(w) = e^{-\frac{jw(M-1)}{2}}\left(h\left(\frac{M-1}{2}\right) + 2\sum_{n=0}^{(M-3)/2} h(n)\cos(\frac{M-1}{2} - n)wT\right) \tag{1.11}$$

Type 2 - M even and even symmetry: the impulse response is $h(n)$,

$$h(n) = h(M-1-n)$$

The transfer function can be represented as

$$H(w) = e^{-\frac{jw(M-1)T}{2}} 2 \sum_{n=0}^{\frac{M}{2}-1} h(n) \cos\left(\frac{M-1}{2} - n\right) wT \quad (1.12)$$

Type 3- M odd and odd symmetry: the impulse response is $h(n)$,

$$h(n) = -h(M-1-n)$$

The transfer function can be represented as

$$H(w) = je^{-j[\frac{w(M-1)}{2}]} \left(2 \sum_{n=0}^{(M-3)/2} h(n) \sin(\frac{M-1}{2} - n) wT \right) \quad (1.13)$$

Type 4 – M odd and even symmetry: the impulse response is $h(n)$

$$h(n) = -h(M-1-n)$$

The transfer function can be represented as

$$H(w) = je^{-j[\frac{wT(M-1)}{2}]} 2 \sum_{n=0}^{(M-1)/2} h(n) \sin\left(\frac{M-1}{2} - n\right) wT \quad (1.14)$$

This paper concentrates mainly on Types 1 and 2. These filter types are used for conventional filtering applications. In both of the two types, the group delay is $\tau(\omega) = \frac{\partial\left[-\left(\frac{M-1}{2}\right)wT\right]}{\partial wT} = N/2$, which is only decided by the frequency ω. Types of the remaining two filters have an additional 90-degree phase shift so that they are more suitable for designing filters like differentiators and Hilbert transformers.

1.5 Digital FIR filter design

Digital filter design usually involves the following basic steps:

1. Determine the desired response or a set of desired responses (In this paper we only design a FIR filter with desired magnitude response, the phase is only connected with the type of the designed filter).

2. Use a type of filter for approximating the desired response (e.g., linear phase FIR filter).

3. Establish a criterion of "satisfying" for the desired result of a filter in the selected solution compared to the desired response.

4. Propose and improve a method to find the ideal filter.

5. Synthesize the algorithm with suitable computing equipment.

6. Compare and analyze the filter performance.

All the four types of linear FIR filters can be achieved by using the properties of their coefficients symmetry. Usually, digital filters design involves the four main steps: approximation, realization, quantisation consideration and implementation. By using software simulating, the proper specifications such as amplitude response and phase properties can be completed [4].

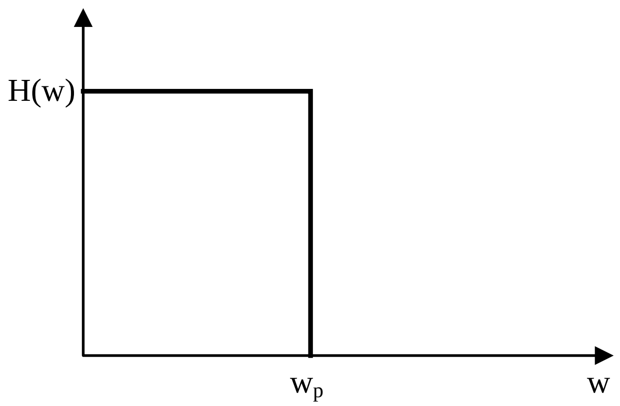


Fig. 1.1 Ideal lowpass digital FIR filter

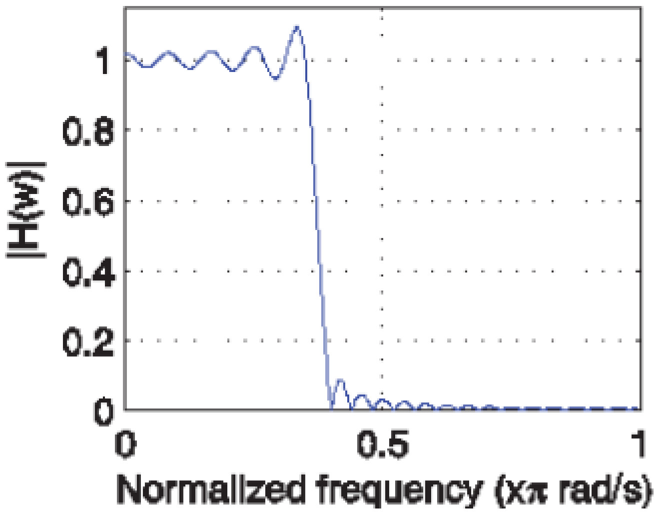


Fig. 1.2 Actual lowpass FIR filter

To design a digital FIR filter, we need to formulate the filter coefficient vector ***c*** from the transfer functions a filter. Besides, cost function like weighted least-squares (WLS) and minimax (MM) are also used to optimize the question.

General FIR digital filters

The transfer function of a general *N*th-order FIR digital filter can be expressed as

$$H(z) = \sum_{n=0}^{N} h_n z^{-n} = \boldsymbol{c}^T \boldsymbol{z}(z) \tag{1.15}$$

$$\boldsymbol{c} = [h_0, h_1, h_2, \dots, h_N]^T \tag{1.16}$$

$$\boldsymbol{z}(z) = [1, z^{-1}, z^{-2}, \dots, z^{-N}]^T \tag{1.17}$$

The vector $\boldsymbol{c}$ in equation (1.16) denotes a filter coefficient vector of dimension (*N*+*1*)×1. Using e^{jw} to replace z in (1.15), the frequency response of the general *N*th-order FIR digital filter can be expressed as

$$H(w) = \sum_{n=0}^{N} h_n e^{-jwn} = \boldsymbol{c}^T \boldsymbol{z}(w) \tag{1.18}$$

$$z(w) = [1, e,^{-jw}, e,^{-j2w}, \dots, e,^{-jNw}]^T \tag{1.19}$$

A FIR digital filter can be specified by the following parameters: the order of the filter, passband cutoff frequency, stopband cutoff frequency, passband and stopband ripple error. Like the Parks-Macllenan method, it uses freqz.m [5] to optimize the digital filter coefficients.

The optimization is aiming to reduce the ripple error of the passband and stopband while keeping a sharp transition band, which can be expressed as

$$\text{minimize: } \delta \tag{1.20}$$

$$\text{subject to } e(\boldsymbol{c}) \le \delta \tag{1.21}$$

An ideal lowpass FIR digital owing an ideal response specification.

$$H(w) = \begin{cases} 1 & 0 \le w \le w_p \\ 0 & w_s \le w \le 1 \end{cases} \tag{1.22}$$

To achieve a minimum value of δ, which is defined as

$$\delta = W(w)|(H(w) - D(w))| \tag{1.23}$$

Where $W\ (w)$, the weighting function can be expressed as

$$W(w) = \begin{cases} 1 & 0 \le w \le w_p \\ 0 & w_p \le w \le w_s \\ 1 & w_s \le w \le 1 \end{cases} \tag{1.24}$$

1.6 Problem Formulation

An Nth-order non-recursive digital FIR filter can be represented by the transfer function

$$H(z) = \sum_{n=0}^{N} h_n z^{-n} = \boldsymbol{c}^T \boldsymbol{z}(z) \tag{1.25}$$

Where $\boldsymbol{c}^T$ is real coefficients vector. N is the total number of filter coefficients. N-1 is its order. For optimization problem, the coefficients vector is

$$\boldsymbol{c}^T = [c_1 c_2 \dots c_N] \tag{1.26}$$

The frequency response can be gained by substituting $z = e^{jTw}$, where T is the sampling period in seconds and w is the frequency.

For the design of a linear-phase FIR digital filter, assume $\Omega = w_i$, $1 \le i \le M$, be the group of frequencies to evaluate the frequency response. Therefore, the error at each sample point in w_i is represented as

$$e_i = H_d(w_i) - H(w_i) \tag{1.27}$$

Where $H_d(w_i)$ is the ideal digital filter frequency response and $H\ (w_i)$ represents the actual frequency response of the designed filter. Because we only design symmetric FIR filter, the group delay is constant as defined as

$$\tau = \frac{N}{2} \tag{1.28}$$

1.7 Coefficients initializing method

As for the coefficients of digital FIR filter have a specific property as shown in Fig. 1.3, an exponential function to approximate them could improve the optimizing efficiency. In this experiment, use an exponential function to initialize the coefficients. As we can find the coefficients of a FIR filter is symmetric, normally it is like the following trend.

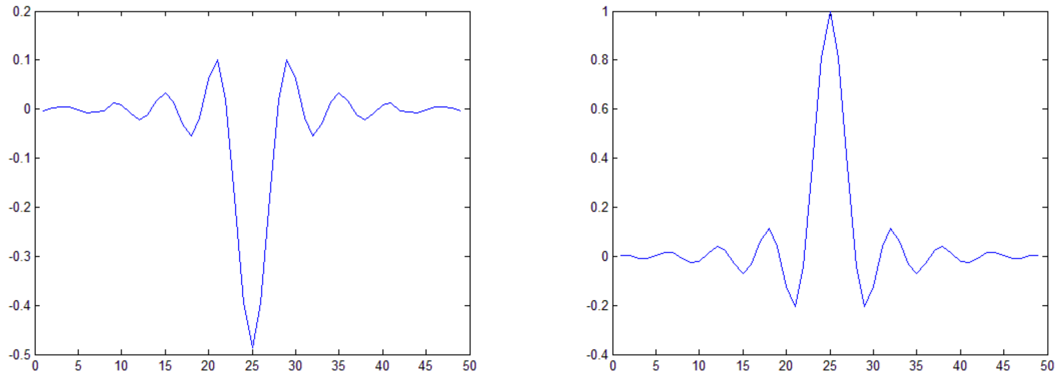


Fig. 1.3 Linear phase FIR filter coefficients initializing method

Experiment results show this can improve the convergence of optimizing progress.

$$y = \exp(x) \tag{1.29}$$

In this way, the convergence velocity becomes much faster than regular method.

1.8 Approximation Criteria

Minimax error designs

Minimax error function is defined as equation (1.30). It searches for the maximum peak ripple throughout a discrete frequency domain by subtracting the magnitude response of the ideal FIR filter to designed filter.

$$\text{error} = \max\{|H_d(e^{j\omega})| - |D(\omega)|\} \tag{1.30}$$

Where $D(\omega)$ is the magnitude of the ideal filter, and $H(e^{jw_i})$ is the magnitude of the designed filter. M is the number of frequency interval.

Least-Squared Error Designs.

Here, this error is defined as

$$E_p = \int_X \left[W(\omega)[|H_d(e^{j\omega})| - D(\omega)]\right]^p d\omega \tag{1.31}$$

Where p is a positive even integer. For FIR filter, due to there are direct efficient algorithms which can approach the error in the minimax sense, the error design is of little practical except for in the case of $p = 2$, E_2 is found very effective. In this case, the error to be minimized is

$$E_2 = \int_X \left[W(\omega)\left[\left|H_d\left(e^{j\omega}\right)\right| - D(\omega)\right]\right]^2 d\omega \tag{1.32}$$

1.9 Fitness function

As mentioned before, the error norm is used as the fitness function in our experiment. Therefore, the problem is transferred to optimize the coefficients by minimizing the error norms, i.e. error function.

Passband/stopband ripples

Passband/ stopband ripples are often expressed in dB,

Passband ripple = $20log_{10}(1 + \delta_p)$ dB,

Minimum stopband attenuation=-$20log_{10}(\delta_p)$ Db

1.10 Conclusion

This section introduces the basic knowledge on digital FIR filter design and concepts associated. The four types of symmetric digital filters are most commonly applied in this field, and some relevant methods are also explored. Coefficients optimization is the main and high-effective aspect by which specifications can be satisfied. Also, the cost function is introduced here. Minimax design technique is easy to complete, and the results obtained is superior as compared to other methods.

There are many ways to design Digital FIR filters, such as windowing functions, frequency sampling method, and non-linear optimization algorithm method. Evolving algorithms have been proved to be meaning in the application of digital FIR filter design. And the following part will introduce the two of them PSO and DE to design some filters and give analysis.

Chapter 2 Linear Phase Digital FIR Filter Design Using PSO Algorithm

2.1. Particle Swarm Optimization

Currently, there are many approaches to FIR filter design, such as window functions, frequency sampling method and best uniform approximation. These methods are all based on the approximation to frequency characteristics of ideal FIR filters. Researchers recently have proposed Simulated Annealing Approach (SA) [6] and Genetic Algorithms (GA) [7]. As we know, GA is difficult to realize due to the complexity of coding and SA costs too much computation. PSO is a random search algorithm and has been successfully applied to many real-world problems.

PSO algorithm is one of the population-based stochastic algorithms for searching optimal solutions which is based on social-psychological principles. Unlike DE algorithm, it does not use selection strategy, typically, all population individuals survive from the beginning and update until the end. During that process, the quality of individuals can be improved by their interactions over time [8].

PSO was first presented in 1995 by James Kennedy and Russell C. Eberhart. PSO simulates a kind of social optimization in the animal swarm like ant group or fish schooling. Such a problem is given, and respondent methods to evaluate a proposed solution to it are in the form of a fitness function. Like the behavior in the group such as communication structure or social network are also defined, and neighbors can interact with the other particles. Then a population of individuals defined randomly at the problem solutions is initialized. These are candidate solutions exit in the objective space. These candidate solutions will be updated during the iterative process. The fitness of the candidate solutions and their positions are also remembered, and they are all available to their neighbors. It is convenient to choose a better candidate in each population.

Usually, a vector of D dimensions is randomly initialized in a specified space, is conceptualized as a point in a high-dimensional Cartesian coordinate system [8]. These points which are called particles can move around in the space. They often operate simultaneously and have the ability to convergence towards one direction within the searching space like animals behaviors, so that they are referred to as particle swarm. Comparing to Jehad I. Ababneh [23], we designed some more complex and higher dimensional linear phase digital FIR filters in this chapter.

PSO has proved itself very useful in multivariable problems where all variables are numerical values. It links to Artificial Intelligence generally and with bird flocks, fish schools and swarm theory particularly are effective. On the other hand, PSO also combines with Evolutionary Computation which is evolved from Genetic Algorithm. Normally, particles move in a Euclidean problem space. Each particle follows its neighbors for searching a better way to find the food. [9]

PSO learns the knowledge itself from the iteration progress and solves the optimization problem. All the particles have fitness values which are calculated by the fitness function for optimization and have velocities to direct their step towards the objective solutions.

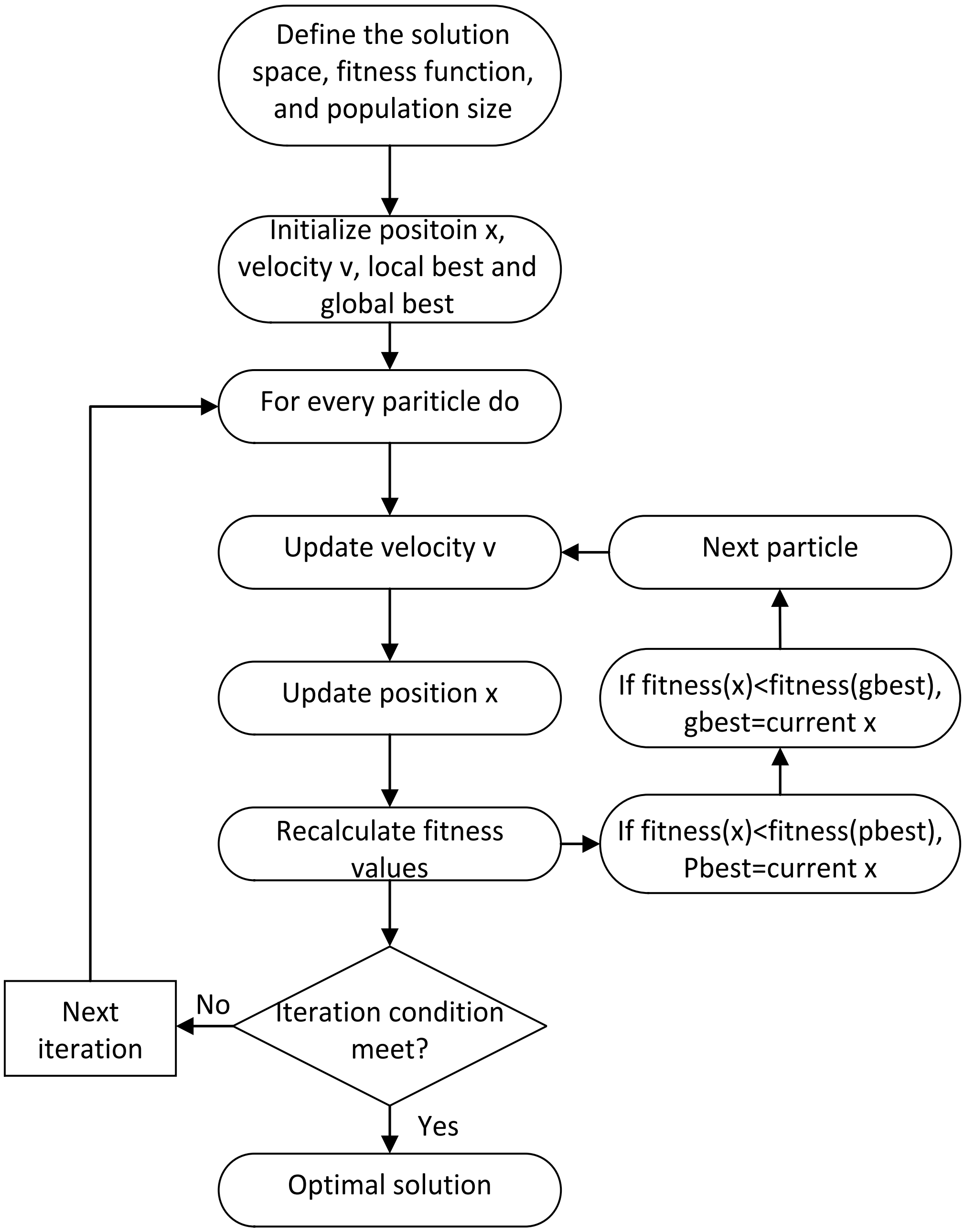


Fig. 2.1 Flow chart of the PSO algorithm

The initial populations as a vector are generated as some random values in the specified domain and then change for improvement by iterations. In each iteration, the particle updates itself by following two ‘best’ values. The first one is the optimal solution that the particle with the best fitness in this generation. This solution is called the local best. The other optimal solution is called the global optima which is the particle achieved the best fitness of all the population until now.

Normally, there are two characteristics which are used to represent the particle in the swarm:

1. The current position

2. The current velocity

The populations updated themselves according to the change of their position and velocity. These two vectors can be calculated by the following equations:

$$v_i^{k+1} = wv_i^k + c_1r_1(pbest_i - s_i^k) + c_2r_2(gbest - s_i^k) \tag{2.1}$$

$$s_i^{k+1} = s_i^k + v_i^{k+1} \tag{2.2}$$

v_i^k is the current velocity of i at iteration k, s_i^k is the current position of i at iteration k. c_1 and c_2 are two positive constants and r_1 and r_2 are random numbers between [0, 1].

The velocity value is limited to the range of [-$Vmax$, $Vmax$].

The velocities of particles on each generation are constrained to a maximum. If a random acceleration makes the velocity on that iteration go beyond the maximum velocity, then the velocity on that generation is set to $Vmax$. On the other side, if the reduction is less than -$Vmax$, then it will be set to $-Vmax$. In this way, an unacceptable information lost will be forbidden during the mutation process.

The algorithm evaluates the particles vector in each generation in the fitness value $f\ (x)$ and compares the result to the best vector which is called $pbest_i$ attained until now. If the current result is better than $pbest_i$ so far, the vector p_i is updated with the current position s_i, and the previous best function result $pbest_i$ is updated with the current result.

The particles are always updated along with iterations within the region centered on the centroid of the current best $pbest_i$ and $gbest_i$. While it also gives an opportunity for the individuals to explore new space around the best values. After a lot of calculations, the particles will converge to optimal solutions, and the optimal fitness is obtained as well.

To implement PSO to solve the filter design, the filter coefficients $\{h(0), h(1), \dots, h(N-1)\}$ represent the position of the particle.

The pseudo code of the procedure for designing a linear phase digital FIR filter is as follows:

$For\ each\ group\ of\ coefficients$
DO {$Initialize\ population$
}

Do{
$For\ each\ population${
$Calculate\ fitness\ value\ using\ equation$ (1.11), (1.12), (1.13), (1.14) and (1.30)

$If\ the\ fitness\ value\ is\ better\ than\ the\ best\ fitness\ value\ (pBest)\ in\ history$
$set\ current\ value\ as\ the\ new\ pBest$
}

$Choose\ the\ indiviual\ with\ the\ best\ fitness\ value\ of\ all\ the\ particles\ as\ the\ gBest$
$For\ each\ individual\{$
$Calculate\ individual\ velocity\ according\ to\ equation\ (a)$
$Update\ individual\ position\ according\ to\ equation\ (b)$
}
}While maximum iterations or minimum error criteria is not attained, repeat.

After finishing the optimizing process, an individual which has the best fitness value should be obtained in the form of $\{h(0), h(1), \ldots, h(N-1)\}$

2.2 Fitness function

In PSO, the value of fitness function can judge the particle's position whether it is 'better.' As mention above, we use the minimax error as the fitness function of FIR digital filter, that is

$$fitness = E = \max\{H(\omega) - D(\omega)\} \tag{2.3}$$

Where $H(\omega)$ is the frequency response of designed filter, and $D\,(\omega)$ represents the frequency response of the ideal lowpass and highpass filters.

$$H(e^{jw}) = \begin{cases} 1, & 0 \le w \le w_c \\ 0, & w_c \le w \le \pi \end{cases} \tag{2.4}$$

$$H(e^{jw}) = \begin{cases} 0, & 0 \le w \le w_c \\ 1, & w_c \le w \le \pi \end{cases} \tag{2.5}$$

And for bandpass filter, the frequency response is

$$H(e^{jw}) = \begin{cases} 1, & w_{c1} \le w \le w_{c2} \\ 0, & elsewhere \end{cases} \tag{2.6}$$

And for bandstop filter, the frequency response is

$$H(e^{jw}) = \begin{cases} 0, & w_{c1} \le w \le w_{c2} \\ 1, & elsewhere \end{cases} \tag{2.7}$$

The search scope of the coefficient of the filter is set as $[-1, 1]$. The number of frequency interval is $N = 201$, with population size $P = 200$. In each running of PSO, the maximum iteration varies from 200, 300, 500, and 1000 for order 12, 24, 36 and 48, respectively.

2.3. Experiments and results

This section presents the simulations performed for the design of all four types, i.e., FIR LP, HP, BP and BS filters. Each filter order (N) is taken as 12, 24, 36, and 48 respectively. For the LP filter, passband (normalized) edge frequency w_p=0.45; stopband (normalized) edge frequency w_s=0.55; For the HP filter, stopband (normalized) edge frequency w_p=0.45; passband (normalized) edge frequency w_p=0.55; For the BP filter, lower stop band (normalized) edge frequency w_{s1}=0.25; lower passband (normalized) edge frequency w_{p1}=0.35; upper passband (normalized) edge frequency w_{p2}=0.6; upper stopband (normalized) edge frequency w_{s2}=0.7. For the BS filter, lower passband (normalized) edge frequency w_{p1}=0.3; lower stop band (normalized) edge frequency w_{s1}=0.4; upper stopband (normalized) edge frequency w_{s2}=0.55; upper passband (normalized) edge frequency w_{p2}= 0.65.

The original population of each of filter examples are generated using the way which is introduced in the first chapter. We randomly select 7, 13, 19, 25 figures from [0, 1] and sort these figures as one side of the initial population of the odd-symmetric filters of 12th-order, 24th-order, 36th-order, and 48th-order, respectively. The opposite side of the filter coefficients is copied from these coefficients which are already produced using the above way. The size of the population is determined by the repeats of the above process.

The experiment works on Intel (R) Core i5-3210M, 2.50GHz, 8G RAM, Windows 10 and Matlab R2014a.

The example are divided to two part: one is simulated for type1, and the other part is for type2. Lowpass, highpass, bandpass, and bandstop filters are designed. Their frequency response is as follows:

Maximum velocity=1, $w_{max} = 1$, $w_{min} = 0.5$, $c_1 = 2$, $c_2 = 2$, $r_f = 1$, $r_g = 1$

2.3.1 Type1 lowpass linear phase digital FIR filter

Type1 lowpass FIR filter with order 12

Table 2.1 Coefficients of 12th-order type1 LP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(13)	0.005534994356476	h(4) = h(10)	-0.210273228457961
h(2) = h(12)	0.204903829885383	h(5) = h(9)	0.002978951624785
h(3) = h(11)	-0.002359268425988	h(6) = h(8)	0.633865285274643
h(7)	0.979474555157335		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with PSS algorithm is showed in Table 2.2, respectively.

Table 2.2 Design result of 12th-order LP-FIR filter by PSO

Time/s	Stopband error	Passband error

144.075300	0.151279546816109	0.150398090335351

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.2.

The figure on the right shows the magnitude response in dB, the local details of the passband, its group delay, and phase features.

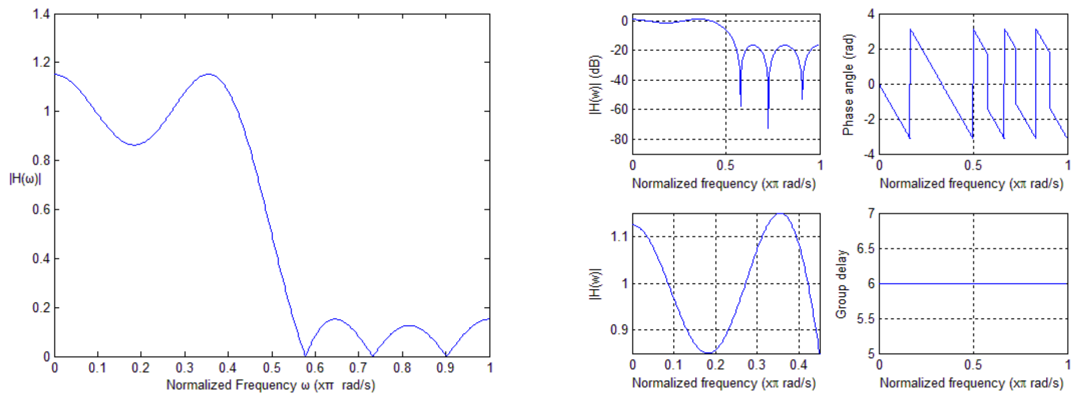


Fig. 2.2 Magnitude, ripple errors of 12th-order LP-FIR filter by PSO

Type1 lowpass FIR filter with order 24

Table 2.3 Coefficients of 24th-order type1 LP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(25)	-0.00205313558952449	h(8) = h(18)	0.0562946888009923
h(2) = h(24)	-0.030068149381552	h(9) = h(17)	-0.000581504493755191
h(3) = h(23)	-0.00147132225764667	h(10) = h(16)	-0.10145624481207
h(4) = h(22)	0.0260337333074784	h(11) = h(15)	-0.00176296327102027
h(5) = h(21)	0.00206282005990487	h(12) = h(14)	0.31798107800902
h(6) = h(20)	-0.0399928058440625	h(13)	0.498109088452561
h(7) = h(19)	0.00261279638810157		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.4, respectively.

Table 2.4 Design result of 24th-order LP-FIR filter by PSO

Time/s	Stopband error	Passband error
172.832061	0.046248431572409	0.053365558441687

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.3.

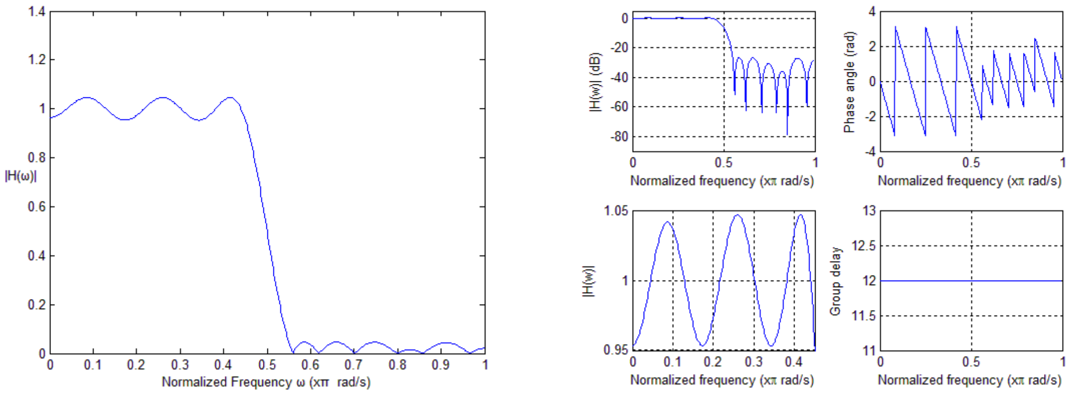


Fig. 2.3 Magnitude, ripple errors of 24th-order LP-FIR filter by PSO

Type1 lowpass order 36

Table 2.5 Coefficients of 36th-order type1 LP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(37)	-0.000848687622753	h(11) = h(27)	0.005178935904278
h(2) = h(36)	-0.014373261726844	h(12) = h(26)	0.070252878147661
h(3) = h(35)	0.003532976138188	h(13) = h(25)	-0.004009188000524
h(4) = h(34)	0.020972225219487	h(14) = h(24)	-0.113059749018623
h(5) = h(33)	0.000261333849933	h(15) = h(23)	0.000922838929777
h(6) = h(32)	-0.024552465492260	h(16) = h(22)	0.196319899422876
h(7) = h(31)	0.003499169897124	h(17) = h(21)	-0.005800980449815
h(8) = h(30)	0.036518218751672	h(18) = h(20)	-0.608830155468197
h(9) = h(29)	-0.004067059332583	h(19)	-0.949698223596581
h(10) = h(28)	-0.052709642218845		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.6, respectively.

Table 2.6 Design result of 36th-order LP-FIR filter by PSO

Time/s	Stopband error	Passband error
406.050082	0.016303333542335	0.016968828284386

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.4.

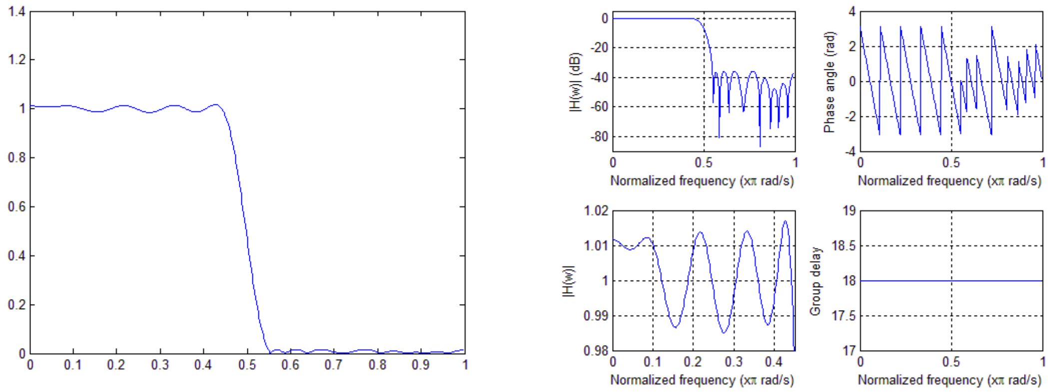


Fig. 2.4 Magnitude, ripple errors of 36th-order LP-FIR filter by PSO

Type1 lowpass FIR filter with order 48

Table 2.7 Coefficients of 48th-order type1 LP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(49)	-0.001047264802628	h(14) = h(36)	-0.034660689509657
h(2) = h(48)	-0.004659896562826	h(15) = h(35)	0.004719266242802
h(3) = h(47)	0.003306790794211	h(16) = h(34)	0.046425707658136
h(4) = h(46)	0.00581997949029	h(17) = h(33)	-0.006608506408141
h(5) = h(45)	-0.002073757257038	h(18) = h(32)	-0.062984244647794
h(6) = h(44)	-0.009249000318886	h(19) = h(31)	0.006155543726317
h(7) = h(43)	0.003448015255106	h(20) = h(30)	0.095378325180374
h(8) = h(42)	0.013281408303607	h(21) = h(29)	-0.007321790729720
h(9) = h(41)	-0.004504414159347	h(22) = h(28)	-0.166505155145664
h(10) = h(40)	-0.016130603174843	h(23) = h(27)	0.007741285042549
h(11) = h(39)	0.002097831633416	h(24) = h(26)	0.510589821652021
h(12) = h(38)	0.022569671716071	h(25)	0.793783575156937
h(13) = h(37)	-0.005590606138566		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.8, respectively.

Table 2.8 Design result of 48th-order LP-FIR filter by PSO

Time/s	Stopband error	Passband error
2617.883373	0.0073759340395515	0.0073776557844607

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.5.

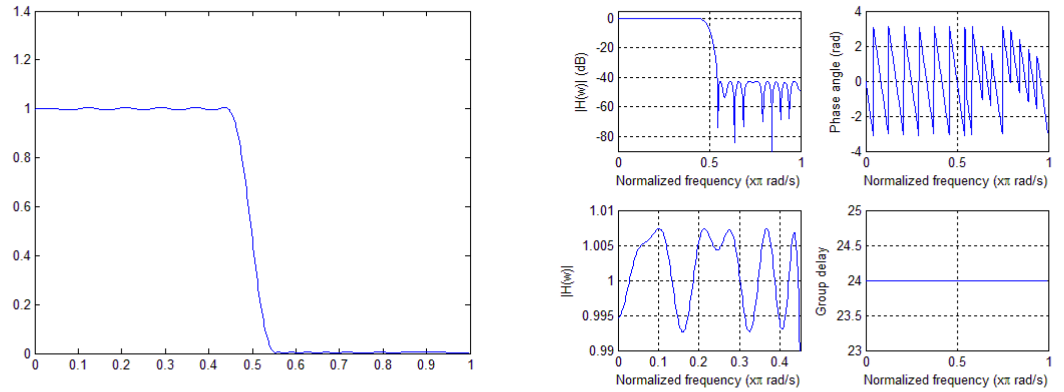


Fig. 2.5 Magnitude, ripple errors of 48th-order LP-FIR filter by PSO

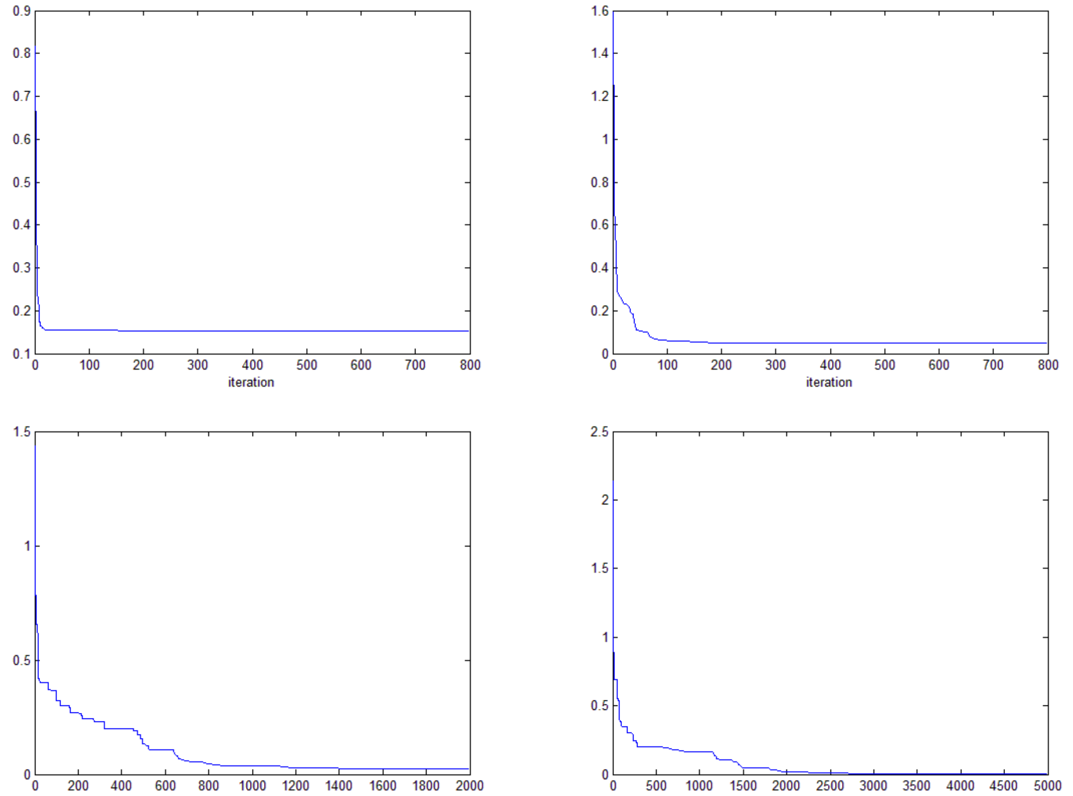


Fig. 2.6 Convergence behaviors of PSO in design LP-FIR with orders of 12th, 24th, 36th, and 48th

2.3.2. Type1 Highpass linear phase digital FIR filter

Type1 Highpass linear phase digital FIR filter with order 12

Table 2. 9 Coefficients of 12th-order type1 HP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(13)	0.000699867644626	h(4) = h(10)	-0.208747738841508
h(2) = h(12)	0.203362039300643	h(5) = h(9)	0.010380376631198

h(3) = h(11)	-0.000955508310356	h(6) = h(8)	0.624370820348988
h(7)	-0.975158774797411		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.10, respectively.

Table 2. 10 Design result of 12th-order HP-FIR filter by PSO

Time/s	Stopband error	Passband error
34.558574	0.149579872844649	0.152893741918110

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.7.

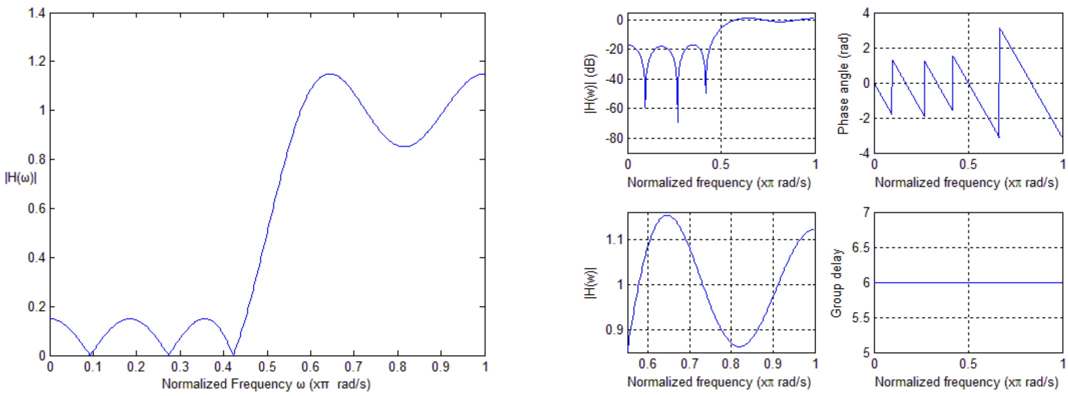


Fig. 2.7 Magnitude, ripple errors of 12th HP-FIR lowpass filter by PSO

Type 1 highpass linear phase digital FIR filter with order 24

Table 2.11 Coefficients of 24th-order type1 HP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(25)	-0.001907719929447	h(8) = h(18)	0.107553738673346
h(2) = h(24)	-0.055324456754076	h(9) = h(17)	-0.002029337109109
h(3) = h(23)	0.002978848699407	h(10) = h(16)	-0.197381510034050
h(4) = h(22)	0.054053250962448	h(11) = h(15)	0.006846698762999
h(5) = h(21)	-0.005152605134815	h(12) = h(14)	0.611147624673500
h(6) = h(20)	-0.072714988789047	h(13)	-0.963200080999353
h(7) = h(19)	0.003124128814972		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.12, respectively

Table 2.12 Design result of 24th-order HP-FIR filter by PSO

Time/s	Stopband error	Passband error

431.147374	0.045813959055968	0.045990750663258

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.8.

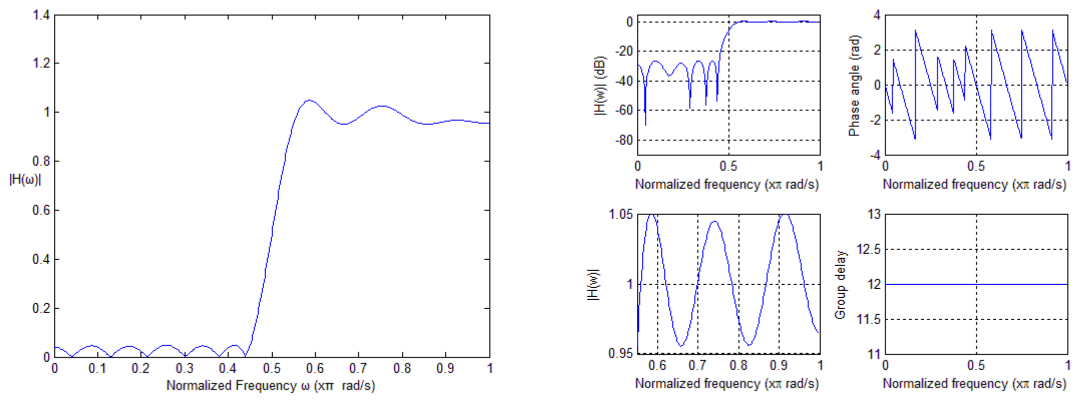


Fig. 2.8 Magnitude, ripple errors of 24th-order HP-FIR filter by PSO

Type 1 highpass linear phase digital FIR filter with order 36

Table 2.13 Coefficients of 36th-order type1 HP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(37)	0.000257940835591	h(11) = h(27)	-0.004614175665226
h(2) = h(36)	0.015851822991697	h(12) = h(26)	-0.076993982614866
h(3) = h(35)	-0.000491936905874	h(13) = h(25)	-0.000225186192316
h(4) = h(34)	-0.020685296757903	h(14) = h(24)	0.116684493870300
h(5) = h(33)	0.003804097208211	h(15) = h(23)	-0.004807348118044
h(6) = h(32)	0.027935193987771	h(16) = h(22)	-0.203959876121314
h(7) = h(31)	0.000218954155286	h(17) = h(21)	0.002424322968095
h(8) = h(30)	-0.036723866724770	h(18) = h(20)	0.631109560654634
h(9) = h(29)	0.003909211381214	h(19)	-0.999636331503199
h(10) = h(28)	0.053262003380551		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.14, respectively.

Table 2.14 Design result of 36th-order HP-FIR filter by PSO

Time/s	Stopband error	Passband error
395.028087	0.015911927701090	0.016021904562858

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.9.

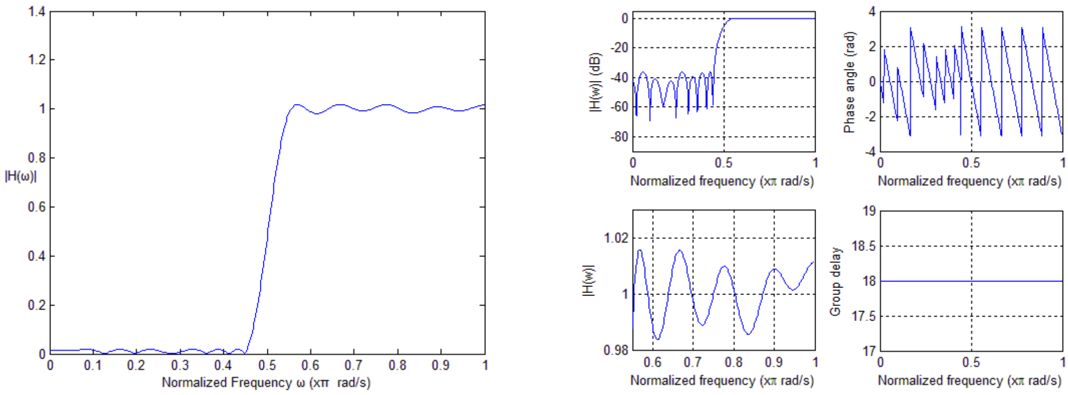


Fig. 2.9 Magnitude, ripple errors of 36th-order HP-FIR filter by PSO

Type 1 highpass linear phase digital FIR filter with order 48

Table 2.15 Coefficients of 48th-order type1 HP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(49)	-0.000603441469539	h(14) = h(36)	-0.040673064911518
h(2) = h(48)	-0.006267240220903	h(15) = h(35)	0.002990087861222
h(3) = h(47)	0.000882865918032	h(16) = h(34)	0.055846976629837
h(4) = h(46)	0.008148050131759	h(17) = h(33)	-0.003693795258497
h(5) = h(45)	-0.001839497968108	h(18) = h(32)	-0.078841933621902
h(6) = h(44)	-0.010526362466545	h(19) = h(31)	0.004421671842661
h(7) = h(43)	0.001265204169247	h(20) = h(30)	0.117417322608263
h(8) = h(42)	0.016199629333704	h(21) = h(29)	-0.003699979773455
h(9) = h(41)	-0.002210154896543	h(22) = h(28)	-0.204014489335479
h(10) = h(40)	-0.021937089750019	h(23) = h(27)	0.004587720204273
h(11) = h(39)	0.002346245169983	h(24) = h(26)	0.625918538674400
h(12) = h(38)	0.029893743977659	h(25)	-0.990087899311546
h(13) = h(37)	-0.003288283096666		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.16, respectively.

Table 2.16 Design result of 48th-order HP-FIR filter by PSO

Time/s	Stopband error	Passband error
4438.526250	0.005342504707092	0.005322669623213

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.10.

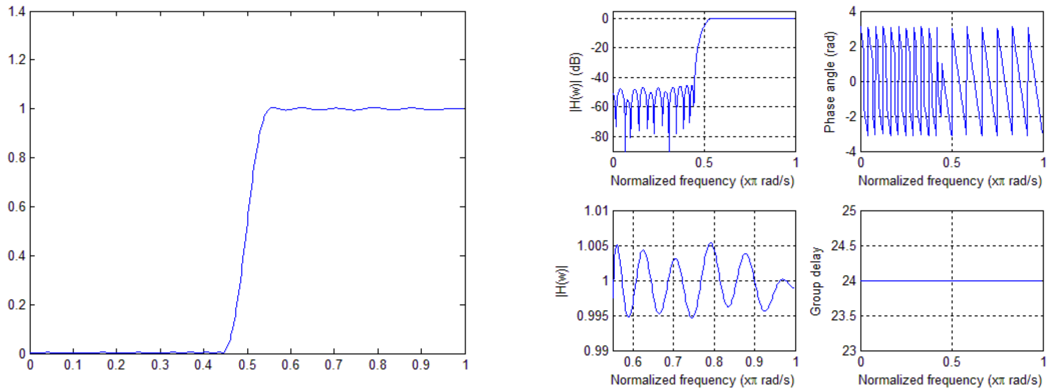


Fig. 2.10 Magnitude, ripple errors of 48th-order HP-FIR filter by PSO

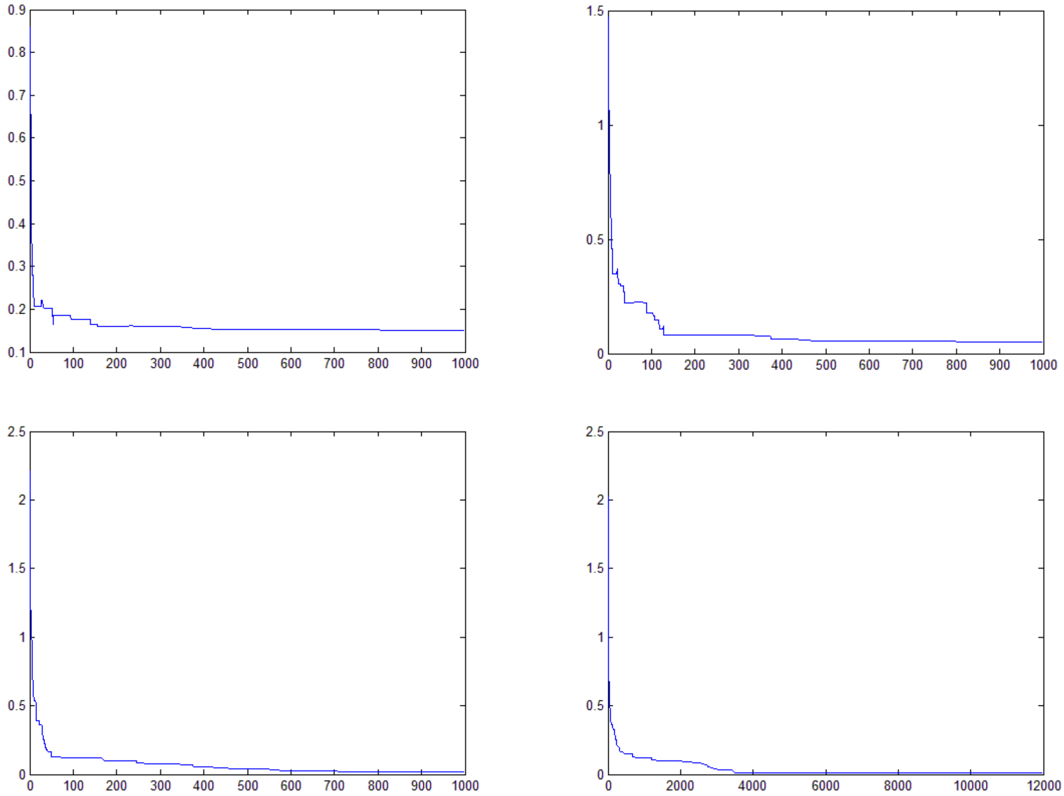


Fig. 2.11 Convergence behaviors of PSO in design HP-FIR with orders of 12th, 24th, 36th, and 48th

2.3.3. Type1 Bandpass linear phase digital FIR filter

Type1 Bandpass linear phase digital FIR filter with order 12

Table 2.17 Coefficients of 12th-order type1 BP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(13)	0.0312026489838628	h(4) = h(10)	-0.0570630050733439
h(2) = h(12)	0.00539934806848757	h(5) = h(9)	-0.277312559770176

h(3) = h(11)	0.149374001755597	h(6) = h(8)	0.0476680091323586
h(7)	0.378345935579198		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.18, respectively.

Table 2.18 Design result of 12th-order BP-FIR filter by PSO

Time/s	Stopband error	Passband error
147.013430	0.186476983648942	0.208422639560898

The magnitude responses of the linear phase digital FIR filters designed using the PSO algorithms for the filter of are given in Fig 2.12.

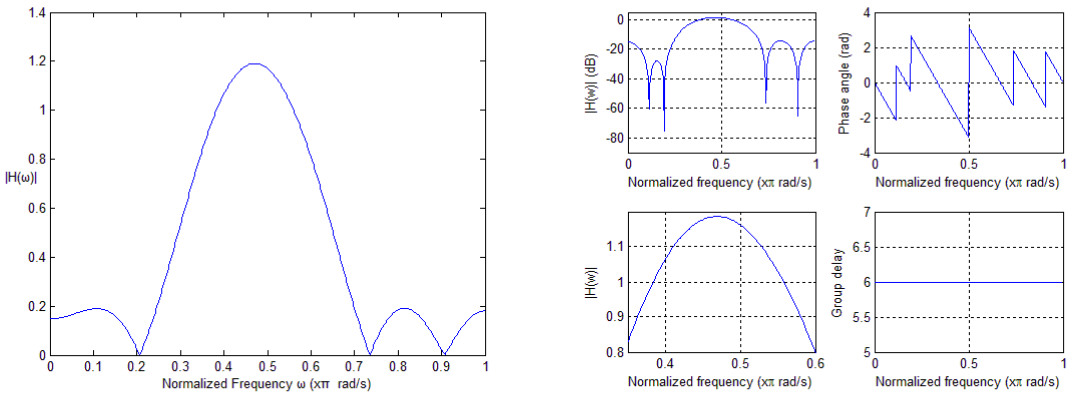


Fig. 2.12 Magnitude, passband and stopband errors of 12th-order BP-FIR lowpass filter

Type 1 bandpass linear phase digital FIR filter with order 24

Table 2.19 Coefficients of 24th-order type1 BP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(25)	-0.022831234638210	h(8) = h(18)	-0.051825209777017
h(2) = h(24)	-0.013210758361870	h(9) = h(17)	-0.328946451064623
h(3) = h(23)	-0.079511782200070	h(10) = h(16)	0.155022612546842
h(4) = h(22)	0.104970109564400	h(11) = h(15)	0.826929849162887
h(5) = h(21)	0.136964193758678	h(12) = h(14)	-0.059169805097528
h(6) = h(20)	-0.089474043268753	h(13)	-0.996832368860821
h(7) = h(19)	-0.033363721334422		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.20, respectively.

Table 2.20 Design result of 24th-order BP-FIR filter by PSO

Time/s	Stopband error	Passband error

215.402234	0.061758283413739	0.063718444449077

The magnitude responses of the linear phase digital FIR filters designed using the PSO, PSO algorithms for the filter of are given in Fig 2.13.

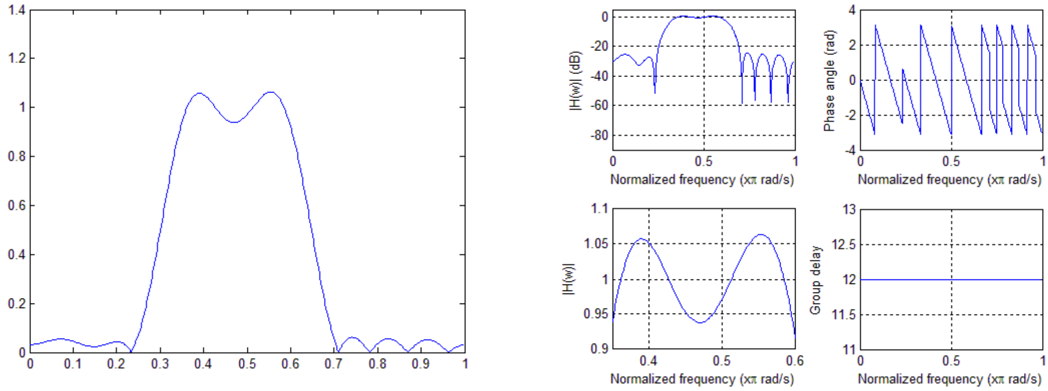


Fig. 2.13 Magnitude, ripple errors of 24th-order BP-FIR filter by PSO

Type 1 bandpass linear phase digital FIR filter with order 36

Table 2.21 Coefficients of 36th-order type1 BP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(37)	-0.000505428202119794	h(11) = h(27)	-0.0464646850831868
h(2) = h(36)	-0.00281619589815389	h(12) = h(26)	0.0322793100346188
h(3) = h(35)	0.000610846544854953	h(13) = h(25)	0.0154005739825305
h(4) = h(34)	-0.0191331357877065	h(14) = h(24)	0.0184805956704867
h(5) = h(33)	-0.00717224237784085	h(15) = h(23)	0.110752571969189
h(6) = h(32)	0.020567172970441	h(16) = h(22)	-0.05511145997133
h(7) = h(31)	0.00743666122569187	h(17) = h(21)	-0.277403463772834
h(8) = h(30)	0.0034297177512027	h(18) = h(20)	0.0292053587486496
h(9) = h(29)	0.0191099865744274	h(19)	0.356195418739576
h(10) = h(28)	-0.0379904435590879		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.22, respectively.

Table 2.22 Design result of 36th-order BP-FIR filter by PSO

Time/s	Stopband error	Passband error
431.318895	0.0228430634111649	0.0370987264169751

The magnitude responses of the linear phase digital FIR filters designed using the PSO, PSO algorithms for the filter of are given in Fig 2.14

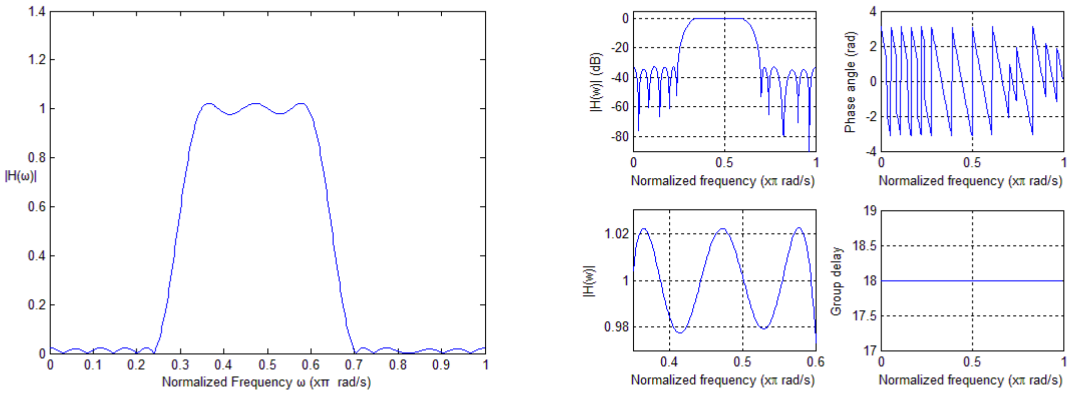


Fig. 2.14 Magnitude, ripple errors of 36th-order BP-FIR filter by PSO

Type 1 bandpass linear phase digital FIR filter with order 48

Table 2.23 Coefficients of 48th-order type1 BP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(49)	-0.001618373921611	h(14) = h(36)	0.022259478588033
h(2) = h(48)	-0.001950510667381	h(15) = h(35)	0.065611290125299
h(3) = h(47)	-0.002558648107667	h(16) = h(34)	-0.106846864275660
h(4) = h(46)	-0.017722790997972	h(17) = h(33)	-0.141188237958171
h(5) = h(45)	0.002079850886173	h(18) = h(32)	0.080731455266420
h(6) = h(44)	0.026636485822660	h(19) = h(31)	0.036321081289342
h(7) = h(43)	0.000792983850795	h(20) = h(30)	0.053380247329694
h(8) = h(42)	0.004335209850867	h(21) = h(29)	0.332597512079348
h(9) = h(41)	0.007503070213622	h(22) = h(28)	-0.145446727933191
h(10) = h(40)	-0.054146624068345	h(23) = h(27)	-0.790748394588430
h(11) = h(39)	-0.027369381119378	h(24) = h(26)	0.079765300554222
h(12) = h(38)	0.056553279439342	h(25)	0.997926201265134
h(13) = h(37)	0.016730690361832		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.24, respectively.

Table 2.24 Design result of 48th-order BP-FIR filter by PSO

Time/s	Stopband error	Passband error
3738.516128	0.006662125873949	0.006669725210366

The magnitude responses of the linear phase digital FIR filters designed using the PSO, PSO algorithms for the filter of are given in Fig 2.15.

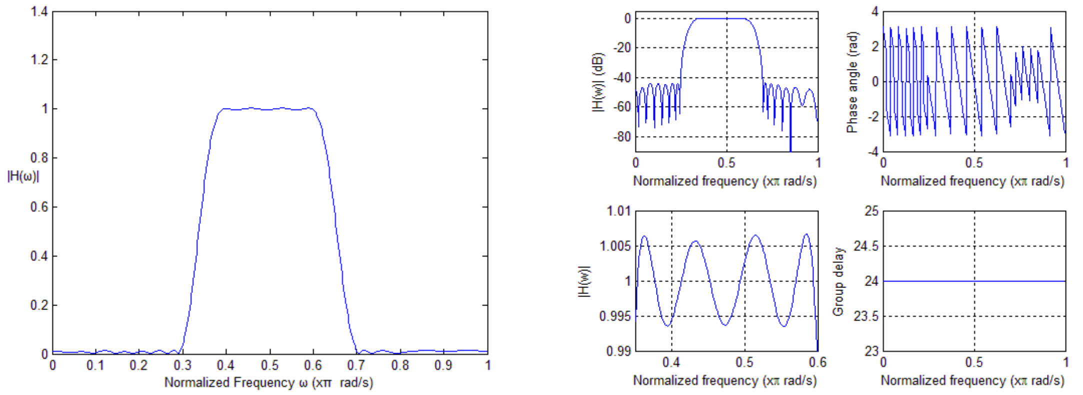


Fig. 2.15 Magnitude, ripple errors of 24th-order BP-FIR filter by PSO

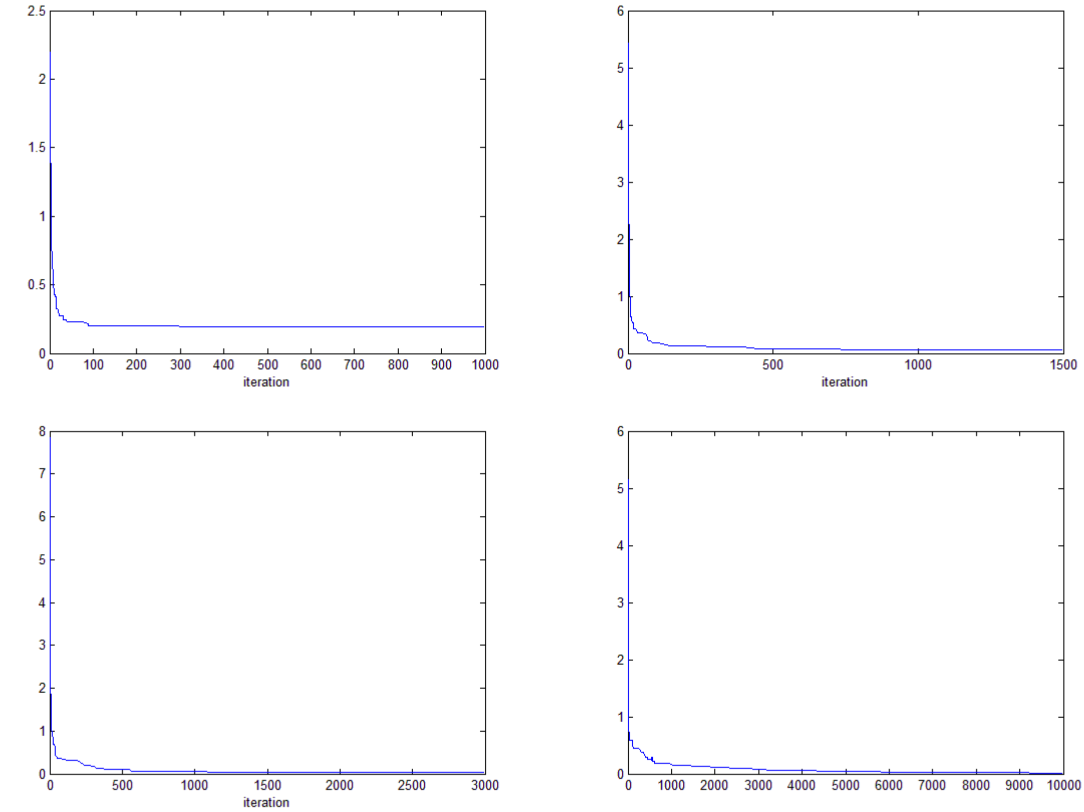


Fig. 2.16 Convergence behaviors of PSO in design BP-FIR with orders of 12th, 24th, 36th, and 48th

2.3.4. Type1 Bandstop linear phase digital FIR filter

Type 1 Bandstop linear phase digital FIR filter with order 12

Table 2.25 Coefficients of 12th-order type1 BS-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(13)	0.0678683524944843	h(4) = h(10)	0.0437816942939301
h(2) = h(12)	-0.0694555402705273	h(5) = h(9)	0.212412742613462

h(3) = h(11)	-0.155838698632166	h(6) = h(8)	-0.0242971361994448
h(7)	0.75125472351821		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.26, respectively.

Table 2.26 Design result of 12th-order BS-FIR filter by PSO

Time/s	Stopband error	Passband error
842.874272	0.107590998821996	0.163215646048442

The magnitude responses of the linear phase digital FIR filters designed using the PSO, PSO algorithms for the filter of are given in Fig 2.17.

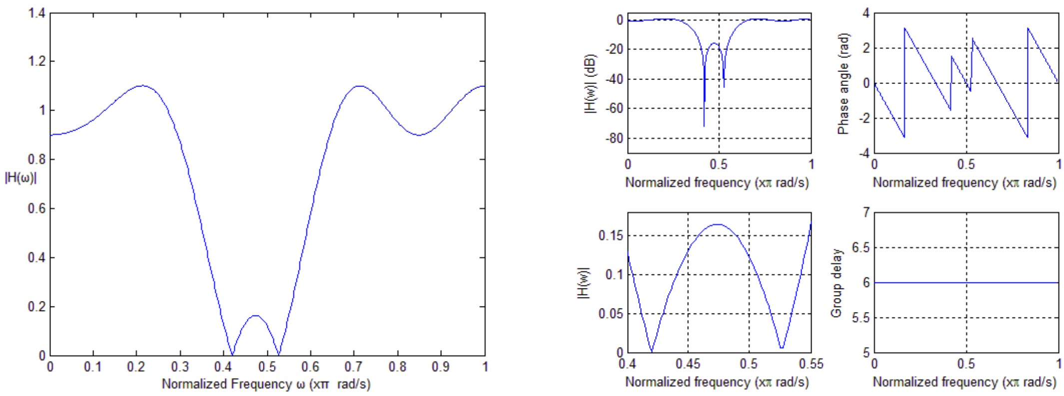


Fig. 2.17 Magnitude, ripple errors of 12th-order BS-FIR filter by PSO

Type 1 bandstop linear phase digital FIR filter with order 24

Table 2.27 Coefficients of 24th-order type1 BS-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(25)	-0.010041643905328	h(8) = h(18)	-0.073108749021779
h(2) = h(24)	0.008752099017537	h(9) = h(17)	0.003874843950236
h(3) = h(23)	-0.046154207062890	h(10) = h(16)	-0.128901644993284
h(4) = h(22)	-0.054018306619055	h(11) = h(15)	0.055099357284101
h(5) = h(21)	0.055800717510152	h(12) = h(14)	0.948686432340606
h(6) = h(20)	0.098450746325749	h(13)	-0.098563580368962
h(7) = h(19)	-0.052583358491048		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.28, respectively.

Table 2.28 Design result of 24th-order BS-FIR filter by PSO

Time/s	Stopband error	Passband error

619.408664	0.060703136028247	0.059849388853956

The magnitude responses of the linear phase digital FIR filters designed using the PSO, PSO algorithms for the filter of are given in Fig 2.18.

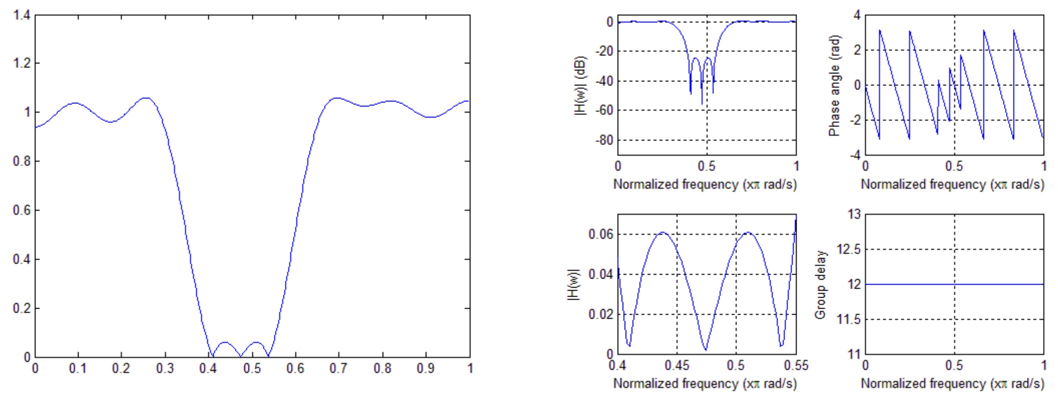


Fig. 2.18 Magnitude, ripple errors of 24th-order BS-FIR filter by PSO

Type 1 bandstop linear phase digital FIR filter with order 36

Table 2.29 Coefficients of 36th-order type1 BS-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(37)	0.017773874586921	h(11) = h(27)	0.071488794603505
h(2) = h(36)	0.001358913804423	h(12) = h(26)	0.099192906544670
h(3) = h(35)	-0.031103805366174	h(13) = h(25)	-0.055278314979214
h(4) = h(34)	-0.011038198368998	h(14) = h(24)	-0.056801273948647
h(5) = h(33)	0.019636917755613	h(15) = h(23)	-0.007673251726469
h(6) = h(32)	0.005196694103630	h(16) = h(22)	-0.141106459476036
h(7) = h(31)	0.006304187674171	h(17) = h(21)	0.070004788584924
h(8) = h(30)	0.025301612786365	h(18) = h(20)	0.991911540626346
h(9) = h(29)	-0.045480389649695	h(19)	-0.093920328972359
h(10) = h(28)	-0.064897559056597		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.30, respectively.

Table 2.30 Design result of 36th-order BS-FIR filter by PSO

Time/s	Stopband error	Passband error
738.516128	0.020408839055092	0.023572479676884

The magnitude responses of the linear phase digital FIR filters designed using the PSO, PSO algorithms for the filter of are given in Fig 2.19.

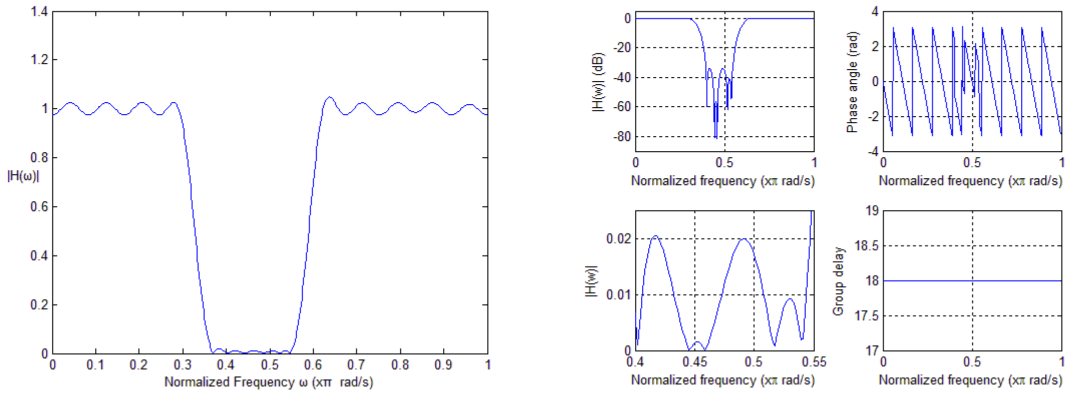


Fig. 2.19 Magnitude, ripple errors of 36th-order BS-FIR filter by PSO

Type 1 bandstop linear phase digital FIR filter with order 48

Table 2.31 Coefficients of 48th-order type1 BS-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1)= h(49)	-0.000169704101268	h(14)= h(36)	0.040923016277587
h(2)= h(48)	-0.002204526890458	h(15)= h(35)	0.028142078783271
h(3)= h(47)	0.001145023600625	h(16)= h(34)	-0.019351053472599
h(4)= h(46)	0.008965859010271	h(17)= h(33)	0.002463417125072
h(5)= h(45)	-0.001773679137716	h(18)= h(32)	-0.024925849525594
h(6)= h(44)	-0.013633565384252	h(19)= h(31)	-0.080217909596378
h(7)= h(43)	-0.000562151020954	h(20)= h(30)	0.061815100463508
h(8)= h(42)	0.005274843402903	h(21)= h(29)	0.189943507899946
h(9)= h(41)	-0.001095910503299	h(22)= h(28)	-0.064809608668370
h(10)= h(40)	0.012215261004327	h(23)= h(27)	-0.290064707351143
h(11)= h(39)	0.008397866441987	h(24)= h(26)	0.028674685088707
h(12)= h(38)	-0.033560659786482	h(25)	-0.999614460016997
h(13)= h(37)	-0.022828339160021		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.32, respectively.

Table 2.32 Design result of 48th-order BS-FIR filter by PSO

Time/s	Stopband error	Passband error
1118.671875	0.007381528454473	0.007252200677178

The magnitude responses of this linear phase digital FIR filters designed using the PSO algorithms are given in Fig 2.20.

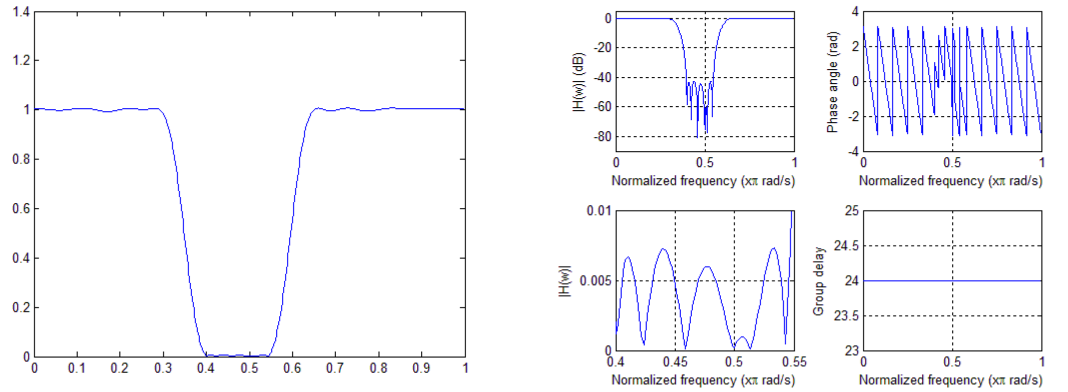


Fig. 2.20 Magnitude, ripple errors of 48th-order LP-FIR filter by PSO

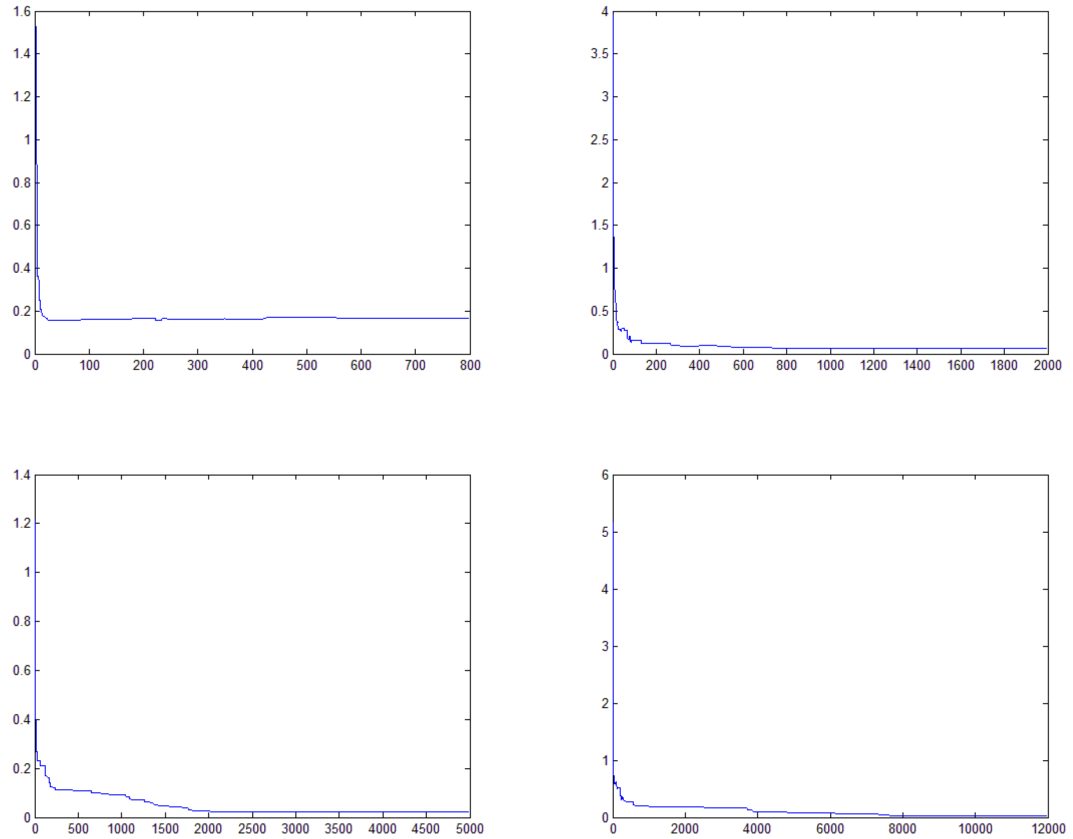


Fig. 2.21 Convergence behaviors of PSO in design BS-FIR with orders of 12th, 24th, 36th, and 48th

Figures above show the amplitude response, magnitude response and convergence behavior of type 1 linear phase digital FIR filters designed by PSO. From the results, we could know that PSO works well in designing lower order filters. While in the application of higher order filters, it is difficult to find the optimum. Actually, in the accrual experiments, we spend too many iterations to get the ideal solutions. All these illustrate that PSO has its limit to apply in the designing of linear phase digital FIR filters.

In the next part, we designed two types of linear phase digital FIR filters which are lowpass and bandpass filters all with order 25 and 49. The parameters are same with before.

2.3.5 Type 2 lowpass linear phase digital FIR filter

This section presents the simulations performed for the design of two type 2 linear phase digital filters of LP and BP. Each filter order (N) is taken as 25 and 49 respectively. For the LP filter, passband (normalized) edge frequency w_p=0.45; stopband (normalized) edge frequency w_s=0.55; For the BP filter, lower stop band (normalized) edge frequency w_{s1}=0.25; lower passband (normalized) edge frequency w_{p1}=0.35; upper passband (normalized) edge frequency w_{p2}=0.6; upper stopband (normalized) edge frequency w_{s2}=0.7.
The original populations of each filter example are generated using the way which is similar to the type1 filter design. We randomly select 13, 25 figures from [0, 1] and sort these figures as one side of the initial population of the even symmetric filter of 25th-order and 49th-order, respectively. During the optimization process, the other side of the coefficients will be copied from the existing part.

Type 2 lowpass linear phase digital FIR filter of order 25

Table 2.33 Coefficients of 25th-order type2 LP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(26)	0.021970049995995	h(8) = h(19)	0.082335959350398
h(2) = h(25)	-0.042272763743529	h(9) = h(18)	0.103866164049824
h(3) = h(24)	-0.032531084894305	h(10) = h(17)	-0.128142059153321
h(4) = h(23)	0.037341010410492	h(11) = h(16)	-0.180423644142344
h(5) = h(22)	0.042498398732192	h(12) = h(15)	0.321345127863449
h(6) = h(21)	-0.057245976097162	h(13) = h(14)	0.963366910397667
h(7) = h(20)	-0.061658345456958		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.34, respectively.

Table 2.34 Design result of 25th-order type 2 LP-FIR filter by PSO

Time/s	Stopband error	Passband error
2337.243058	0.041201848758123	0.041827017886749

The magnitude responses of this kind of linear phase digital FIR filter designed using the PSO algorithms are given in Fig 2.22.

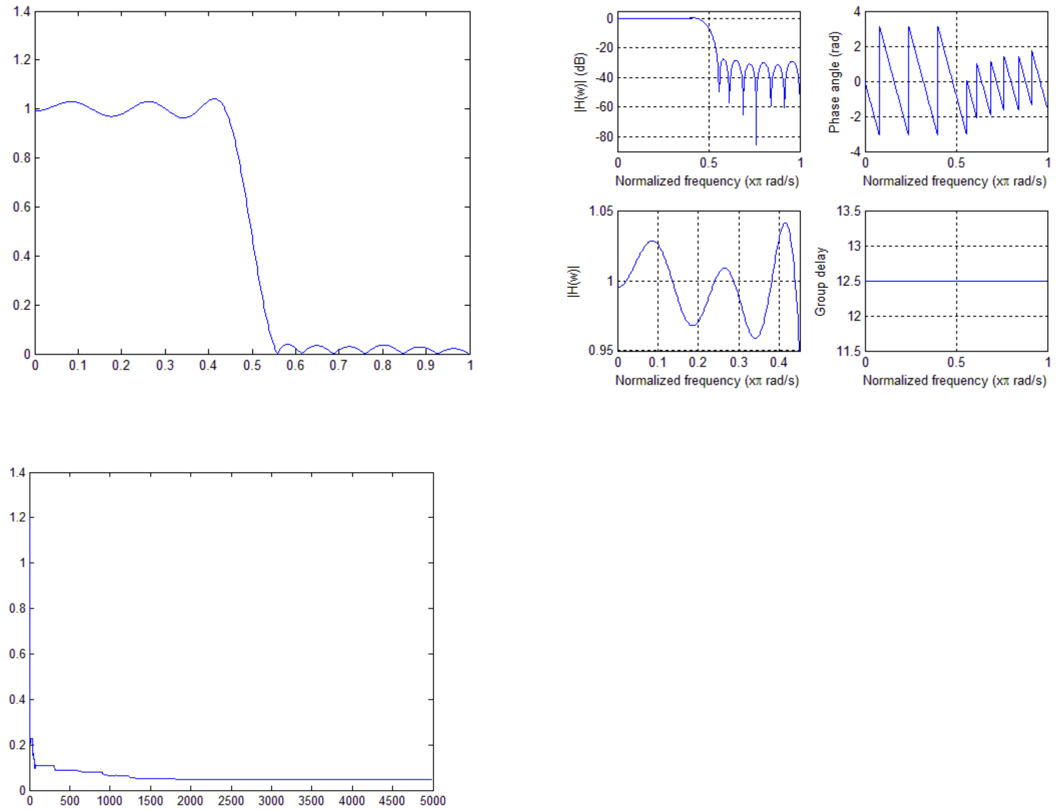


Fig. 2.22 Magnitude, ripple errors of 25th-order type2 LP-FIR filter by PSO

Type 2 lowpass linear phase digital FIR filter of order 49

Table 2.35 Coefficients of 49th-order type2 LP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(50)	0.006133118233681	h(14) = h(37)	-0.028993261817678
h(2) = h(49)	-0.003915934995176	h(15) = h(36)	-0.036547454718272
h(3) = h(48)	-0.006571712845000	h(16) = h(35)	0.040315607310961
h(4) = h(47)	0.003328196544599	h(17) = h(34)	0.048766636234117
h(5) = h(46)	0.006043747413523	h(18) = h(33)	-0.056466921677217
h(6) = h(45)	-0.007475530660646	h(19) = h(32)	-0.070307616619783
h(7) = h(44)	-0.011715760628665	h(20) = h(31)	0.083980461774395
h(8) = h(43)	0.010091974525746	h(21) = h(30)	0.108472348438981
h(9) = h(42)	0.014564232870161	h(22) = h(29)	-0.137012274576683
h(10) = h(41)	-0.016197555729881	h(23) = h(28)	-0.199552203474627
h(11) = h(40)	-0.021298473430499	h(24) = h(27)	0.328834975057105
h(12) = h(39)	0.021332519373483	h(25) = h(26)	0.999999995619789
h(13) = h(38)	0.026643157781947		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.36, respectively.

Table 2.36 Design result of 49th-order type 2 LP-FIR filter by PSO

Time/s	Stopband error	Passband error
3917.663918	0.006352918489874	0.006352918120974

The magnitude responses of this kind of linear phase digital FIR filter designed using the PSO algorithms are given in Fig 2.23

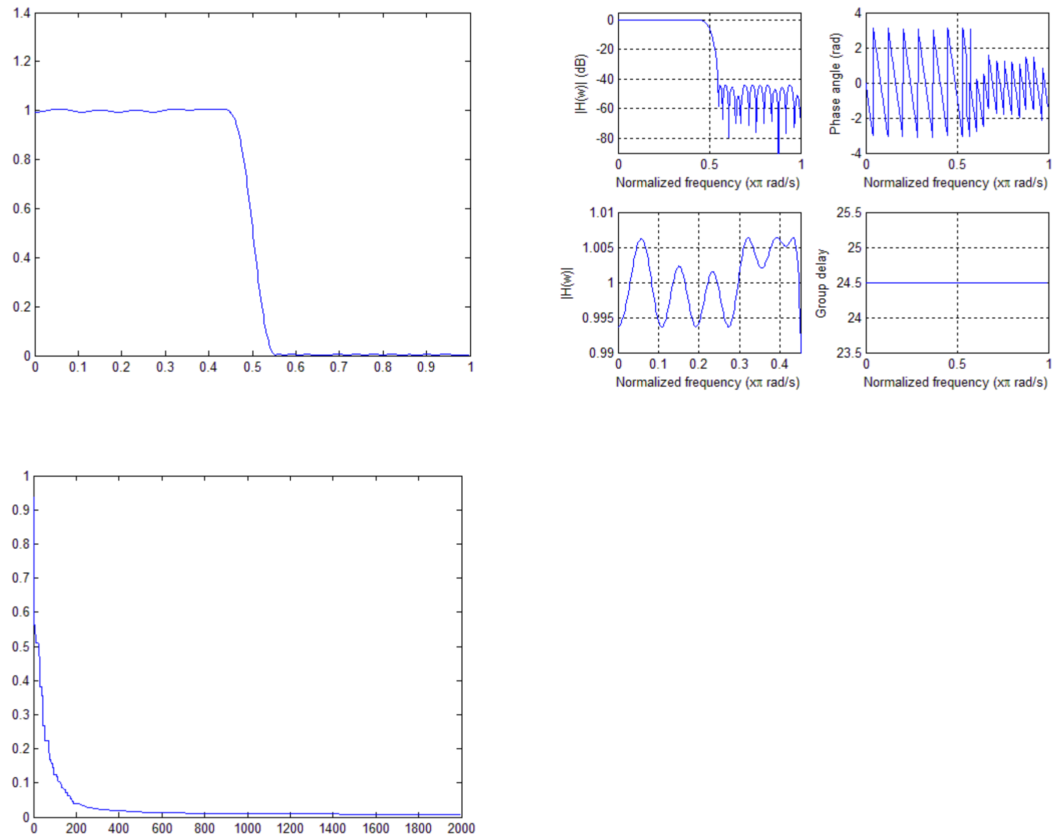


Fig. 2.23 Magnitude, ripple errors of 49th-order type2 LP-FIR filter by PSO

2.3.6 Type 2 bandpass linear phase digital FIR filter

Type 2 bandpass linear phase digital FIR filter of order 25

Table 2.37 Coefficients of 25th-order type2 BP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(26)	0.064277809109691	h(8) = h(19)	-0.013984716456613
h(2) = h(25)	-0.009032475608441	h(9) = h(18)	0.284788418048947
h(3) = h(24)	0.093347612440077	h(10) = h(17)	0.302167142362405
h(4) = h(23)	-0.003111707306286	h(11) = h(16)	-0.789744282114093
h(5) = h(22)	-0.231606894349378	h(12) = h(15)	-0.740982679869700
h(6) = h(21)	-0.037423639547491	h(13) = h(14)	0.986400264232522
h(7) = h(20)	0.135608559903375		

The computational time, passband and stopband ripple error of the linear phase digital FIR

filter design with POS algorithm is showed in Table 2.38, respectively.

Table 2.38 Design result of 25th-order type 2 BP-FIR filter by PSO

Time/s	Stopband error	Passband error
3362.091974	0.048768630452890	0.048847523383255

The magnitude responses of this kind of linear phase digital FIR filter designed using the PSO algorithms are given in Figure 2.24

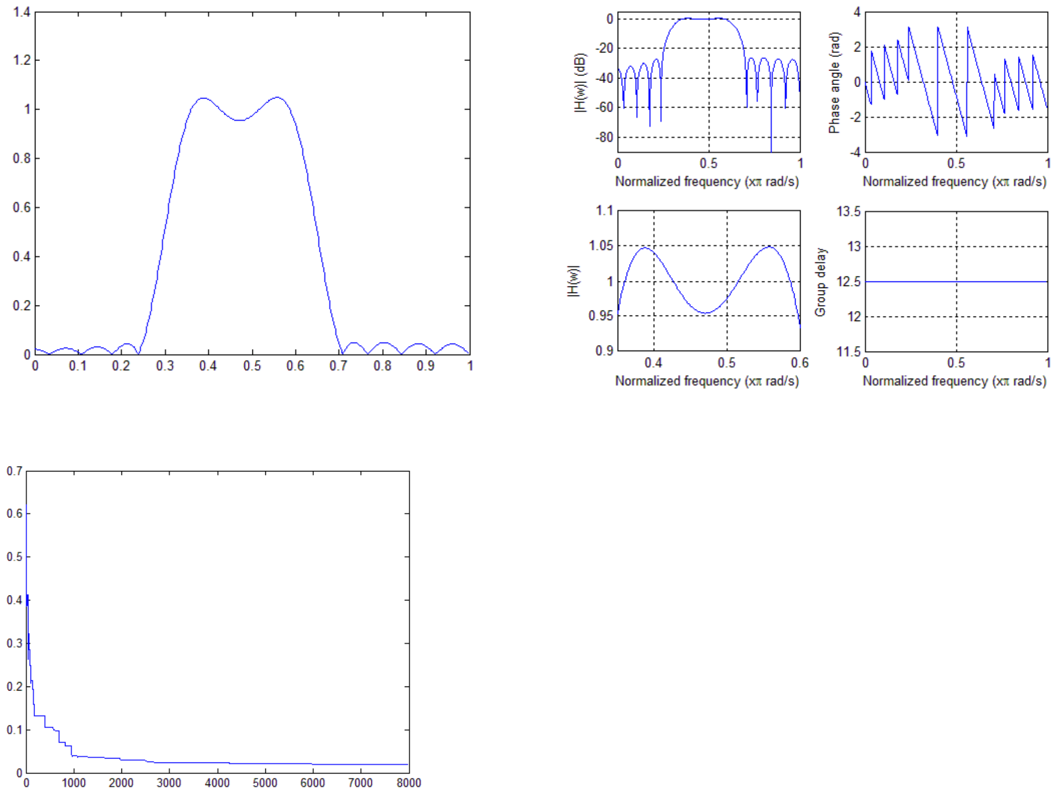


Fig. 2.24 Magnitude, ripple errors of 25th-order type2 BP-FIR filter by PSO

Type 2 bandpass linear phase digital FIR filter of order 49

Table 2.39 Coefficients of 49th-order type2 BP-FIR filter by PSO

h(n)	coefficients	h(n)	coefficients
h(1) = h(50)	0.002599894980289	h(14) = h(37)	-0.002287724405297
h(2) = h(49)	-0.006812698384992	h(15) = h(36)	0.079383761190836
h(3) = h(48)	-0.000433581046170	h(16) = h(35)	-0.000618259132574
h(4) = h(47)	-0.013716402812096	h(17) = h(34)	-0.238148623867656
h(5) = h(46)	-0.018271813196385	h(18) = h(33)	-0.040482977001151
h(6) = h(45)	0.028851092443477	h(19) = h(32)	0.141975039958735
h(7) = h(44)	0.023829775222506	h(20) = h(31)	-0.013421656659549
h(8) = h(43)	-0.008467114675016	h(21) = h(30)	0.292287685218639

h(9) = h(42)	0.015798397948258	h(22) = h(29)	0.307972940193189
h(10) = h(41)	-0.027163291325662	h(23) = h(28)	-0.804789027283914
h(11) = h(40)	-0.084239102191063	h(24) = h(27)	-0.744729617416242
h(12) = h(39)	0.031173172208782	h(25) = h(26)	0.998030911267727
h(13) = h(38)	0.070060370982379		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with POS algorithm is showed in Table 2.40, respectively.

Table 2.40 Design result of 49th-order type 2 BP-FIR filter by PSO

Time/s	Stopband error	Passband error
3325.421875	0.007445696141175	0.006589674897734

The magnitude responses of this kind of linear phase digital FIR filter designed using the PSO algorithms are given in Fig 2.25

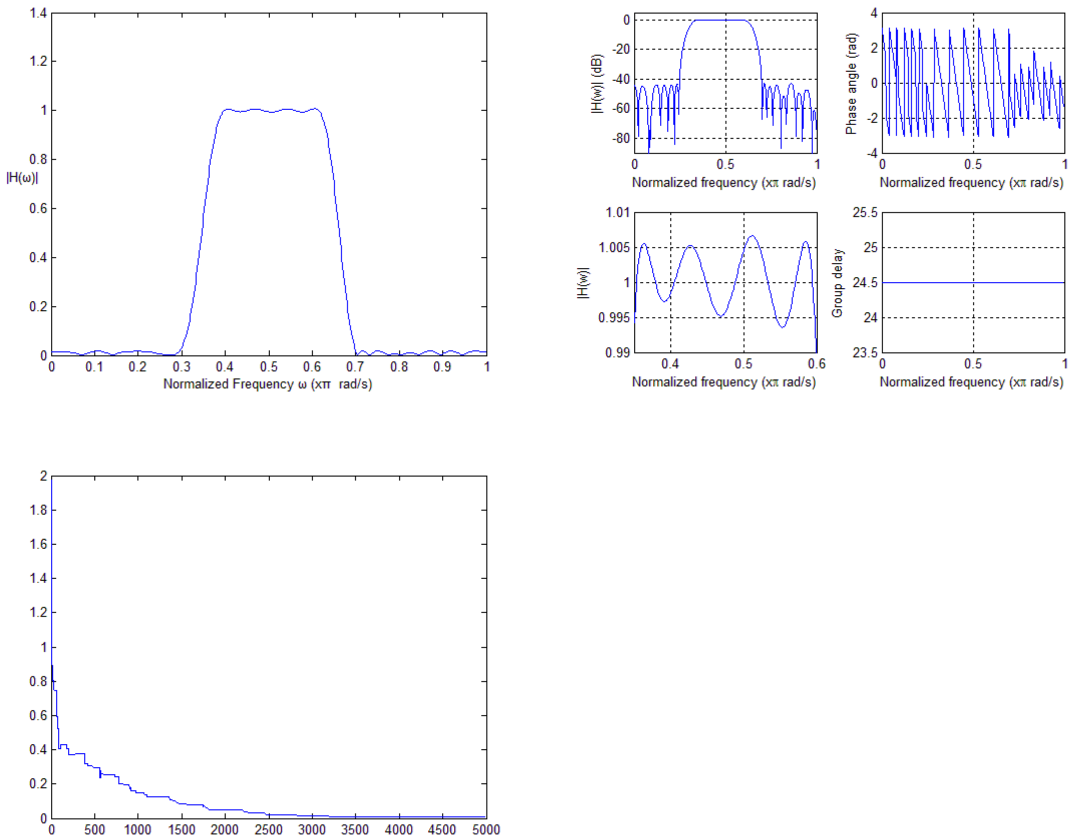


Fig. 2.25 Magnitude, ripple errors of 49th-order type2 BP-FIR filter by PSO

Fig 2.24, Fig 2.25 show the amplitude response, magnitude response and convergence behaviour of type2 FIR filters with 25 and 49 orders designed by PSO. From the results generated for these filters, it is clear that PSO also works very well in searching optimal solutions for such kind of linear phase digital FIR filters.

2.4 Conclusion

PSO was first applied in the field of neural network training. Since then, many more fields of application have been explored, including telecommunications, data mining, computer science, control system, power systems, signal processing, image processing, and many others. Hundreds of publications have been published reporting all these applications of PSO. PSO algorithm has been developed from solving unstrained, single-objective optimization problems to constrained, multi-objective optimization problems, which are dynamically changing landscapes and with multiple solutions.

It is very difficult to realize FIR filter design of high order with normal optimization. PSO performs well in achieving flat passband and stopband frequency response. Experiments show the results of using PSO to find the optimum filter coefficients and the results obtained are approximate to that of Parks-Macllenan algorithm. We have designed order 12, 24, 36, and 48 of FIR filter type1 and 25, 49 of type 2 using PSO. PSO using minimax strategy presents satisfying filter performance and acceptable convergence speed. But it costs too much time to search global optimum in high-dimensional optimization problems. It also can be noticed in the design of digital filters. Although it achieves satisfying performance in designing lower order linear phase digital FIR filters, it costs so many iterations to find an acceptable solution of higher order filters. That is the disadvantage of PSO and the reason for it cannot be applied to the strictly practical engineering programs. Improvement progress in FIR filter design is explored via changing population size and iterations in PSO algorithm. Larger population size in PSO could lead to better filter performance while it will stop till a threshold sometimes. It means PSO is not always stable for high dimensional filters design problem. On the other hand, the convergence speed will become lower because the computing amount increases. In fact, PSO is a mature algorithm in applying in the constrained optimization field. Here we use it as a comparison for generally designing linear phase digital FIR filters. In the next two chapters, we will introduce a more effective and efficient algorithm for designing linear phase linear phase digital FIR filters.

Chapter 3 Linear Phase Digital FIR Filter Design Based on DE Algorithm

3.1 Introduction

In the optimization process of a difficult task, the first choice will usually be a problem-specific heuristics. The common approach to solving an optimization problem is to design an objective function which can replace the problem's objectives while limited to its constraints. In most cases, the objective function describes this kind of optimization problem as a minimization or maximization question [14]. In such cases, the objective function is more accurately defines as a cost function. Many direct search methods use a greedy principle to search the optimal solution. Usually, the greedy method process converges quite fast, but it sometimes takes the risk of being trapped in local optima. So that a method called inherently parallel search techniques appear and get improvement by researchers. It has the capability to overcome this deficiency by running several vectors simultaneously. Such techniques like Differential Evolutionary (DE) algorithm use expert knowledge to achieve a superior performance. DE has the advantages of the four aspects:

1) Ability to handle non-differentiable, nonlinear and multimodal cost functions

2) Parallelizability to cope with computation intensive cost functions.

3) Few control variables to steer the minimization. The parameters are easy to control.

4) Good convergence properties.

3.2 Differential Evolution

The DE algorithm is heuristics algorithm based on the population like genetic algorithms (GAs) using the three same operators; mutation, crossover, and selection to optimize an objective function or cost function over the course of successive generations (Holland, 1975). Of the three operators, mutation operation is used as a search mechanism and selection operates to lead the search toward the direction in the search space. Besides, crossover operation is used to increase the diversity of the perturbed parameter values, which can take children individuals from one parent more frequent than it does from others. If the new vector produces a child with better objective value than before, then the new individual replaces the target vector in the next generation. [12]

The main steps of the DE algorithm is given below:

$$Initialization$$

$$Evaluation$$

$$\mathbf{Repeat}$$

$$mutation$$

$$crossover$$

$$selection$$

$$\textbf{Until}(termination\ criteria\ are\ met)$$

NP is the number of parameter vectors and $\mathbf{x} \in \boldsymbol{R}^d$ is the parameter vectors in the population, where d is the dimension of each individual. Usually, the initial population is generated randomly within the search space or using specific values by the designer. In the next generation, new population will be created from the current population $\{x_i | i = 1, NP\}$, where i indexes the vectors that make up the population. More information will be provided by the mutation and crossover operations and inherited by the selection operation.

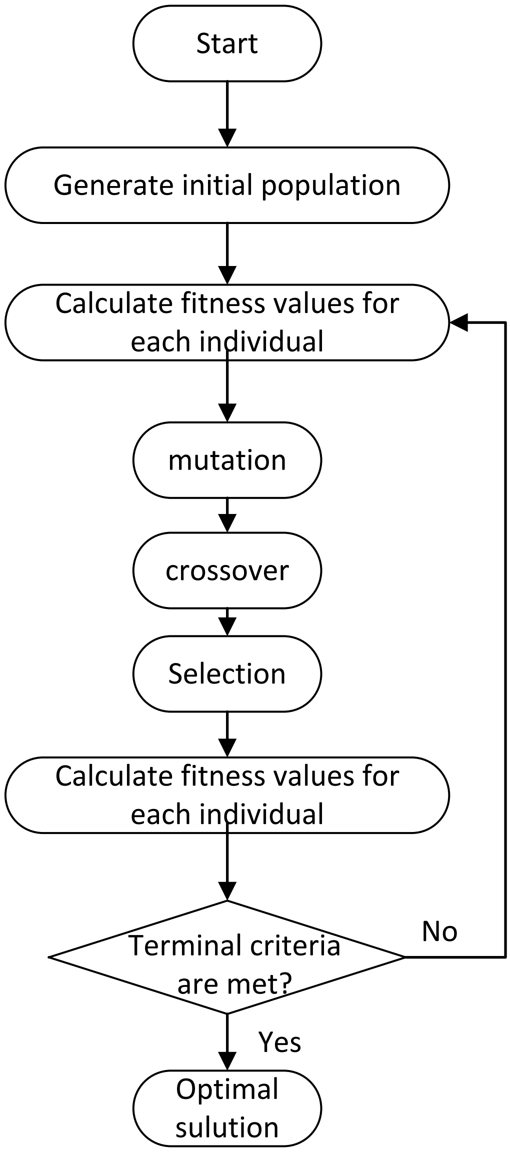


Fig. 3.1 The flow chart of DE algorithm

As for the linear phase digital FIR filter design, the input of the algorithm are a group of coefficients $\{h(0), h(1), \dots, h(N-1)\}$ which initializing using the way in Chapter 1.7. And at the end of the optimization, only one group with optimum fitness value will be obtained. In every generation, each parent parameter vector is targeted for the crossover with a vector to produce a trial vector. As a result, the trial vector is called a child of the two parent vectors, and it will compete with the target vector in the following steps. CR is the probability of crossover which controls the fraction of the parameter values that are succeeded from the mutant. While it is possible some children values can exceed the search space. If it is found to violate the bounds after mutation and crossover operations, it is limited in such a way that the bounds are regulated. Then, the objective function values are calculated with the children vectors.

A. Mutation operation

There are several version of mutation operation. All of them aim to produce new values from the original ones. An initial mutation individual vector y_i is generated by randomly selecting three members of the population, x_{i1}, x_{i2} and x_{i3}. Then y_i is produced as

$$y_i = x_{i1} + F(x_{i2} - x_{i3}) \tag{3.1}$$

Where F is a positive scale factor, effective values for which are typically less than one [21]. All the individuals will experience the mutation until D mutations have been made.

Researchers also proposed some other mutation strategies. Except for (3.1), the most useful strategies are:

"Best/1."

$$y_i = x_{best} + F(x_{i1} - x_{i2}) \tag{3.2}$$

"Current to best/1."

$$y_i = x_i + F(x_{i1} - x_{i2}) + F(x_{best} - x_i) \tag{3.3}$$

"Best/2."

$$y_i = x_{best} + F(x_{i1} - x_{i2}) + F(x_{i3} - x_{i4}) \tag{3.4}$$

"Rand/2."

$$y_i = x_{i1} + F(x_{i2} - x_{i3}) + F(x_{i4} - x_{i5}) \tag{3.5}$$

Where the index $i1$, $i2$, $i3$, $i4$, $i5$ are different with the current population i which represent the random different integers generated within range $[1, NP]$.

B. Crossover Operation

After the mutation operation, a value v_{ji}^{G} is produced which is in the position of x_{ji}^{G}. Crossover operation is then applied to each pair of the target value x_{ji}^{G} and its corresponding mutant value v_{ji}^{G} to decide a new value: u_{ji}^{G}. DE employs the crossover as the basic version:

$$u_{ji}^{G} = \begin{cases} x_{ji}^{G} & if\ rand < CR\ or\ j = jrand \\ v_{ji}^{G} & otherwise \end{cases} \tag{3.6}$$

In (3.6), CR is defined within the range (0.7, 1]. $jrand$ is a randomly chosen integer in the range [1, D].

C. Selection Operation

After the mutation and crossover operations, some new parameters values are added to the current population. Meanwhile, we will remove some of the members which exceed the search space. Then, the cost function values of all the vectors are calculated. After that, a selection operation is performed. The objective function value of each member $f(u_{ji}^{G})$ is compared to that of its corresponding target vector $f(x_{ji}^{G})$ one by one in the current population. If the target vector has better objective function value than the corresponding trial vector, the target vector will maintain and the trial vector will not enter the population of the next generation. Otherwise, the trial vector will replace the target vector in the population for the next generation. The selection operation can be expressed as follows:

$$x_{ji}^{G+1} = \begin{cases} u_{ji}^{G} & if\ f\left(u_{ji}^{G}\right) > f\left(x_{ji}^{G}\right) \\ x_{ji}^{G} & otherwise \end{cases} \tag{3.7}$$

3.3 Control parameters

Many researchers have been working on finding best control parameters for DE algorithm. Suitable control parameters are important for DE algorithm to increase the speed of searching. Recently, some researchers provided new effective ways to optimize the control parameters. Normally, the implemented strategies by Storn in [13] and [14] are:

1) $F \in [0.5, 1]$

2) $CR \in [0.8, 1]$

1) $Np = 10 \cdot D$

Liu and LAmpinen in [14] set control parameters to $F = 0.9$, $CR = 0.9$. The values were chosen based on discussions in [25]. Ali and Törn in [15] empirically obtained an optimal value for CR. They chose to use $CR = 0.5$.

In this chapter, we use the constant control parameter mechanism during the iteration and the control parameter Np keeps no change as well. Both F and CR are applied at the individual level.

The original DE algorithm has three control parameters that need to be adjusted by the user. Different parameters could have influence in different function problems. Is there a set of universal control parameters which we could use for all functions? Janez Brest [26] has done research on this question. When we implement a self-adaptive parameters mechanism in this design, there is no apparent effectiveness in the result. So we still adapt the constant values for this design.

3.4 Experimental results

In this section, we present the performance of DE algorithm. All the four kinds of filters including lowpass, highpass, bandpass and bandstop linear phase digital filters are designed in which the passband and stopband edges are the same with the experiments with PSO, while in the higher order filters design experiments, the amplitude response obtained by DE algorithm are better than that of PSO.

As introduced above, the population is produced using the same way with PSO algorithm. The original populations of each of filter examples are generated using the way which is introduced in the first chapter. We randomly select 7, 13, 19, 25 figures from [0, 1] and sort these figures as one side of the initial population of the odd-symmetric filters of 12th-order, 24th-order, 36th-order, and 48th-order, respectively. The opposite side of the filter coefficients are copied from these coefficients which are already produced using the above way. The size of the population is determined by the repeats of the above process.

This section presents the simulations performed for the design of all four types, i.e., FIR LP, HP, BP and BS filters. Each filter order (N) is taken as 12, 24, 36, and 48 respectively. For the LP filter, passband (normalized) edge frequency w_p=0.45; stopband (normalized) edge frequency w_s=0.55; For the HP filter, stopband (normalized) edge frequency w_p=0.45; passband (normalized) edge frequency w_p=0.55; For the BP filter, lower stop band (normalized) edge frequency w_{s1}=0.25; lower passband (normalized) edge frequency w_{p1}=0.35; upper passband (normalized) edge frequency w_{p2}=0.6; upper stopband (normalized) edge frequency w_{s2}=0.7. For the BS filter, lower passband (normalized) edge frequency w_{p1}=0.3; lower stop band (normalized) edge frequency w_{s1}=0.4; upper stopband (normalized) edge frequency w_{s2}=0.55; upper passband (normalized) edge frequency w_{p2}= 0.65.

The control parameters of differential evolution are assumed to be:

$F = 0.5$; $Cr = 0.9$; $Np = 300$

3.4.1 Type1 Lowpass linear phase digital FIR filter

Low pass linear phase digital FIR of order 12

Table 3.1 Coefficients of 12th-order type1 LP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(13)	0.003356933066366	h(4) = h(10)	-0.203222025163861
h(2) = h(12)	0.216195677190416	h(5) = h(9)	0.003823813584292
h(3) = h(11)	-0.005268830864304	h(6) = h(8)	0.625958064716752
h(7)	0.982860987543980		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.2, respectively.

Table 3.2 Design result of 12th-order type 1 LP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
109.343750	0.147463170638424	0.147463170638891

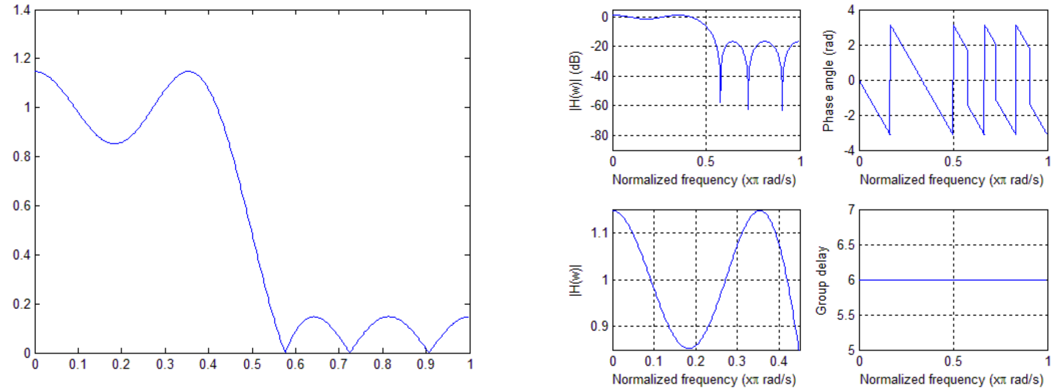


Fig. 3.2 Magnitude, passband and stopband errors of 12th-order LP-FIR filter

Low pass linear phase digital FIR of order 24

Table 3.3 Coefficients of 24th-order type1 LP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(25)	-0.002091888156658	h(8) = h(18)	0.110235029367616
h(2) = h(24)	-0.061278144668594	h(9) = h(17)	-0.003532231879551
h(3) = h(23)	0.003445958039656	h(10) = h(16)	-0.196330390317633
h(4) = h(22)	0.048037343614139	h(11) = h(15)	0.003729433812886
h(5) = h(21)	-0.002887768176983	h(12) = h(14)	0.608500093874971
h(6) = h(20)	-0.071127573483767	h(13)	0.955947677485284
h(7) = h(19)	0.003268881154211		

The computational time, passband and stopband ripple error of the linear phase digital FIR

filter design with DE algorithm is showed in Table 3.4, respectively.

Table 3.4 Design result of 24th-order type 1 LP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
83.781250	0.043536322158584	0.043536385497003

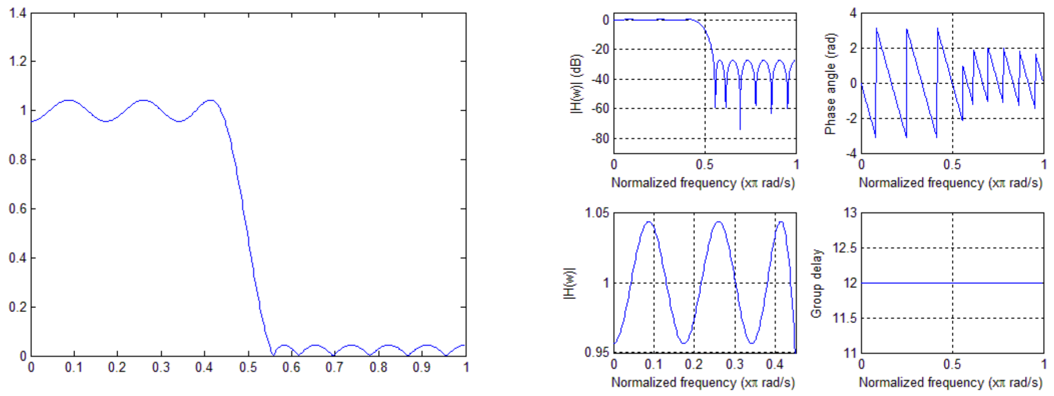


Fig. 3.3 Magnitude, passband and stopband errors of 24th-order LP-FIR filter

Low pass linear phase digital FIR of order 36

Table 3.5 Coefficients of 36th-order type1 LP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(37)	0.001060554707919	h(11) = h(27)	-0.002966600818705
h(2) = h(36)	0.020023154546873	h(12) = h(26)	-0.071737687959856
h(3) = h(35)	-0.001876040941070	h(13) = h(25)	0.003249514161013
h(4) = h(34)	-0.017316159121495	h(14) = h(24)	0.108844666805916
h(5) = h(33)	0.001818266650366	h(15) = h(23)	-0.003476021473003
h(6) = h(32)	0.025049835021479	h(16) = h(22)	-0.191187294115230
h(7) = h(31)	-0.002229542293749	h(17) = h(21)	0.003609069036859
h(8) = h(30)	-0.035446337600527	h(18) = h(20)	0.588674285336614
h(9) = h(29)	0.002605849845932	h(19)	0.924020195958094
h(10) = h(28)	0.049986655362062		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.6, respectively.

Table 3.6 Design result of 36th-order type 1 LP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
140.765625	0.014029908627723	0.014029882185380

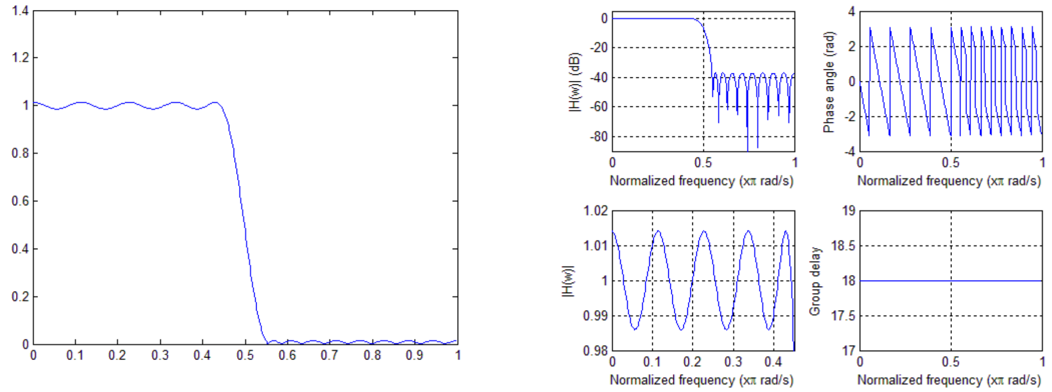


Fig. 3.4 Magnitude, passband and stopband errors of 36th-order LP-FIR filter by DE

Low pass linear phase digital FIR of order 48

Table 3.7 Coefficients of 48th-order type1 LP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1)= h(49)	0.000577169249311	h(14)= h(36)	0.040548440095227
h(2)= h(48)	0.007353919084185	h(15)= h(35)	-0.002954800764217
h(3)= h(47)	-0.001002034314323	h(16)= h(34)	-0.055321144026549
h(4)= h(46)	-0.007506410051835	h(17)= h(33)	0.003235439121256
h(5)= h(45)	0.001136231927526	h(18)= h(32)	0.077846653364401
h(6)= h(44)	0.010967193999411	h(19)= h(31)	-0.003565478784235
h(7)= h(43)	-0.001493993501392	h(20)= h(30)	-0.116101572188713
h(8)= h(42)	-0.015702148386770	h(21)= h(29)	0.003633813679254
h(9)= h(41)	0.001890291664999	h(22)= h(28)	0.202067710350229
h(10)= h(40)	0.021825632319143	h(23)= h(27)	-0.003838131095333
h(11)= h(39)	-0.002221066792568	h(24)= h(26)	-0.619266925497313
h(12)= h(38)	-0.029689938876034	h(25)	-0.971301468465874
h(13)= h(37)	0.002603329897494		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.8, respectively.

Table 3.8 Design result of 48th-order type 1 LP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
837.750000	0.004770844291314	0.004778773524617

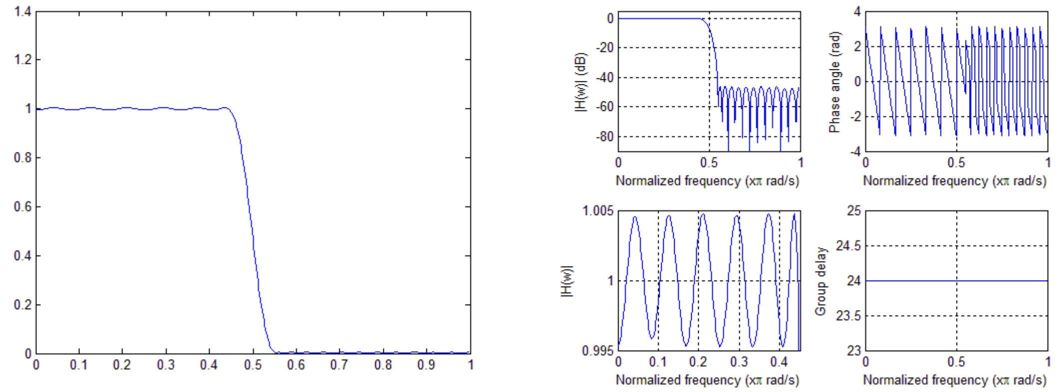


Fig. 3.5 Magnitude, passband and stopband errors of 48th-order LP-FIR filter by DE

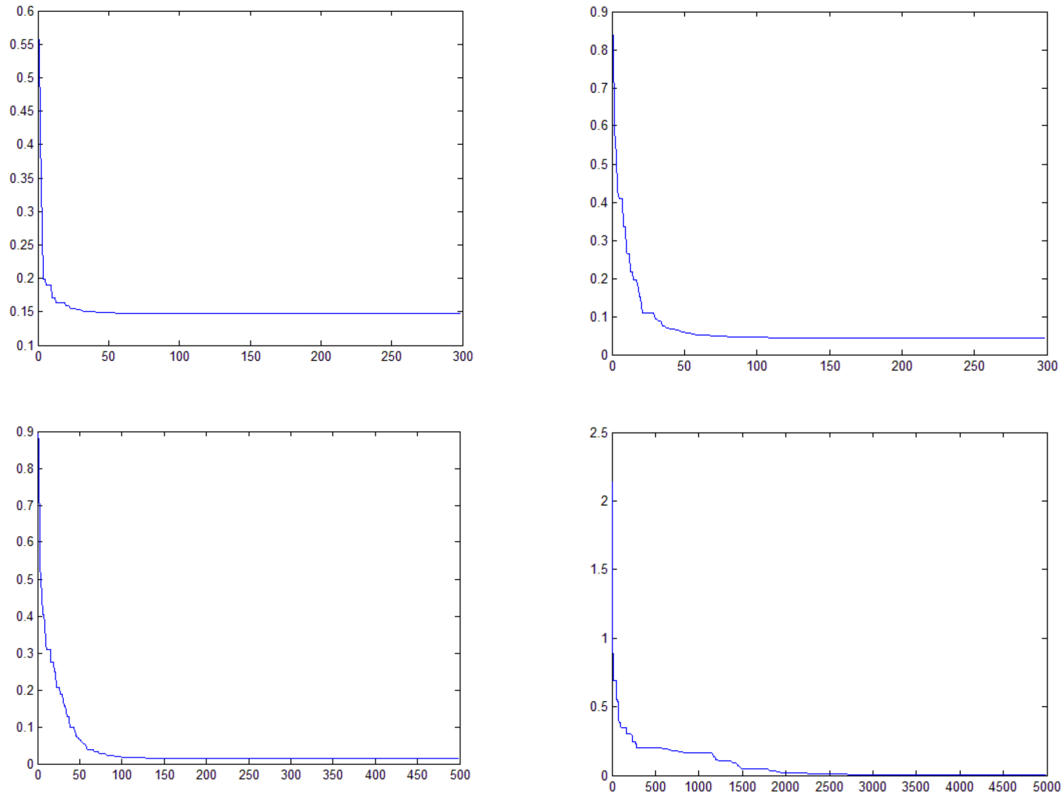


Fig. 3.6 Convergence behaviors of DE in design LP-FIR with orders of 12th, 24th, 36th, and 48th

As shown from Fig 3.6, we can see that the convergence speed performs better in lower order linear phase digital filters. Overall, the convergence speed of DE is obviously better than PSO when designing both the low and high order lowpass linear phase digital FIR filters.

3.4.2 Type1 Highpass linear phase digital FIR filter

High pass linear phase digital FIR of order 12

Table 3.9 Coefficients of 12th-order type1 HP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(13)	-0.003282329548479	h(4) = h(10)	0.198708620104767
h(2) = h(12)	-0.211393618296447	h(5) = h(9)	-0.003738757560840
h(3) = h(11)	0.005151844163382	h(6) = h(8)	-0.612055021377365
h(7)	0.968661795291265		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.10, respectively.

Table 3.10 Design result of 12th-order type 1 HP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
97.109375	0.147463199395232	0.147463252897533

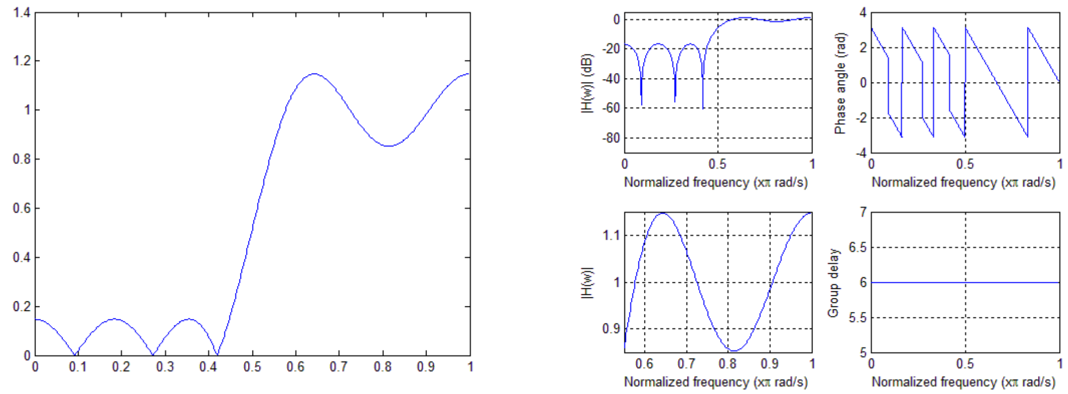


Fig. 3.7 Magnitude, passband and stopband errors of 12th-order HP-FIR filter by DE

High pass linear phase digital FIR of order 24

Table 3.11 Coefficients of 24th-order type1 HP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(25)	0.002279099244290	h(8) = h(18)	-0.112476127589328
h(2) = h(24)	0.059529617333640	h(9) = h(17)	0.002951602672274
h(3) = h(23)	-0.002266548430024	h(10) = h(16)	0.202357818012432
h(4) = h(22)	-0.051443373793945	h(11) = h(15)	-0.005448481993456
h(5) = h(21)	0.004469497147086	h(12) = h(14)	-0.622956367851575
h(6) = h(20)	0.073423566352991	h(13)	0.986200454308054
h(7) = h(19)	-0.002718626431955		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.12, respectively.

Table 3.12 Design result of 24th-order type 1 HP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
141.546875	0.044509573845163	0.044615847837733

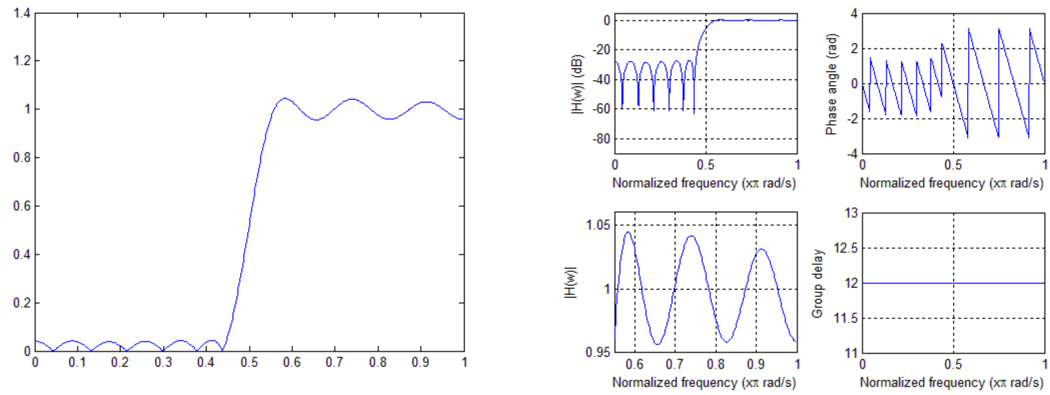


Fig. 3.8 Magnitude, passband and stopband errors of 24th-order HP-FIR filter by DE

High pass linear phase digital FIR of order 36

Table 3.13 Coefficients of 36th-order type1 HP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(37)	0.001192189306089	h(11) = h(27)	-0.003148273739399
h(2) = h(36)	0.019996949749088	h(12) = h(26)	-0.074476657227930
h(3) = h(35)	-0.001523998292533	h(13) = h(25)	0.003261693013235
h(4) = h(34)	-0.018336366691731	h(14) = h(24)	0.112246108021640
h(5) = h(33)	0.001978238219003	h(15) = h(23)	-0.003500286431115
h(6) = h(32)	0.025701691334988	h(16) = h(22)	-0.198115503798683
h(7) = h(31)	-0.002244430364865	h(17) = h(21)	0.004023787542263
h(8) = h(30)	-0.036800022830453	h(18) = h(20)	0.608735631937650
h(9) = h(29)	0.002472337730191	h(19)	-0.963191610880132
h(10) = h(28)	0.051668478858324		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.14, respectively.

Table 3.14 Design result of 36th-order type 1 HP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
145.156250	0.014281709927772	0.014345989865743

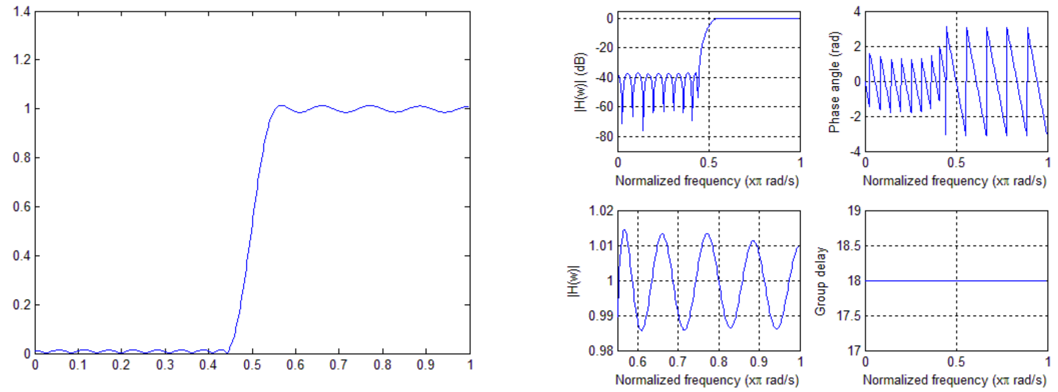


Fig. 3.9 Magnitude, passband and stopband errors of 36th-order HP-FIR filter by DE

High pass linear phase digital FIR of order 48

Table 3.15 Coefficients of 48th-order type1 HP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(49)	-0.000550819122188	h(14) = h(36)	-0.041313273255694
h(2) = h(48)	-0.007681886627144	h(15) = h(35)	0.002997424791901
h(3) = h(47)	0.001047964508741	h(16) = h(34)	0.056513342974403
h(4) = h(46)	0.007558591008403	h(17) = h(33)	-0.003302335232598
h(5) = h(45)	-0.001160351433942	h(18) = h(32)	-0.079343945940823
h(6) = h(44)	-0.011230932630377	h(19) = h(31)	0.003571977222632
h(7) = h(43)	0.001519427887386	h(20) = h(30)	0.118519777367760
h(8) = h(42)	0.016014582785383	h(21) = h(29)	-0.003758752503881
h(9) = h(41)	-0.001882676466337	h(22) = h(28)	-0.206145632378415
h(10) = h(40)	-0.022249170696177	h(23) = h(27)	0.003887524960827
h(11) = h(39)	0.002273872228019	h(24) = h(26)	0.631683723741535
h(12) = h(38)	0.030399852998353	h(25)	-0.998787365048833
h(13) = h(37)	-0.002641936861787		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.16, respectively.

Table 3.16 Design result of 48th-order type 1 HP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
983.703125	0.004729648325497	0.004729204159407

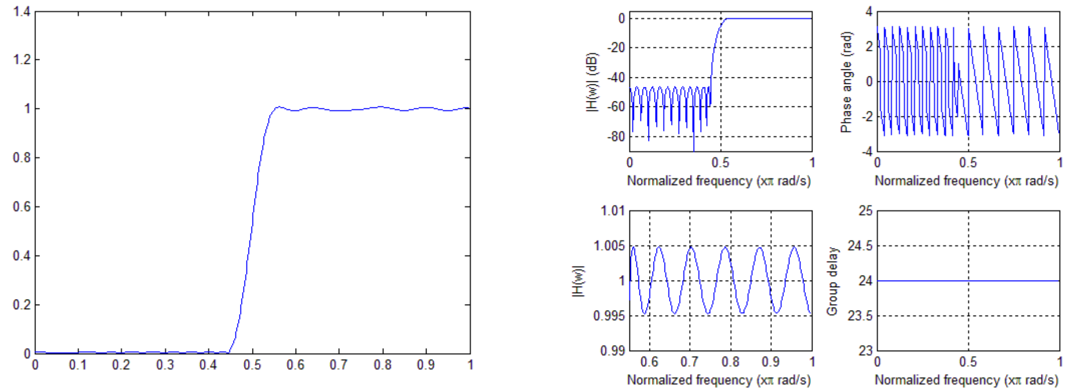


Fig. 3.10 Magnitude, passband and stopband errors of 48th-order HP-FIR filter by DE

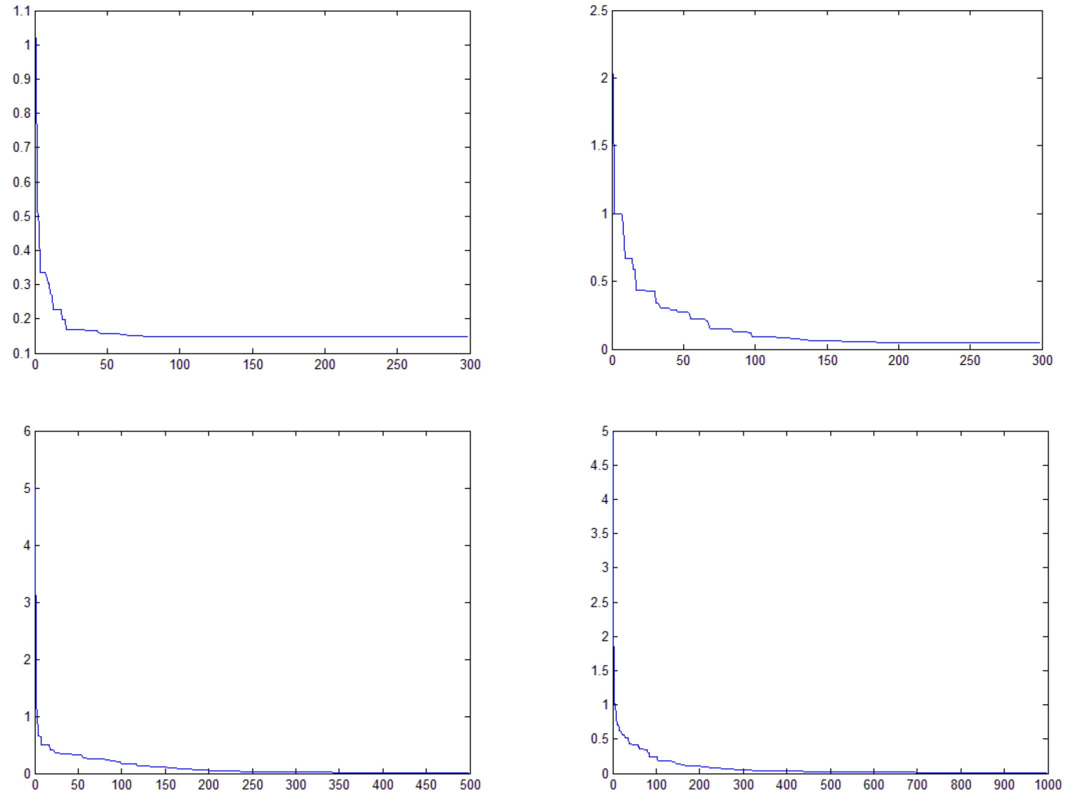


Fig. 3.11 Convergence behaviors of DE in design HP-FIR with orders of 12th, 24th, 36th, and 48th

The convergence behaviors in Fig 3.11 show that the DE algorithm always converges very quickly in lower order filters in this problems. It is because a small number of coefficients are more effective to be optimized, while high dimensional coefficients with an initial need some generations to search the parameters to suitable values. DE algorithm shows a great advantage in this problem.

3.4.3 Type1 Bandpass linear phase digital FIR filter

Bandpass linear phase digital FIR of order 12

Table 3.17 Coefficients of 12th-order type1 BP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(13)	0.054071137797601	h(4) = h(10)	-0.191821738371749
h(2) = h(12)	-0.011886587593314	h(5) = h(9)	-0.723813190544350
h(3) = h(11)	0.310924608494477	h(6) = h(8)	0.014434159531836
h(7)	0.777542571918165		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.18, respectively.

Table 3.18 Design result of 12th-order type 1 BP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
85.250000	0.186688041227229	0.186687974413917

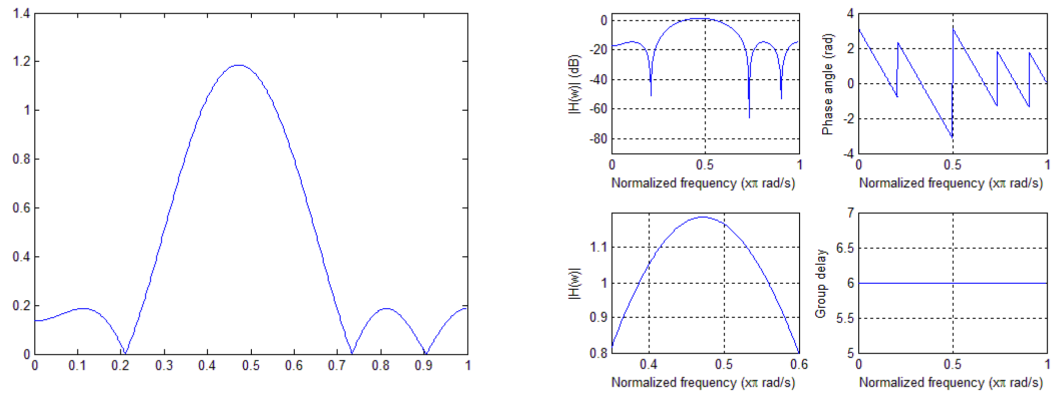


Fig. 3.12 Magnitude, passband and stopband errors of 12th-order BP-FIR filter by DE

Band pass linear phase digital FIR of order 24

Table 3.19 Coefficients of 24th-order type1 BP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(25)	0.014629469184218	h(8) = h(18)	0.048245901038093
h(2) = h(24)	-0.006572874927834	h(9) = h(17)	0.317889014913214
h(3) = h(23)	0.061561258674009	h(10) = h(16)	-0.137062701146521
h(4) = h(22)	-0.121851001404879	h(11) = h(15)	-0.762559025724374
h(5) = h(21)	-0.142004117647309	h(12) = h(14)	0.076898374291133
h(6) = h(20)	0.069166246446750	h(13)	0.973630189552257
h(7) = h(19)	0.032619762185506		

The computational time, passband and stopband ripple error of the linear phase digital FIR

filter design with DE algorithm is showed in Table 3.20, respectively.

Table 3.20 Design result of 24th-order type 1 BP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
84.656250	0.058327396074577	0.058332583584781

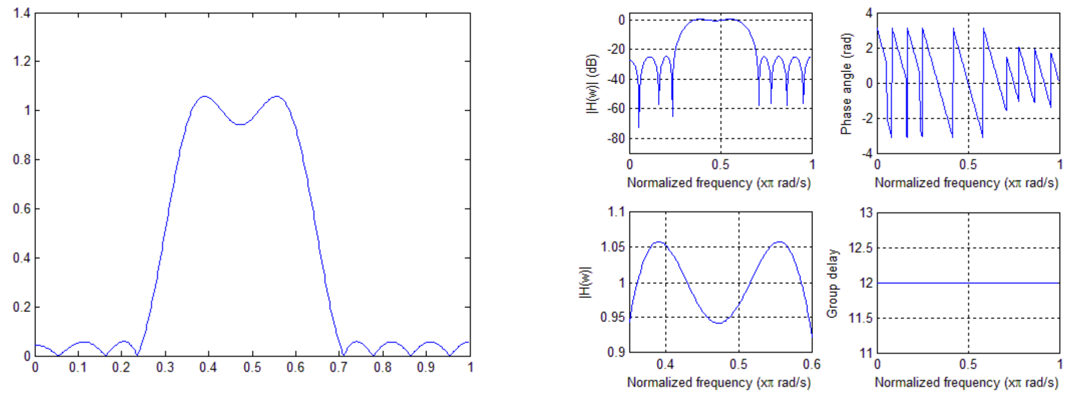


Fig. 3.13 Magnitude, passband and stopband errors of 24th-order BP-FIR filter by DE

Band pass linear phase digital FIR of order 36

Table 3.21Coefficients of 36th-order type1 BP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(37)	0.002136343060196	h(11) = h(27)	-0.131040719249228
h(2) = h(36)	-0.003169467188803	h(12) = h(26)	0.075258496372844
h(3) = h(35)	0.006969444969399	h(13) = h(25)	0.033039514050233
h(4) = h(34)	-0.049642999721532	h(14) = h(24)	0.050942526522823
h(5) = h(33)	-0.025791036167287	h(15) = h(23)	0.316412452529280
h(6) = h(32)	0.050301172772330	h(16) = h(22)	-0.137148398220764
h(7) = h(31)	0.013975876793220	h(17) = h(21)	-0.758114544742329
h(8) = h(30)	0.021625749500595	h(18) = h(20)	0.075714068129521
h(9) = h(29)	0.058968679200407	h(19)	0.958836301056490
h(10) = h(28)	-0.094141461639211		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.22, respectively.

Table 3.22 Design result of 36th-order type 1 BP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
172.343750	0.019501477342182	0.019499285954434

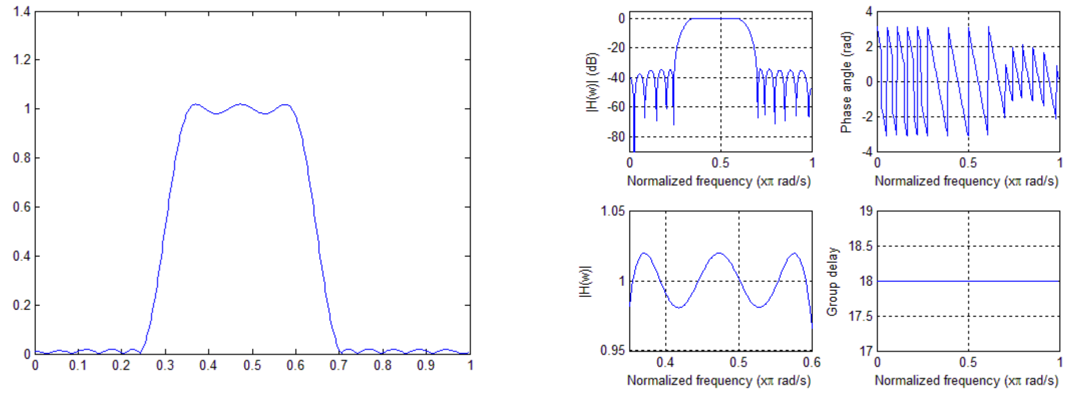


Fig. 3.14 Magnitude, passband and stopband errors of 36th-order BP-FIR filter by DE

Band pass linear phase digital FIR of order 48

Table 3.23 Coefficients of 48th-order type1 BP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(49)	0.001258571785522	h(14) = h(36)	-0.019624443178171
h(2) = h(48)	0.003071500705862	h(15) = h(35)	-0.058289693365850
h(3) = h(47)	0.001239485191595	h(16) = h(34)	0.094987407695314
h(4) = h(46)	0.017422828990808	h(17) = h(33)	0.127121469457387
h(5) = h(45)	-0.002258504229024	h(18) = h(32)	-0.072255157699577
h(6) = h(44)	-0.023492695201383	h(19) = h(31)	-0.033375743454839
h(7) = h(43)	-0.001000879296741	h(20) = h(30)	-0.047040524741474
h(8) = h(42)	-0.003148490333456	h(21) = h(29)	-0.297758273750171
h(9) = h(41)	-0.006383716073370	h(22) = h(28)	0.130045437506002
h(10) = h(40)	0.048214991995511	h(23) = h(27)	0.706863870453515
h(11) = h(39)	0.025507127694839	h(24) = h(26)	-0.070398696608198
h(12) = h(38)	-0.051258773260191	h(25)	-0.893971618648453
h(13) = h(37)	-0.014700451993516		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.24, respectively.

Table 3.24 Design result of 48th-order type 1 BP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
419.500000	0.006254411520480	0.006255115566308

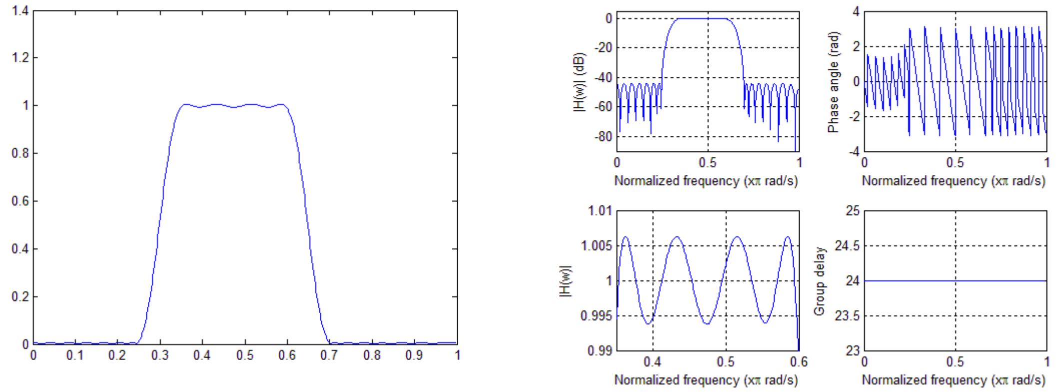


Fig. 3.15 Magnitude, passband and stopband errors of 48th-order BP-FIR filter by DE

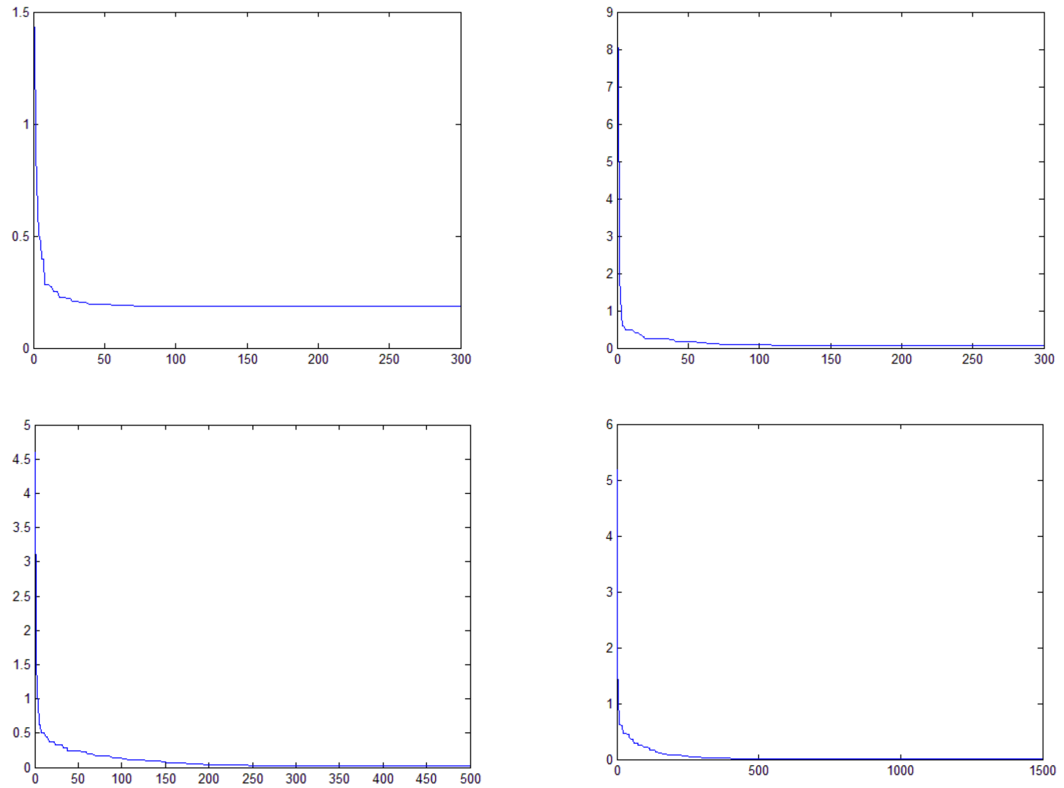


Fig. 3.16 Convergence behaviors of DE in design BP-FIR with orders of 12th, 24th, 36th, and 48th

For the bandpass filters designing problems, PSO has a little difficulty in finding the global optimum which can be seen in a large number of iterations. Meanwhile, DE does very well in this problem and has almost the same computational time on the designing of lowpass and highpass linear phase digital filters.

3.4.4 Type1 Bandstop linear phase digital FIR filter

Band stop linear phase digital FIR of order 12

Table 3.25 Coefficients of 12th-order type1 BS-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(13)	0.057886848104863	h(4) = h(10)	0.043270195700670
h(2) = h(12)	-0.080245785385358	h(5) = h(9)	0.292548727338990
h(3) = h(11)	-0.202302481639780	h(6) = h(8)	-0.049288666324459
h(7)	0.983019402386199		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.26, respectively.

Table 3.26 Design result of 12th-order type 1 BS-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
87.406250	0.134842534921531	0.134842532491527

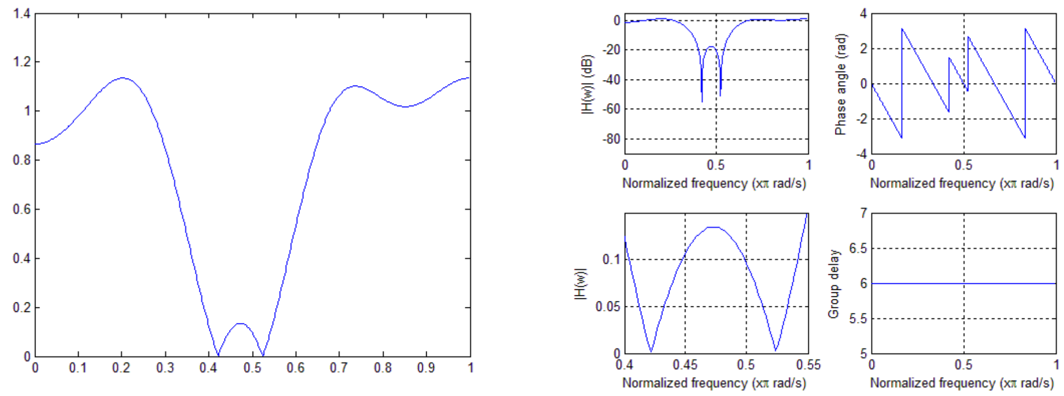


Fig. 3.17 Magnitude, passband and stopband errors of 12th-order BS-FIR filter by DE

Band stop linear phase digital FIR of order 24

Table 3.27 Coefficients of 24th-order type1 BS-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(25)	-0.009248364925037	$h(8) = h(18)$	-0.064907633736886
h(2) = h(24)	-0.003168908781731	$h(9) = h(17)$	0.001111465831486
h(3) = h(23)	-0.033675255884003	$h(10) = h(16)$	-0.120516163475313
h(4) = h(22)	-0.070684958946190	$h(11) = h(15)$	0.060530633563979
h(5) = h(21)	0.071208749996012	$h(12) = h(14)$	0.922898099174558
h(6) = h(20)	0.084618822600080	$h(13)$	-0.086475572273550
h(7) = h(19)	-0.047210723058361		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.28, respectively.

Table 3.28 Design result of 24th-order type 1 BS-FIR filter by DE

Time/s	Passband ripple	Stopband ripple

146.015625	0.054747732174394	0.054747770363342

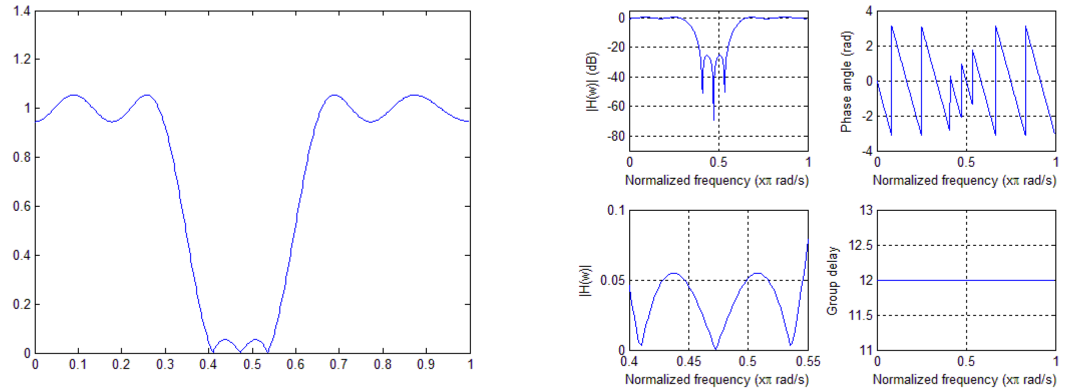


Fig. 3.18 Magnitude, passband and stopband errors of 24th-order BS-FIR filter by DE

Band stop linear phase digital FIR of order 36

Table 3.29 Coefficients of 36th-order type1 BS-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(37)	-0.011780172018308	h(11) = h(27)	-0.070647806177290
h(2) = h(36)	-0.002350433140128	h(12) = h(26)	-0.100737601174706
h(3) = h(35)	0.029410812671224	h(13) = h(25)	0.054800561557716
h(4) = h(34)	0.008762034053417	h(14) = h(24)	0.066263011724357
h(5) = h(33)	-0.025350852111311	h(15) = h(23)	0.000654336598772
h(6) = h(32)	-0.006997839784328	h(16) = h(22)	0.128636054539624
h(7) = h(31)	-0.001772171355354	h(17) = h(21)	-0.063704215932024
h(8) = h(30)	-0.014902483686254	h(18) = h(20)	-0.995092021611334
h(9) = h(29)	0.041780062305684	h(19)	0.094312650809566
h(10) = h(28)	0.061997893290103		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.30, respectively.

Table 3.30 Design result of 36th-order type 1 BS-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
146.843750	0.012053682650155	0.012078800209884

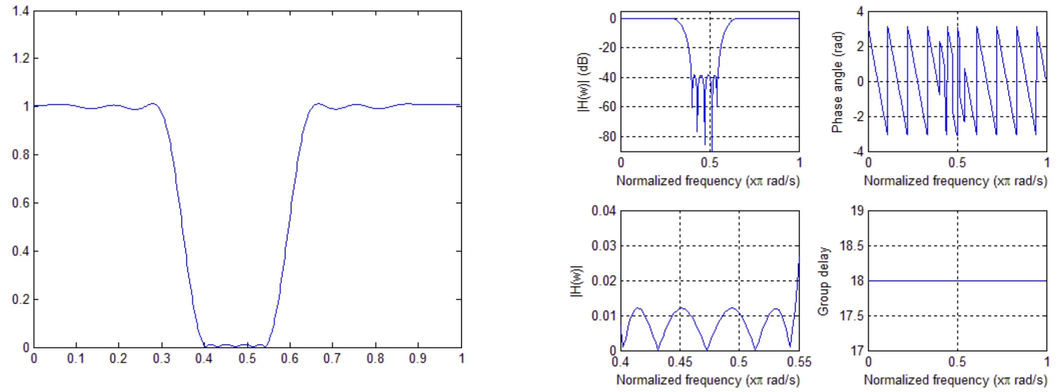


Fig. 3.19 Magnitude, passband and stopband errors of 36th-order BS-FIR filter by DE

Band stop linear phase digital FIR of order 48

Table 3.31 Coefficients of 48th-order type1 BS-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(49)	-0.005449714257318	$h(14) = h(36)$	-0.013641760040768
h(2) = h(48)	0.003322540352091	$h(15) = h(35)$	0.039473309257911
h(3) = h(47)	0.007410904256812	$h(16) = h(34)$	0.059331249814252
h(4) = h(46)	-0.000474214325123	$h(17) = h(33)$	-0.065561634598286
h(5) = h(45)	-0.000614330804565	$h(18) = h(32)$	-0.096996970187136
h(6) = h(44)	-0.000516776861493	$h(19) = h(31)$	0.052472797483714
h(7) = h(43)	-0.014977089406272	$h(20) = h(30)$	0.066914356479095
h(8) = h(42)	-0.003206947756957	$h(21) = h(29)$	-0.001079207700432
h(9) = h(41)	0.029550508464411	$h(22) = h(28)$	0.117799855309430
h(10) = h(40)	0.010168585240612	$h(23) = h(27)$	-0.059231529150074
h(11) = h(39)	-0.028302747928701	$h(24) = h(26)$	-0.916602722962553
h(12) = h(38)	-0.009089664857397	$h(25)$	0.083662805555308
h(13) = h(37)	0.001955250437040		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.32, respectively.

Table 3.32 Design result of 48th-order type 1 BS-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
430.515625	0.005163127949613	0.005174041464594

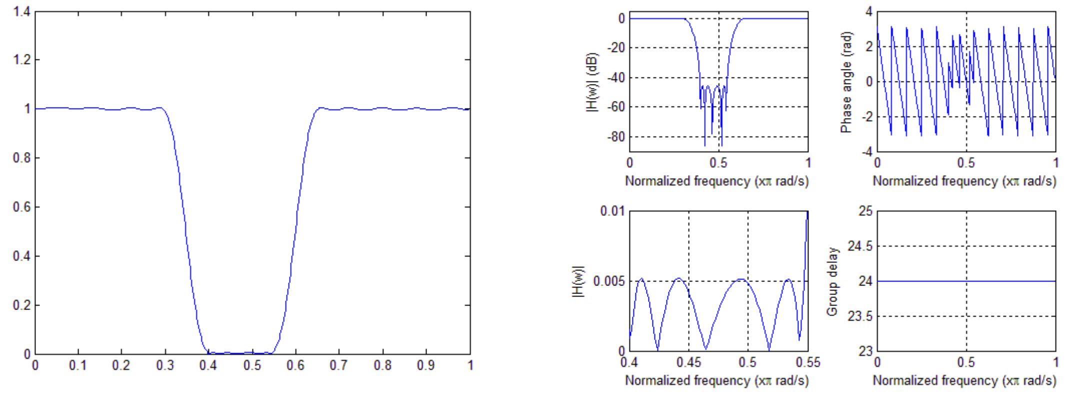


Fig. 3.20 Magnitude, passband and stopband errors of 48th-order BS-FIR filter by DE

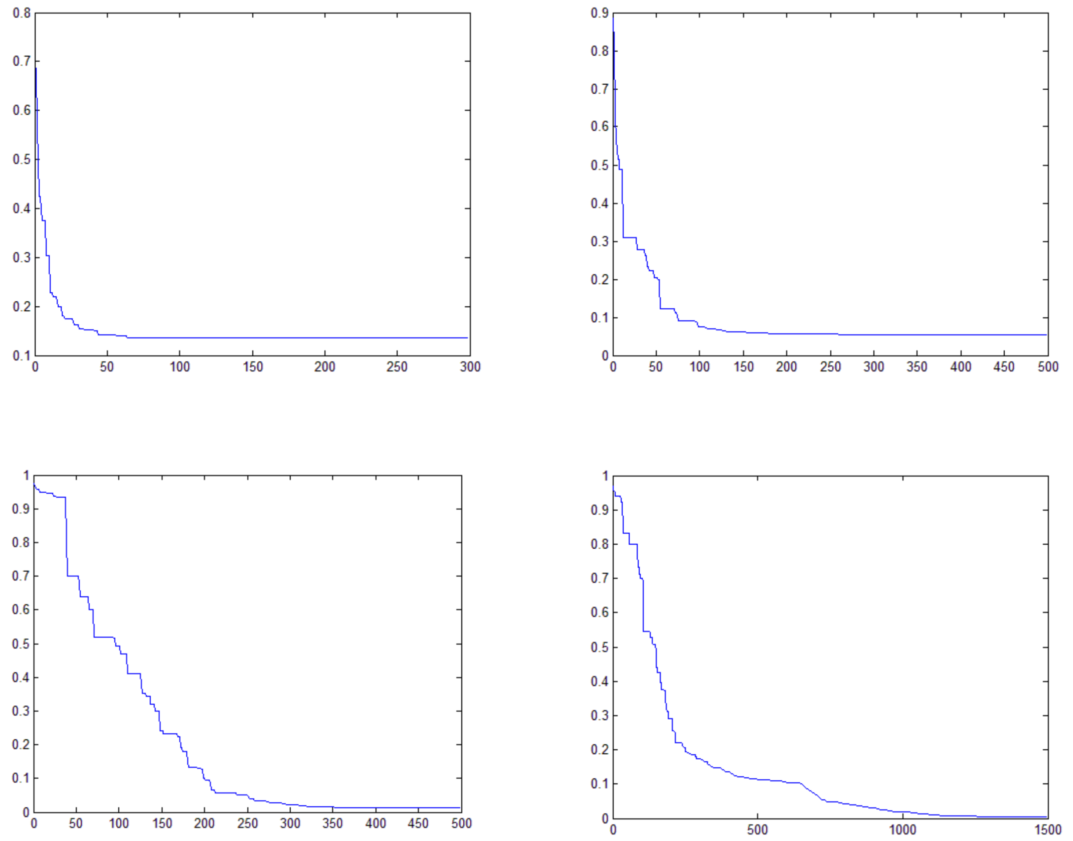


Fig. 3.21 Convergence behaviors of DE in design BS-FIR with orders of 12th, 24th, 36th, and 48th

The figures show that DE has better performance in designing bandstop filters with smaller population size, while the convergence speeds decrease with the increasing of the order of the filters.

In this part, we have presented linear phase digital FIR filter design using DE algorithm. The ripple error is much smaller than that of PSO, which indicates that DE algorithm takes an advantage in the global searching optimal ability when comparing their behavior in the higher order filters.

3.4.5 Type 2 Bandstop linear phase digital FIR filter

This section presents the simulations performed for the design of two kinds linear phase digital filters of type 2 FIR LP and BP filters.
The original population of each filter example is generated using the way which is same with PSO design. We randomly select 13, 25 figures from [0, 1] and sort these figures as one side of the initial population of the even symmetric filter of 25th-order and 49th-order, respectively. During the optimization process, the other side of the coefficients will be copied from the existing part.
Each filter order (N) is taken as 25 and 49 respectively. For the LP filter, passband (normalized) edge frequency w_p=0.45; stopband (normalized) edge frequency w_s=0.55; For the BP filter, lower stop band (normalized) edge frequency w_{s1}=0.25; lower passband (normalized) edge frequency w_{p1}=0.35; upper passband (normalized) edge frequency w_{p2}=0.6; upper stopband (normalized) edge frequency w_{s2}=0.7.

Lowpass linear phase digital FIR of order 25

Table 3.33 Coefficients of 25th-order type2 LP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
$h(1) = h(26)$	0.023033356250670	$h(8) = h(19)$	0.083292460688744
$h(2) = h(25)$	-0.053450263829228	$h(9) = h(18)$	0.100176591983382
$h(3) = h(24)$	-0.038263441099850	$h(10) = h(17)$	-0.139346949506319
$h(4) = h(23)$	0.034480236966874	$h(11) = h(16)$	-0.193029464401514
$h(5) = h(22)$	0.040317948956492	$h(12) = h(15)$	0.334506944438699
$h(6) = h(21)$	-0.055976073572788	$h(13) = h(14)$	0.999934759415984
$h(7) = h(20)$	-0.062831910150983		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.34, respectively.

Table 3.34 Design result of 25th-order type 2 LP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
63.593750	0.037769141721847	0.037769141721847

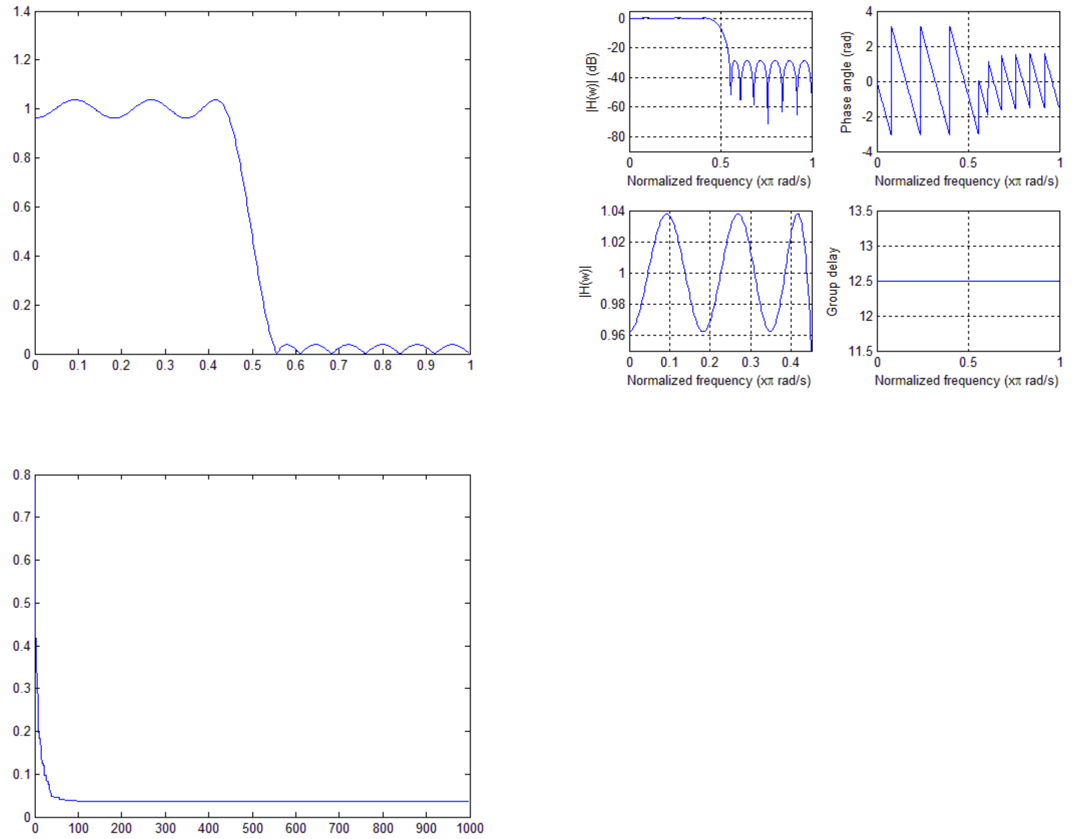


Fig. 3.22 Magnitude, ripple errors of 25th-order type2 LP-FIR filter by DE

Low pass linear phase digital FIR of order 49

Table 3.35 Coefficients of 49th-order type2 LP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(50)	0.001988522032624	h(14) = h(37)	-0.029891732375301
h(2) = h(49)	-0.004601587405793	h(15) = h(36)	-0.030664226189001
h(3) = h(48)	-0.003785858365796	h(16) = h(35)	0.039212234862564
h(4) = h(47)	0.005740073239494	h(17) = h(34)	0.041188796371963
h(5) = h(46)	0.005684438146939	h(18) = h(33)	-0.053623391996254
h(6) = h(45)	-0.008099943822724	h(19) = h(32)	-0.059803202930340
h(7) = h(44)	-0.007458366274536	h(20) = h(31)	0.078280556010685
h(8) = h(43)	0.011217473009429	h(21) = h(30)	0.092149805338929
h(9) = h(42)	0.012199673083538	h(22) = h(29)	-0.126567713252885
h(10) = h(41)	-0.015066808798298	h(23) = h(28)	-0.175621276774517
h(11) = h(40)	-0.015748000441024	h(24) = h(27)	0.301104551651995
h(12) = h(39)	0.021224167574810	h(25) = h(26)	0.899146443483012
h(13) = h(38)	0.021628782233848		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.36, respectively.

Table 3.36 Design result of 49th-order type 2 LP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
463.312500	0.005303839450343	0.005202975378994

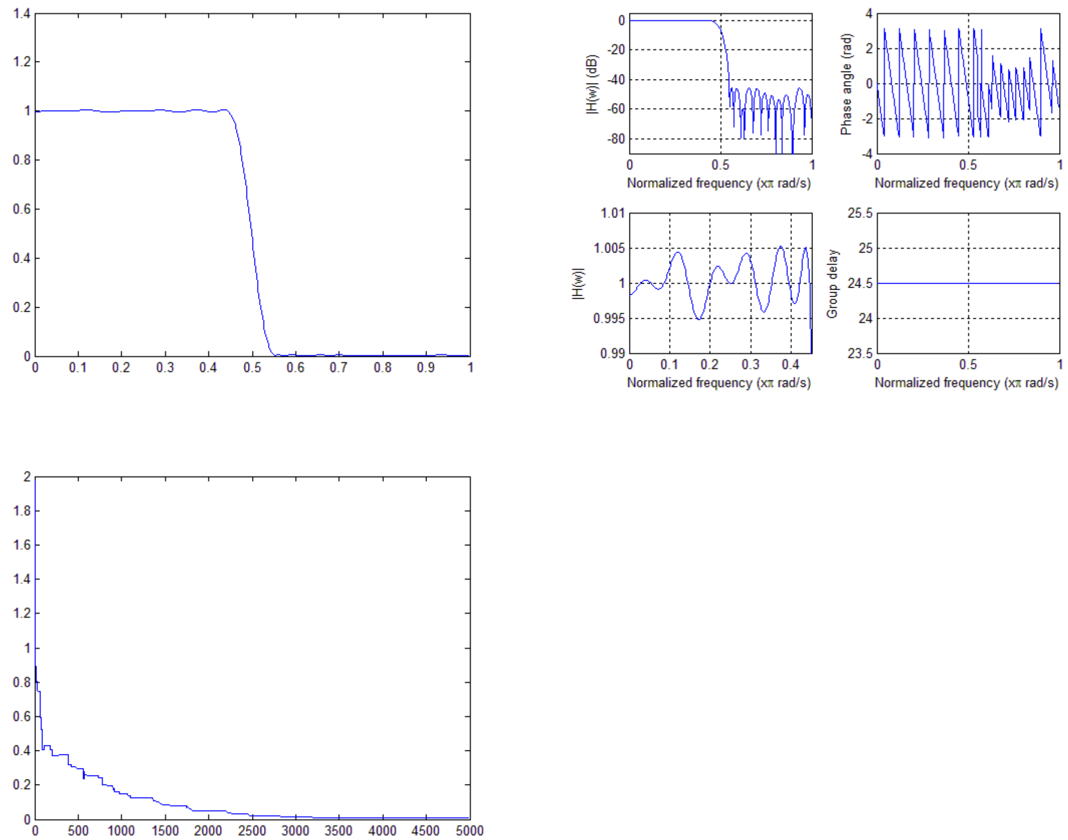


Fig. 3.23 Magnitude, ripple errors of 49th-order type2 LP-FIR filter by DE

3.4.6 Type 2 Bandpass linear phase digital FIR filter

Bandpass linear phase digital FIR of order 25

Table 3.37 Coefficients of 25th-order type2 BP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(26)	-0.062639418590278	h(8) = h(19)	0.024567707572210
h(2) = h(25)	0.018529295898912	h(9) = h(18)	-0.273089954532336
h(3) = h(24)	-0.086418369853777	h(10) = h(17)	-0.291934463714947
h(4) = h(23)	0.016838698812495	h(11) = h(16)	0.805048712327691
h(5) = h(22)	0.243924732505448	h(12) = h(15)	0.751910494681648
h(6) = h(21)	0.051436938252572	h(13) = h(14)	-0.986665334039993
h(7) = h(20)	-0.126359068280142		

The computational time, passband and stopband ripple error of the linear phase digital FIR filter design with DE algorithm is showed in Table 3.38, respectively.

Table 3.38 Design result of 25th-order type 2 BP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
435.125000	0.047106598726385	0.047106598726385

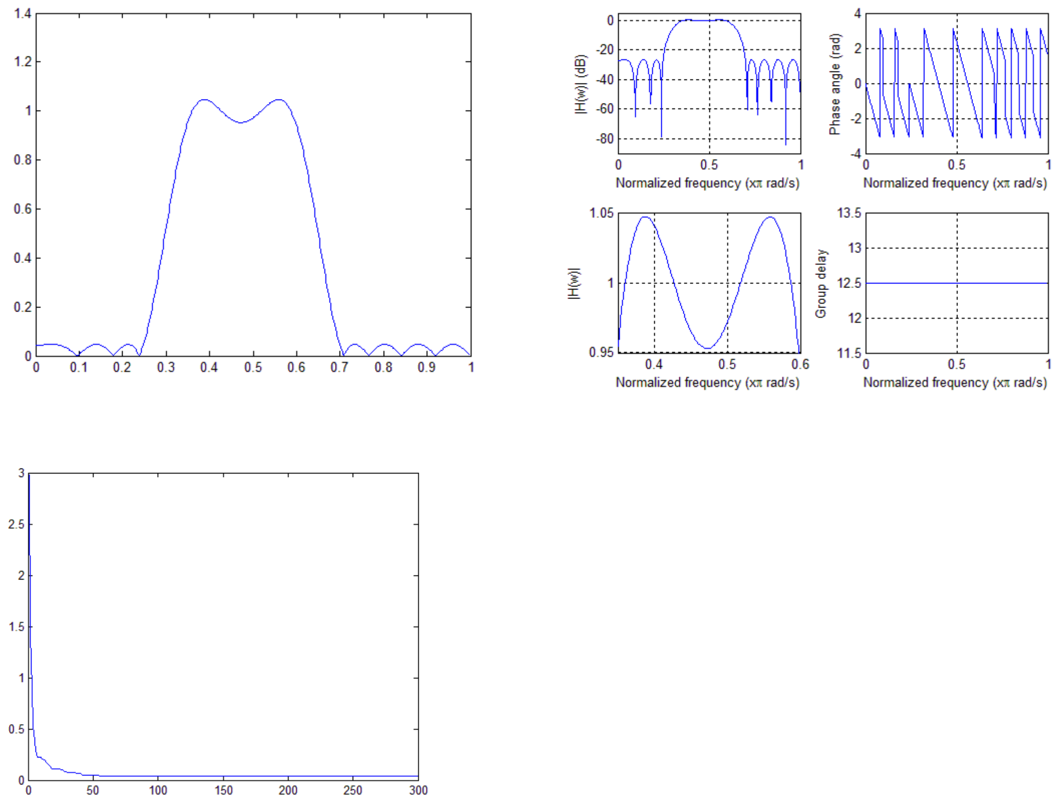


Fig. 3.24 Magnitude, ripple errors of 25th-order type2 BP-FIR filter by DE

Band pass linear phase digital FIR of order 49

Table 3.39 Coefficients of 49th-order type2 BP-FIR filter by DE

h(n)	coefficients	h(n)	coefficients
h(1) = h(50)	-0.002937252291331	h(14) = h(37)	0.000660188037631
h(2) = h(49)	0.007061619925994	h(15) = h(36)	-0.084715343792808
h(3) = h(48)	-0.001658923803247	h(16) = h(35)	-0.000961135944451
h(4) = h(47)	0.016000045694439	h(17) = h(34)	0.238408628796848
h(5) = h(46)	0.020430954897314	h(18) = h(33)	0.039859880371919
h(6) = h(45)	-0.027741624072272	h(19) = h(32)	-0.139139475300554
h(7) = h(44)	-0.023425918632631	h(20) = h(31)	0.015927912482715
h(8) = h(43)	0.006845362521121	h(21) = h(30)	-0.296591019285210
h(9) = h(42)	-0.018332896722350	h(22) = h(29)	-0.308198888978923
h(10) = h(41)	0.028582125796559	h(23) = h(28)	0.804466503630114
h(11) = h(40)	0.084660102977613	h(24) = h(27)	0.745570636805291
h(12) = h(39)	-0.031927199752879	h(25) = h(26)	-0.995414379498280
h(13) = h(38)	-0.069163836444190		

The computational time, passband and stopband ripple error of the linear phase digital FIR

filter design with DE algorithm is showed in Table 3.20, respectively.

Table 3.20 Design result of 49th-order type 2 BP-FIR filter by DE

Time/s	Passband ripple	Stopband ripple
962.765625	0.005934783442553	0.005939168568020

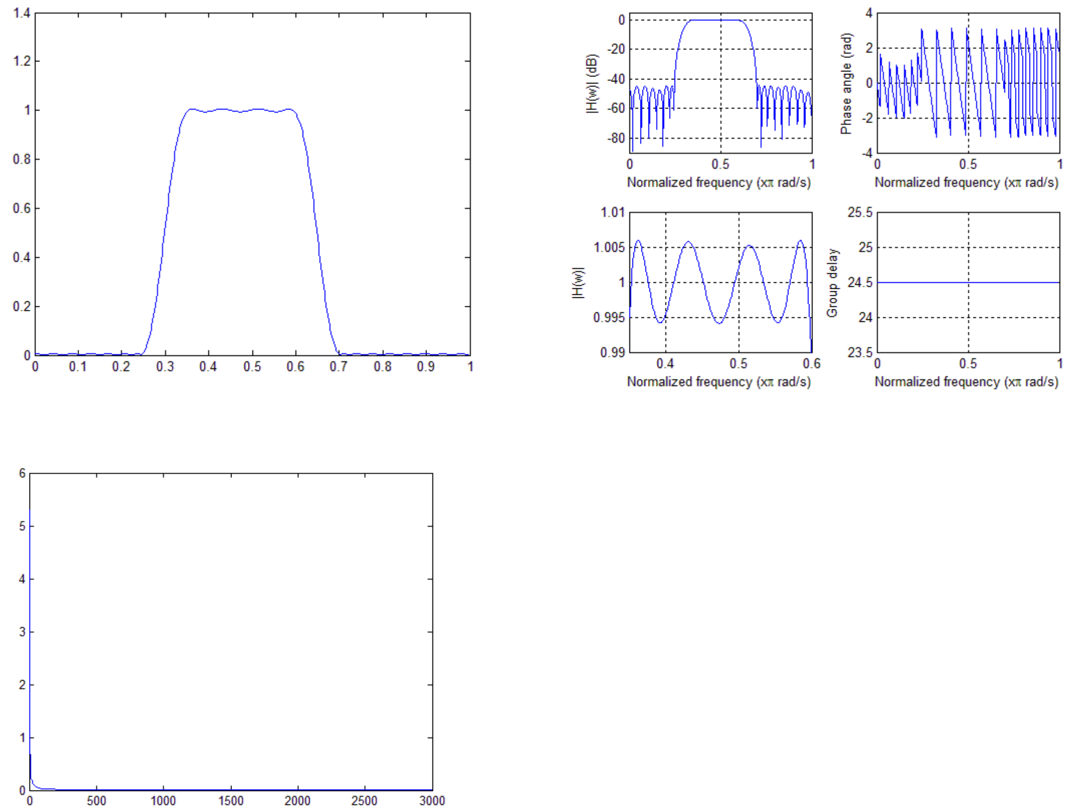


Fig. 3.25 Magnitude, ripple errors of 49th-order type2 BP-FIR filter by DE

3.5. Comparing with PSO

By comparing the results above and the results in chapter 3, we could know the stopband and passband of ripple error of both algorithms are approximately equal for both type1 and type2. The difference is in the transition band. From the results generated by the two algorithms in Fig 2.6, Fig2.11, Fig 2.16, Fig2.21 and Fig 3.6, Fig3.11, Fig 3.16, Fig3.21, it can be concluded that DE works better than PSO with more rapid convergence speed.

As shown in the above figures, it is noticed that for the same level of transition width, the DE results in the lowest stopband ripples for all types of filters, especially in the region near to the transition band. Besides, for the passband attenuation error, they have a similar level. The experimental results show that the DE algorithm achieves the best performance in the ripple error. Also, the algorithm is more stable, and the iterative algorithm can converge to the same solution. The performance of PSO algorithm is also acceptable, but the algorithm is a little unstable. PSO performs better in the design of lower order filters, but it works hard to design high dimensional filters which cost too much time. The final convergence result is easily affected by the parameter size and initial population influences. While along with the stopband frequency goes away, the stopband attenuation decreases using PSO comparing with that of DE. In fact, if we sacrifice the requirement in passband error a little, the stopband attenuation could perform much better than now. Due to PSO algorithm uses a global solution in its iterations to help direct its optimum, which is useful for optimizing low dimension optimization, while filter design is a comples optimization problem and concerned non-linear transformation so that the global solution may lead to pre-mature convergence.

3.6 Conclusion

The DE algorithm is a simple but powerful heuristic approach mainly having several advantages; finding the true global minimum regardless of the initial parameter values, fast convergence, and using a few control parameters. From the simulation studies, it was observed that the convergence speed of the DE is relatively fast and find satisfying solutions for all the questions. Results indicate that DE algorithm is a promising approach for designing these types of linear phase digital FIR filters which are mentioned in this chapter. Meanwhile, the ability to search optimum in such a limited domain shows its potential to apply in the other types of filters regarding the actual need.

The DE algorithm has been applied to the design of linear phase digital FIR filter design with different orders. Meanwhile, due to DE has a more simple structure than GA, works in the continuous domain while GA needs to transfer to the binary domain, it converges much faster than GA. Although the convergence speed decreased when designing high order bandstop filters, the results are still acceptable within 4000 iterations. Experiment results show the DE algorithm has successfully designed FIR filter with desired magnitude response and also can avoid local optimal solution, also, it solution is comparable with PSO. Therefore, the DE algorithm can be successfully used in linear phase digital FIR filter design.

Chapter 4 Linear Phase Digital FIR Filter Design Using Improved DE Algorithms

4.1 Abstract

The cost of time is the most important challenge isuue in the optimization problem, and the convergence speed and mutation progress have a vital effect on the optimization process. Some researchers have tried many ways to increase the speed of the convergence so as to decrease the computational time without changing the platform of the computing hardware and software, such like to increase the probability of selection of evolved individuals and consequently can improve the performance of the DE algorithm. Results show that the modified DE algorithm really can decrease the time of computing and does not hurt the optimization solution in designing linear phase digital FIR filters with lower orders.

4.2 Modified DE algorithm

In recent years, some researchers proposed several new improved DE algorithm, such as Gang Liu (2010) [28], Vasundhara (2013) [29], Shekhar Sharma (2014) [30], while the discussion is all limit to lower order lowpass or highpass linear phase digital filter design.

Recently, considering its computational time, Shamekhi [18] provides a new method to improve the algorithm's efficiency. In fact, the DE algorithm costs time to iterate each time. For DE/rand/bin, each evolves regarding three other random selected particles; one is the current individual and the other two for generating the difference, then compare their fitness values. If the fitness value of the evolved individual is better than that of the original one, then using the new individual to replace the original, otherwise keep the primary one. In all of the types of DE algorithms, they use this mechanism to produce new generations by calculating too many times. In fact, some of the mutations are not useful. In many practices, the evolved individuals for optimizing the average fitness are time- consuming. If there is a way to decrease the number of meaningless iterations in DE algorithms while keeping the efficiency of the mutation process, the time of computation will be decreased, and the performance will be improved. Meanwhile, the selection process also compares the evolved individual with the original one. In this process, the one which has a better fitness value will be selected to be parents in the following iterations. Based on this interpretation, Shamekhi [18] proposed a new job to modify the evolution process.

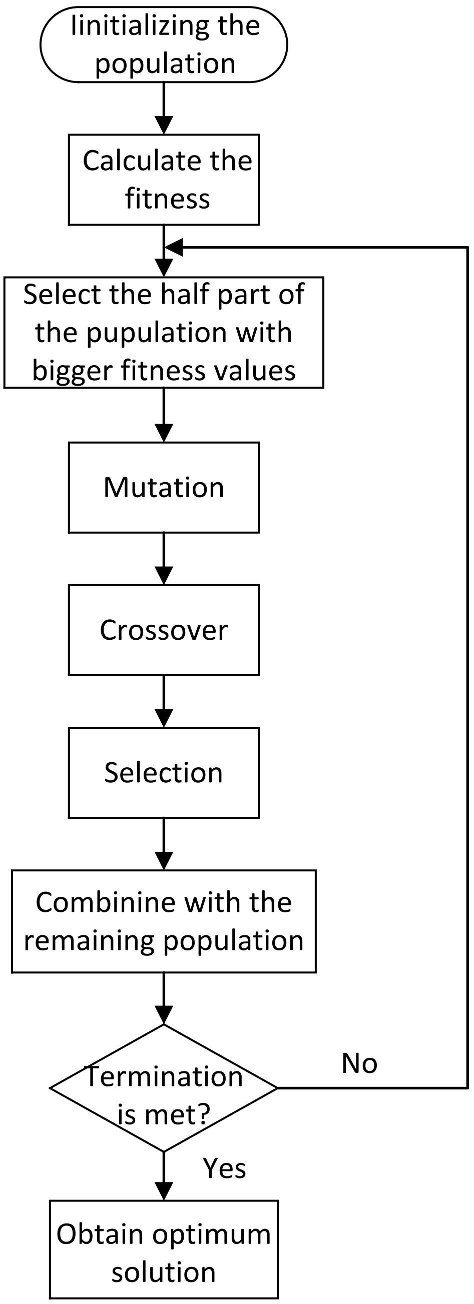


Fig. 4.1 The flow chart of modified DE algorithm

In the new version of DE algorithm, all the particles will be sorted according to their fitness values. This new method does not change the DE algorithm on the stage of mutation which uses three random selected individuals. Then, to increase the number of successful tries in the selection process, Shamekhi [18] only chooses half of the population which have worse fitness values in every generation. This modification can increase the probability of the selection of evolved individuals and consequently can improve the performance of the DE algorithm. Also, in this chapter, we can see that Shamekhi's DE [18] algorithm converges faster than the single strategy DE algorithm.

Table modified DE algorithm (Modified DE/BEST/1/BIN)

Step 1: Select Np individuals $pop_{i,g}$ in the constrained domain.

Step2: Calculate fitness values for all the individuals. $f_{i,g} = f(pop_{i,g}), i = 1{:}Np$

Step3: Divide the total population into two parts. If Np is even then let $n = Np/2$, else $n = (Np + 1)/2$

Step 4: Do steps 5 to 14 until convergence criteria satisfied

Rearrange the population based on their fitness values

Step 5: For $i = 1{:}Np$, do step 6

Step 6: For $j = i + 1{:}Np$,

If $f_j > f_i$ then swap the position of $pop_{i,g}$ and $pop_{j,g}$, and their fitness, respectively.

Step 8: for $i = 1{:}m$ do

Select two different random indexes p_1 and p_2 among the population

Let $newpop_{i,g}$=$pop_{i,g} + F \cdot (pop_{p_1} - pop_{p_2}) + F \cdot (gbestpop - pop_{i,g})$

For $i = 1{:}n$ do

Step 9: Select random $rand$, ($rand$ belong to [0,1]),

If $rand < CR$, then $nextpop_{j,i,g} = newpop_{j,i,g}$ else $nextpop_{j,i,g} = pop_{j,i,g}$

If $f(nextpop_{i,g}) < f_{i,g}$, then $nextpop_{i,g} = \ nextpop_{i,g}$ else $nextpop_{i,g} = pop_{i,g}$

Step 10: For $i = m + 1{:}Np$, $pop_{i,g} = pop_{i,g}$

Here $pop_{i,g}$ represents a group of coefficients with specific length at position i in the generation of g. As we can see clearly, only part of the individuals which have worse fitness values are evolved in each generation. The others owing better fitness values survive and also do contribution to the mutations. The computational complexity can be decreased by doing this. Based on this, the linear phase digital FIR filter design is also modified, and the corresponding results are shown bellows.

4.3 Experiments and result

We applied Shamekhi's DE and original DE into the linear phase digital FIR filter design.

The initial population is generated by the method which is introduced above.

All the control parameters are the same with those were used in chapter 4. We set the other parameters same values for both the algorithms for fair performance comparison. The following parameters are used in our experiments：

1) Population size 200;

2) The maximum number of generations: 200 for filters of order 12, 500 for filters of order 24, 800 for filters of order 36.

Therefore, Shamekhi's DE algorithm [18] and original DE algorithm used the same population size and the same stopping criteria in our experiment.

The average results of 20 independent runs are summarized in Table 4.1 to Table 4.4

Condition: due to the time of each simulation is not steady so that we choose to the average value of 100 times results.

Table 4.1 Computational time of the type1 LP linear FIR filter

Algorithm	12 order	24 order	36 order
Standard DE time/s	57.500000	92.8125	120.250000
Shamekhi's DE algorithm time/s	50.078125	87.046875	117.625000

Table 4.2 Computational time of the type1 HP linear FIR filter

Algorithm	12 order	24 order	36 order
Standard DE time/s	54.406250	43.578125	117.218750
Shamekhi's DE algorithm time/s	50.062500	41.781250	116.281250

Table 4.3 Computational time of the type1 BP linear FIR filter

Algorithm	12 order	24 order	36 order
Standard DE time/s	88.718750/100	95.265625	137.359375
Shamekhi's DE algorithm time/s	82.015625/80	90.578125	137.203125

Table 4.4 Computational time of the type1 BS linear FIR filter

Algorithm	12 order	24 order	36 order
Standard DE time/s	107.015625	182.796875	362.781250
Shamekhi's DE algorithm time/s	103.656250	180.546875	370.859375

All these results are obtained under the same stop criteria condition. In this way, the computational time can reflect the efficiency of these two algorithms under the same condition.

The comparison shows that Shamekhi's DE algorithm [18] spent less time on calculation than original DE algorithm. Meanwhile, its amplitude response behaves as well as the standard one. As for the increase of the filter's order, the computational time of improved algorithm is much less than that of the original algorithm, which is relevant with the complexity of the coefficients.

Fig 4.2 to Fig 4.5 show the convergence speed comparison of Shamekhi's DE algorithm and the original DE algorithm. The red lines represent the new DE algorithm, and the blue ones are the original DE algorithm.

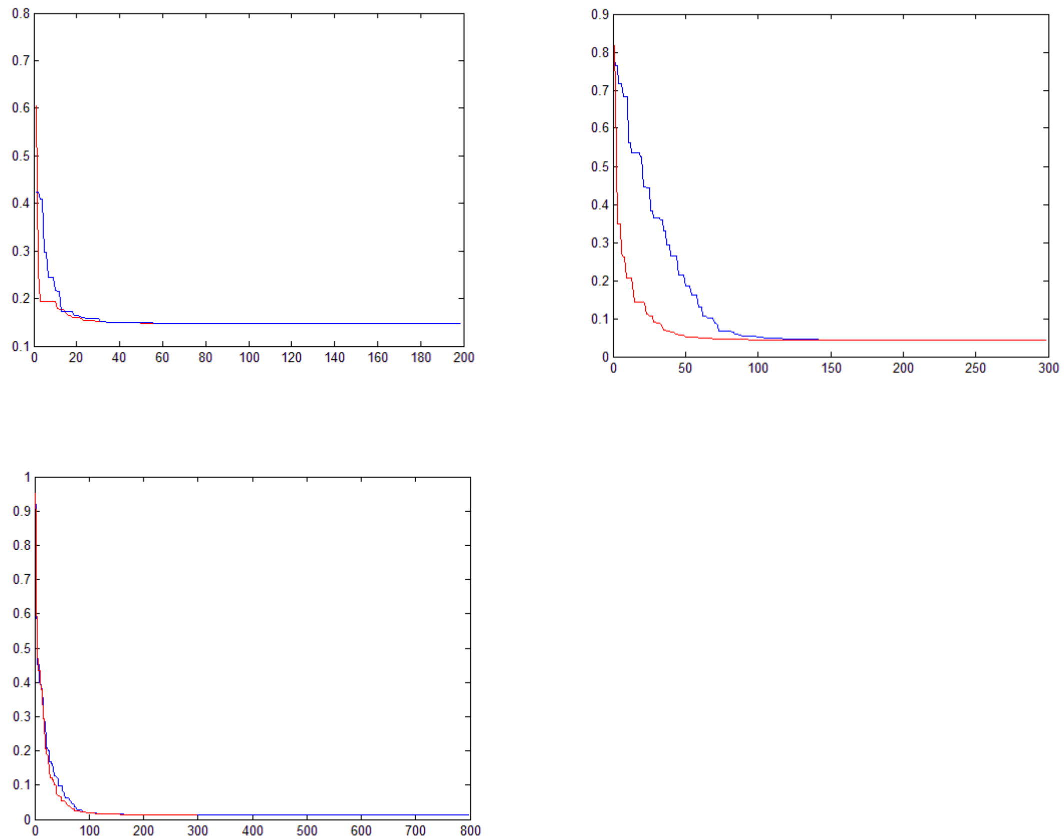


Fig. 4.2 Convergence speed using Shamekhi's DE algorithm and original DE algorithm when designing type 1 Lowpass FIR filter of order 12, 24, and 36.

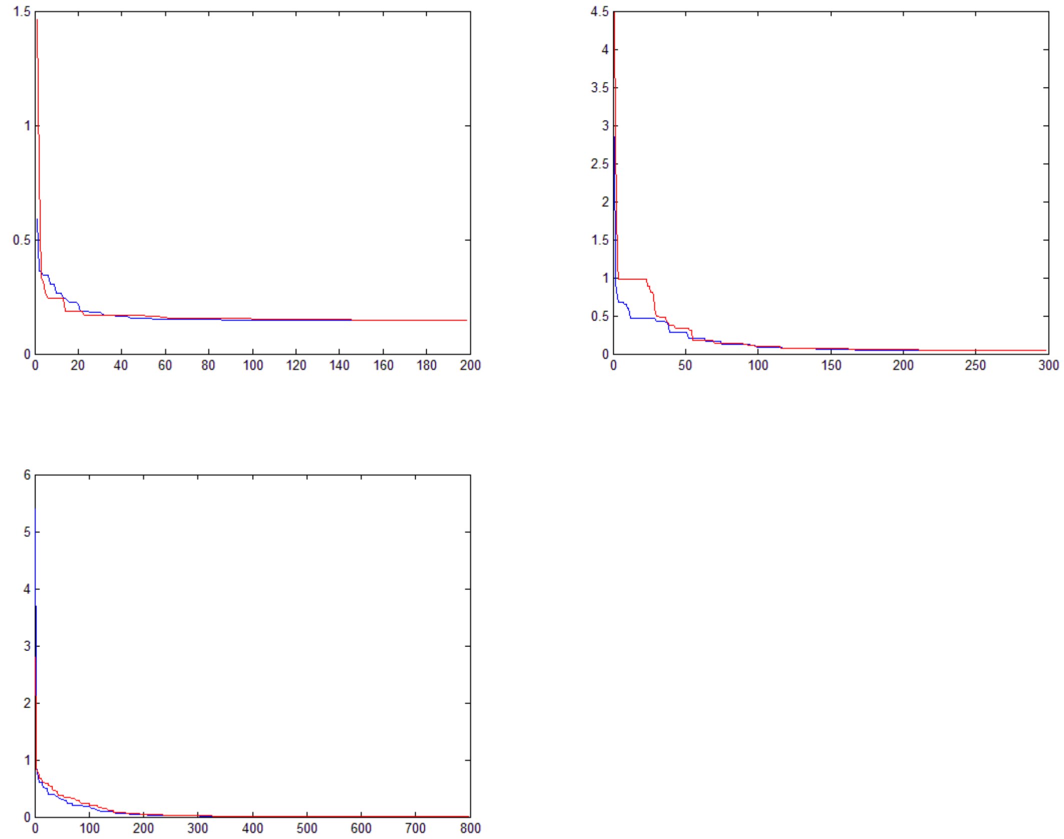
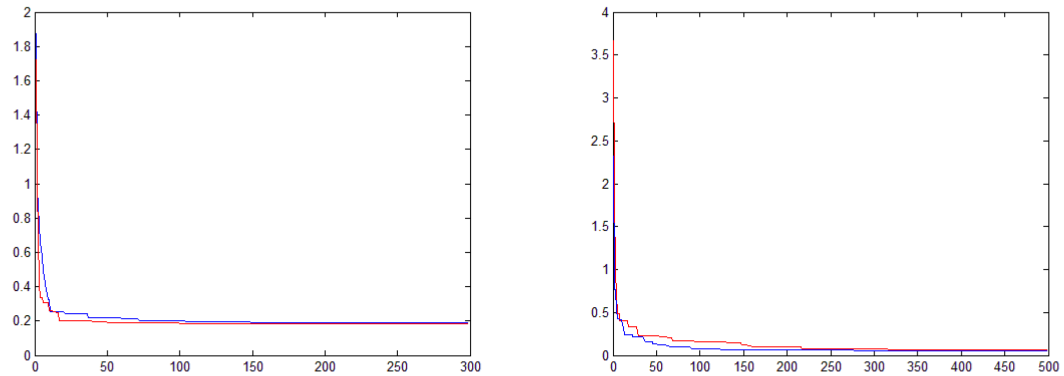


Fig. 4.3 Convergence speed using Shamekhi's DE algorithm and original DE algorithm when designing type 1 Highpass FIR filter of order 12, 24, and 36.



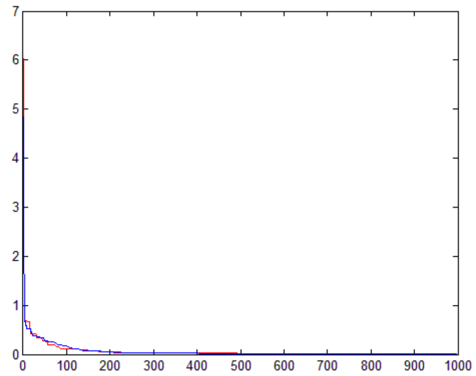


Fig. 4.4 Convergence speed using Shamekhi's DE algorithm and original DE algorithm when designing type 1 Bandpass FIR filter of order 12, 24, and 36.

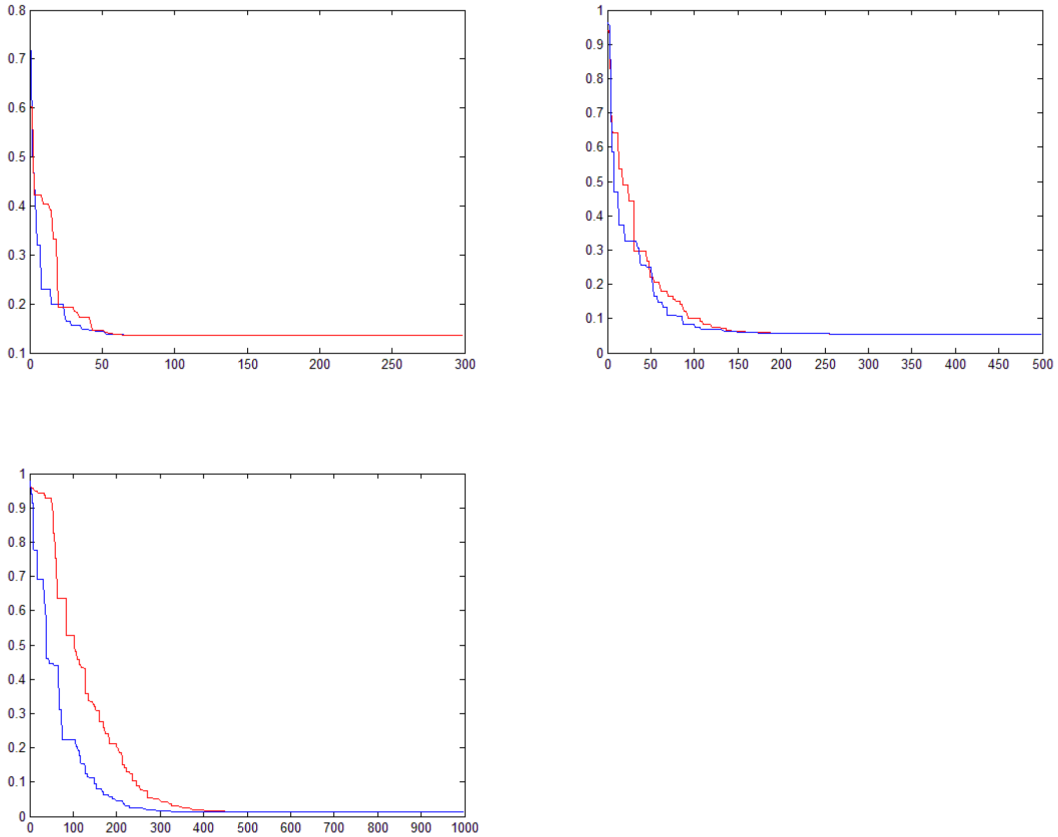


Fig. 4.5 Convergence speed using Shamekhi's DE algorithm and original DE algorithm when designing type 1 Bandstop FIR filter of order 12, 24, and 36.

As can be seen from above, for the four kinds of linear phase digital filters, the Shamekhi's DE algorithm [18] algorithm converges much faster than the original DE algorithm. But their passband and stopband ripple error have no obvious difference. So the new DE algorithm takes on the better convergence ability and keeps searching precision ability in solving linear phase digital FIR filter design problems. A second finding is that it converges much faster when applying in designing lower order digital FIR filter. When the order of digital FIR filter grows, the calculation complexity increases at the same time in an exponential way. In that condition, the decrease in the population counts for a weak part. Besides, when using this method to design filters of order 48, we cannot get an ideal result because it loses too much information when cutting the population into two equal part, that's why we did not design digital FRI filters of order 48. How to decrease the computation amount and also keep abundant information is the direction for researchers to work in future.

DE algorithm has proved to get a wonderful result in digital FIR filter design for both type 1 and type 2 while costing less time. It depends on the mutation strategy and crossover strategy, and the values of associated control parameters. For complex optimization problems like digital FIR filter design, the different mutation strategy, and crossover strategy are more effective than single mutation strategy and crossover strategy. Chunfeng Song [19] has proved the multi- population and multi-strategy (MPMSIDE) are more effective in solving high dimension optimization problem.

4.4 Multi-Population and Multi-Strategy Differential Evolution Algorithm

Several improved DE algorithms like unmanned aerial vehicles (UVAs) [19], aims to overcome the premature convergence and falling into local optimum problems. While there is still some space to improve in the local search ability, convergence speed and optimization accuracy of the algorithm.

The second modification of DE algorithm is proposed by Song [19] which aims to improve the stability and computational time in solving the high dimensional optimization problem, which is called MPMSIDE [19]. The new DE algorithm is based on the standard version of DE, while divides the total population into three groups according to their fitness values. These sub-populations carry out the basic mutation, crossover and selection processes simultaneously. Meanwhile, they communicate with each other to compare the optimization effectiveness. Due to there are several different mutation strategies for searching better solutions with special advantages in different situations, they are used to raise and balance the global searching ability and improve the optimization efficiency.

4.5 The idea of MPMSIDE algorithm

The new algorithm uses real coding and adopts the same mutation, crossover, and selection operations with the standard DE algorithm. To improve the stability and computational time of the DE algorithm, MPMSIDE [19] performs well in optimizing in high dimensional problems like digital FIR filter design. The coefficients of the filter often appear more than 20, so that there is potential in applying MPMSIDE [19] in the filter designing. In the proposed MPMSIDE [19] algorithm, the total population is divided into three different subpopulations according to their fitness values, the standard deviation of fitness and distance between each of them. They are subpopulation with better fitness values, these with worse fitness values and the rest. Firstly, the better ones are used for local search and improve the convergence speed and precision. Secondly, the worse group are used for global search, avoid falling in a local optimum, last but not least, the general ones are used to balance the global optimum and local optimum. Through all these three operations, the coefficients can search in a larger space and find the global optimum. Obviously, the mutation operation is the vital part. It specified mutation strategy determines the evolved direction. The modified DE algorithm improves the global search ability and workable in the filter design.

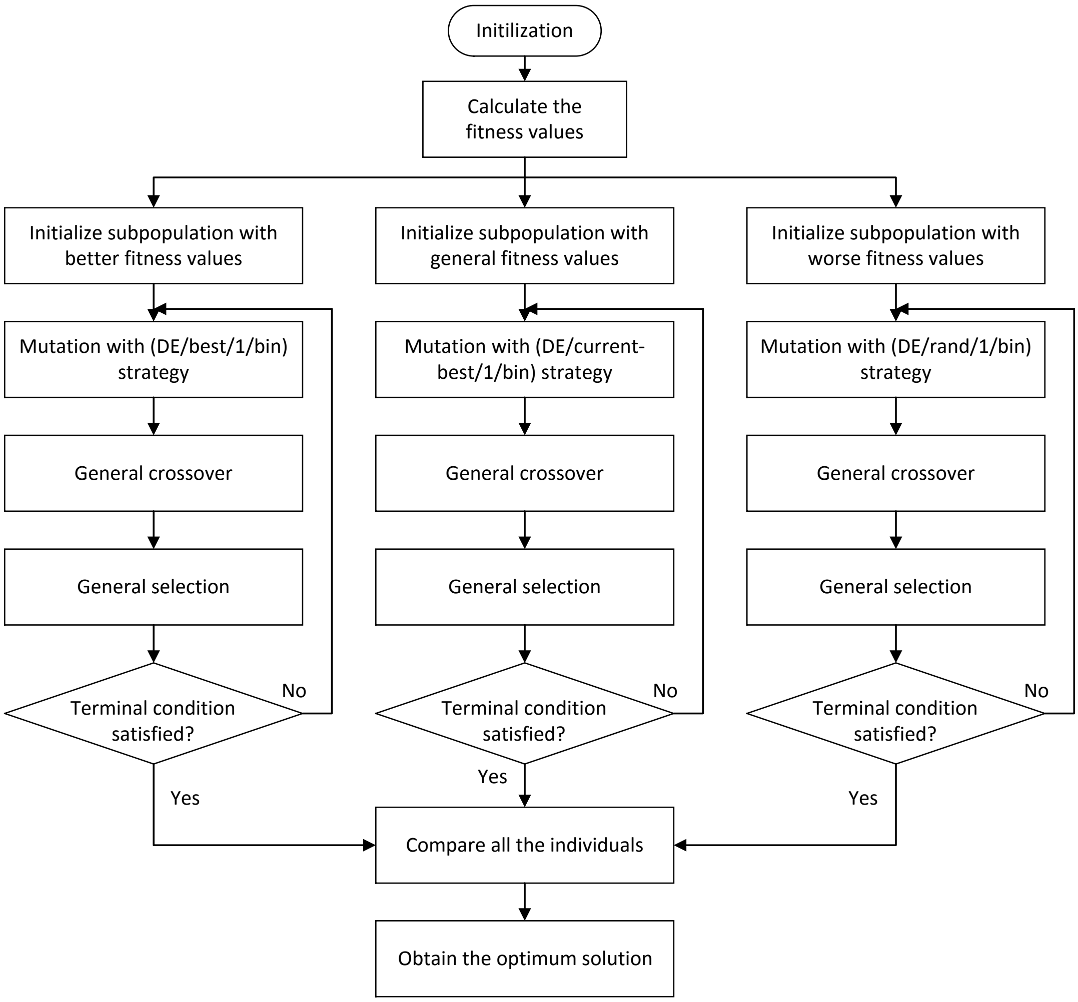


Fig. 4.6 The flow chart of MPMSIDE algorithm

4.6 Experiments and result

To compare the performance of the improved DE algorithm and the original one, the linear phase digital filters of type 1 with order 12, 24, 36, 48 and type 2 49th-order of LP, and BP filters are designed for comparison. The parameters are $F = 0.8, Cr = 0.8, Np = 800$ for both of them.

The coefficients are listed following.

Table 4.5 Coefficients of 48th-order type1 LP-FIR filter by MPMSIDE

h(n)	coefficients	h(n)	coefficients
h(1) = h(49)	-0.000554487341138	h(14) = h(36)	-0.041469769553842
h(2) = h(48)	-0.007717880370785	h(15) = h(35)	0.003008654128674
h(3) = h(47)	0.001053663147148	h(16) = h(34)	0.056724803201836
h(4) = h(46)	0.007584047105795	h(17) = h(33)	-0.003318073049108
h(5) = h(45)	-0.001164269548297	h(18) = h(32)	-0.079642790304331
h(6) = h(44)	-0.011272700451836	h(19) = h(31)	0.003584842504143
h(7) = h(43)	0.001522210893218	h(20) = h(30)	0.118966927875694
h(8) = h(42)	0.016077072149867	h(21) = h(29)	-0.003775693522217
h(9) = h(41)	-0.001895074334936	h(22) = h(28)	-0.206924730896512
h(10) = h(40)	-0.022331992066483	h(23) = h(27)	0.003900552618121
h(11) = h(39)	0.002281520547823	h(24) = h(26)	0.634056485549151
h(12) = h(38)	0.030517108842555	h(25)	0.994674724851117
h(13) = h(37)	-0.002651945744104		

Table 4.6Coefficients of 48th-order type1 HP-FIR filter by MPMSIDE

h(n)	coefficients	h(n)	coefficients
h(1)= h(49)	0.004942275010615	h(14)= h(36)	0.047493390335214
h(2)= h(48)	-0.006282654970427	h(15)= h(35)	-0.004010922957928
h(3)= h(47)	-0.003927505529981	h(16)= h(34)	-0.061672012472897
h(4)= h(46)	0.009280383975010	h(17)= h(33)	0.051507835019222
h(5)= h(45)	-0.001684662288334	h(18)= h(32)	0.052988341627539
h(6)= h(44)	-0.012778891499422	h(19)= h(31)	-0.108393537064484
h(7)= h(43)	0.010712235097072	h(20)= h(30)	0.005594382000799
h(8)= h(42)	0.009173041240655	h(21)= h(29)	0.169356413567068
h(9)= h(41)	-0.022265665760166	h(22)= h(28)	-0.147783461927907
h(10)= h(40)	0.003160233832021	h(23)= h(27)	-0.213652368323309
h(11)= h(39)	0.028791155433956	h(24)= h(26)	0.712763313025740
h(12)= h(38)	-0.024255486494168	h(25)	-0.946692506438855
h(13)= h(37)	-0.022391042949996		

Table 4.7Coefficients of 48th-order type1 BP-FIR filter by MPMSIDE

h(n)	coefficients	h(n)	coefficients
h(1)= h(49)	-0.001362346011734	h(14)= h(36)	0.021641754240610
h(2)= h(48)	-0.003457266292470	h(15)= h(35)	0.065010944017883
h(3)= h(47)	-0.001311452530341	h(16)= h(34)	-0.105883470698279
h(4)= h(46)	-0.019477973498044	h(17)= h(33)	-0.141481375467537
h(5)= h(45)	0.002694041074544	h(18)= h(32)	0.080445784872123
h(6)= h(44)	0.026038566537587	h(19)= h(31)	0.037209257626606
h(7)= h(43)	0.001246079757931	h(20)= h(30)	0.052376344430584
h(8)= h(42)	0.003316211648157	h(21)= h(29)	0.331524014345587
h(9)= h(41)	0.007253396695542	h(22)= h(28)	-0.144890741239610
h(10)= h(40)	-0.053918057746781	h(23)= h(27)	-0.787115713206999
h(11)= h(39)	-0.028249898819973	h(24)= h(26)	0.078350845654467
h(12)= h(38)	0.056911945217988	h(25)	0.995620861602009
h(13)= h(37)	0.016550475887058		

Table 4.8 Coefficients of 48th-order type1 BS-FIR filter by MPMSIDE

h(n)	coefficients	h(n)	coefficients
h(1) = h(49)	-0.004969599128776	h(14) = h(36)	-0.013292416597940
h(2) = h(48)	0.003661358181085	h(15) = h(35)	0.037714511012242
h(3) = h(47)	0.007140066990518	h(16) = h(34)	0.056535729975730
h(4) = h(46)	-0.000378010912006	h(17) = h(33)	-0.063428498744105
h(5) = h(45)	-0.000173114810690	h(18) = h(32)	-0.092537411237229
h(6) = h(44)	-0.000074089810956	h(19) = h(31)	0.050880172269761
h(7) = h(43)	-0.013968297538685	h(20) = h(30)	0.064372327233688
h(8) = h(42)	-0.002811146063786	h(21) = h(29)	-0.000642004733149
h(9) = h(41)	0.028421912868399	h(22) = h(28)	0.113699878374423
h(10) = h(40)	0.009266686468945	h(23) = h(27)	-0.056517323069444
h(11) = h(39)	-0.027263599404913	h(24) = h(26)	-0.879422069250240
h(12) = h(38)	-0.008638928675379	h(25)	0.081687581910393
h(13) = h(37)	0.001620370134812		

Table 4.9 Coefficients of 49th-order type2 LP-FIR filter by MPMSIDE

h(n)	coefficients	h(n)	coefficients
h(1) = h(50)	0.002343292269048	h(14) = h(37)	-0.029940381098606
h(2) = h(49)	-0.006188829028344	h(15) = h(36)	-0.030618340052104
h(3) = h(48)	-0.004416546423011	h(16) = h(35)	0.040234993650728
h(4) = h(47)	0.005614222668444	h(17) = h(34)	0.042451671050342
h(5) = h(46)	0.005523286571543	h(18) = h(33)	-0.055231526612771
h(6) = h(45)	-0.008571268333834	h(19) = h(32)	-0.060939129696917
h(7) = h(44)	-0.008207296696706	h(20) = h(31)	0.079840788048588
h(8) = h(43)	0.012025910077941	h(21) = h(30)	0.094380353931919
h(9) = h(42)	0.011674191465767	h(22) = h(29)	-0.129957035677705
h(10) = h(41)	-0.016555427181454	h(23) = h(28)	-0.178420229406395
h(11) = h(40)	-0.016196531458307	h(24) = h(27)	0.307542838416897
h(12) = h(39)	0.022339249711289	h(25) = h(26)	0.916605548688944
h(13) = h(38)	0.022335488776816		

Table 4.10 Coefficients of 49th-order type2 BP-FIR filter by MPMSIDE

h(n)	coefficients	h(n)	coefficients
h(1) = h(50)	-0.002867399871520	h(14) = h(37)	0.000996391032227
h(2) = h(49)	0.007482011621904	h(15) = h(36)	-0.078570919941793
h(3) = h(48)	-0.001774011656782	h(16) = h(35)	-0.000529974652108
h(4) = h(47)	0.015891573360854	h(17) = h(34)	0.223309567379082
h(5) = h(46)	0.018777257660204	h(18) = h(33)	0.036988546327860
h(6) = h(45)	-0.025876202468773	h(19) = h(32)	-0.130154229785927
h(7) = h(44)	-0.022607242157077	h(20) = h(31)	0.014695468042168
h(8) = h(43)	0.006855255827529	h(21) = h(30)	-0.276998934626188
h(9) = h(42)	-0.017487860498877	h(22) = h(29)	-0.287896753337353
h(10) = h(41)	0.026958014160008	h(23) = h(28)	0.752028459617320
h(11) = h(40)	0.079104063416477	h(24) = h(27)	0.696966308466942
h(12) = h(39)	-0.029655803487066	h(25) = h(26)	-0.930302725953316
h(13) = h(38)	-0.064863858334269		

This part will show the amplitude characteristics of these new designs for a direct understanding the performance of MPMSIDE [19].

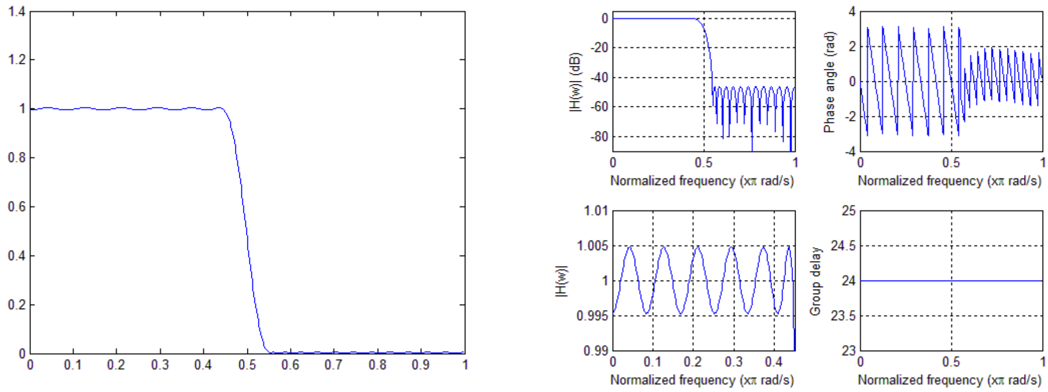


Fig. 4.7 Lowpass linear phase digital filter design of 48 order using MPMSIDE

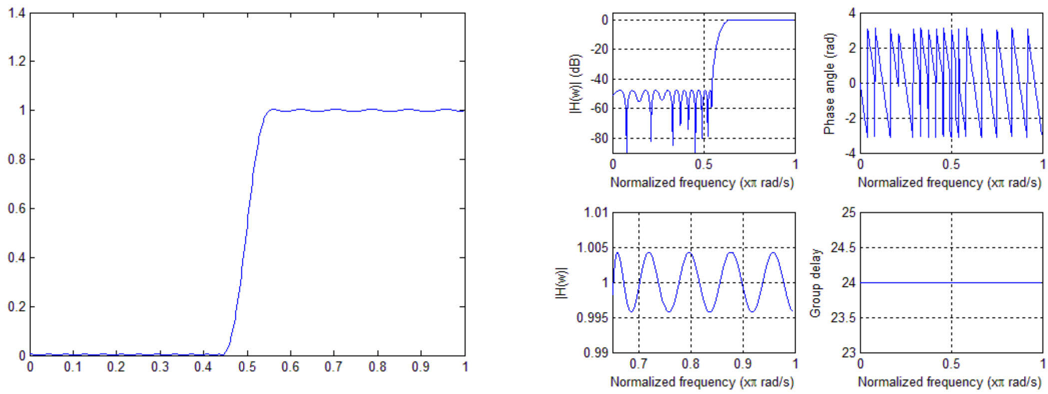


Fig. 4.8 Highpass Linear phase digital filter design of 48 order using MPMSIDE

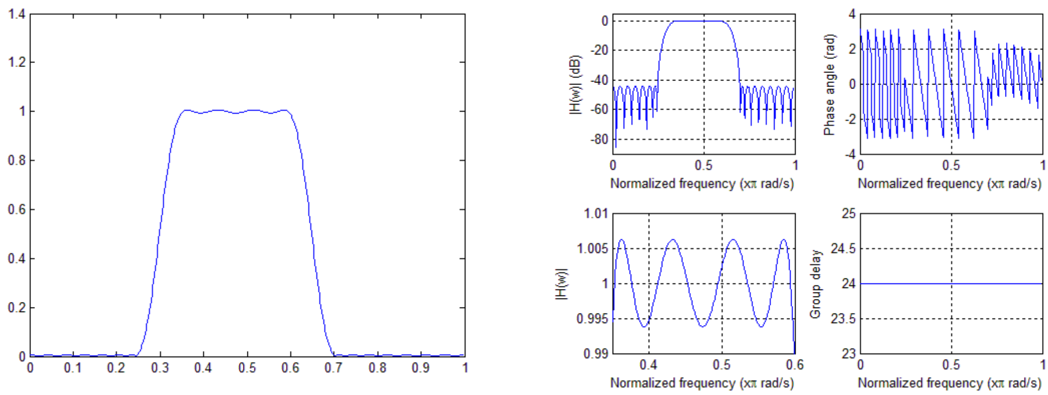


Fig. 4.9 Bandpass Linear phase digital filter design of 48 order using MPMSIDE

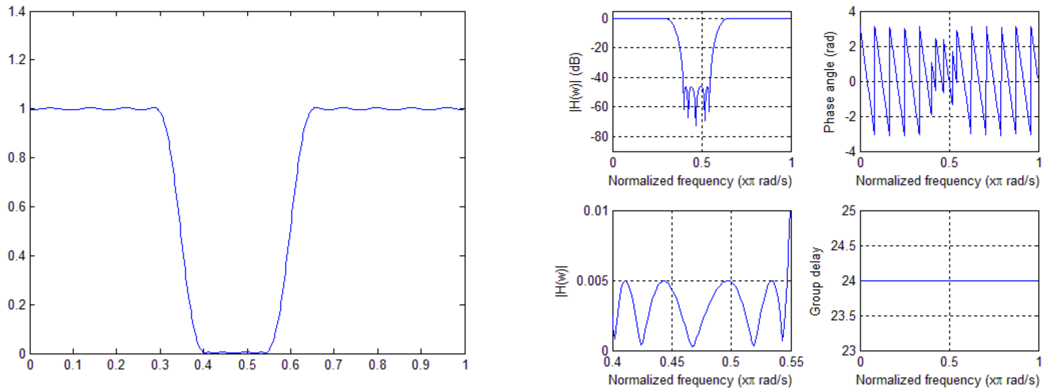


Fig. 4.10 Bandstop linear phase digital filter design of 48 order using MPMSIDE

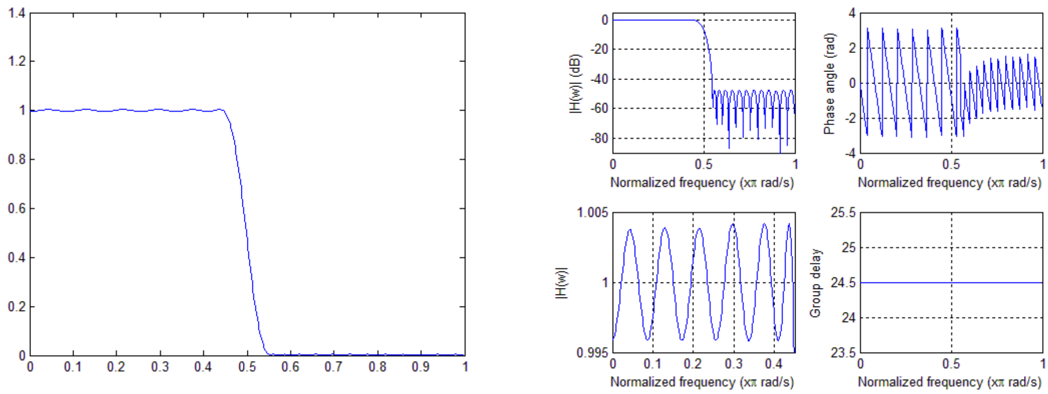


Fig. 4.11 Type 2 Lowpass linear phase digital filter design of 48 order using MPMSIDE

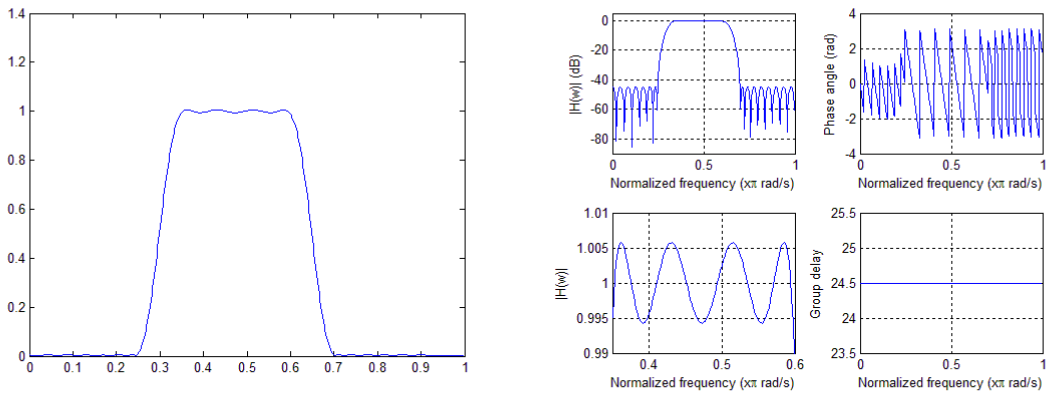


Fig. 4.12 Type 2 Bandpass linear phase digital filter design of 48 order using MPMSIDE

As seen from the following figures, MPMSIDE [19] costs more iterations to get the optima which on the other side show it searches more deeply the original algorithm.

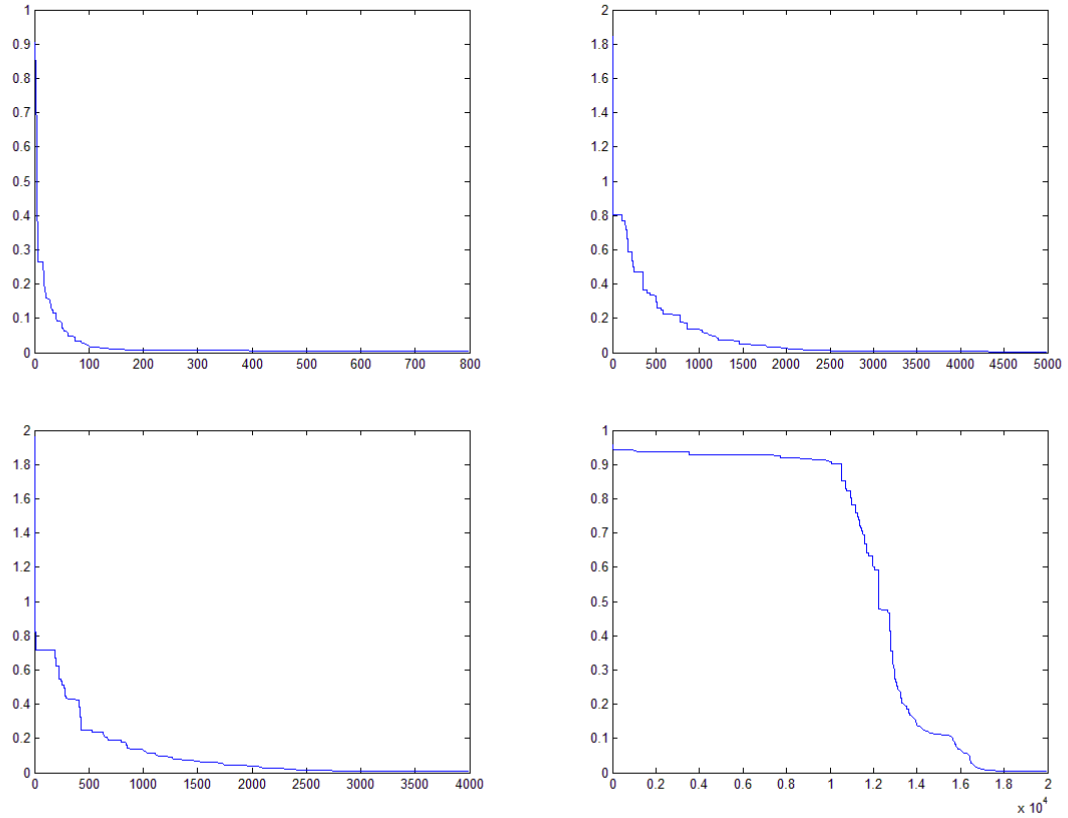


Fig. 4.13 Iterations of FIR filters with 48th-order of LP, HP, BP and BS of MPMSIDE

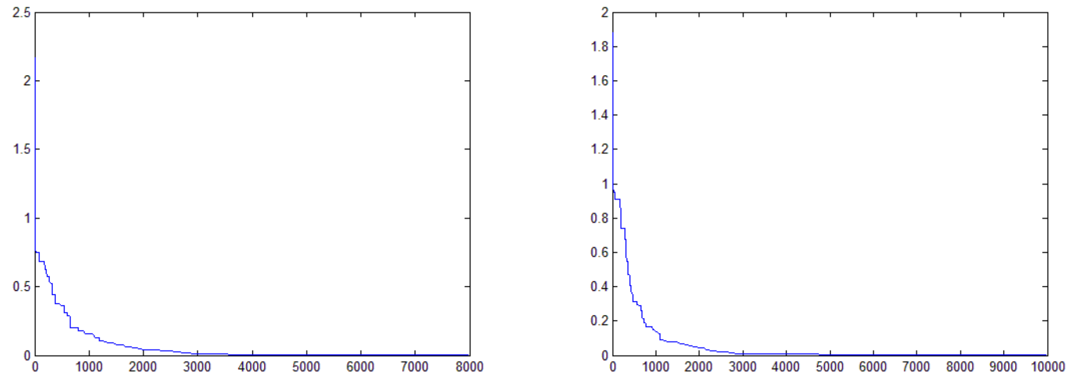


Fig. 4.14 Iterations of FIR filters with 48th-order of LP, HP, BP and BS of MPMSIDE

The tables following are the comparison between original DE algorithm and the improved DE algorithm.

Table 4.11 The comparison of MPMSIDE and the original DE algorithm

	Passband Ripple Error	Stopband Ripple Error
48th-order LP-FIR of MPMSIDE	0.004727419639804	0.004727412417021

48th-order LP-FIR of original DE	0.004770844291314	0.004778773524617
48th-order HP-FIR of MPMSIDE	0.004282558482891	0.004282559414818
48th-order HP-FIR of original DE	0.004729204159408	0.004729648325497
48th-order BP-FIR of MPMSIDE	0.006232409361277	0.006232674958691
48th-order BP-FIR of original DE	0.006254411520480	0.006255115566308
48th-order BS-FIR of MPMSIDE	0.004986279048719	0.004987331167942
48th-order BS-FIR of original DE	0.005163127949613	0.005174041464594
49th-order LP-FIR of MPMSIDE	0.004190352585071	0.004190070750841
49th-order LP-FIR of original DE	0.005303839450343	0.005202975378994
49th-order BP-FIR of MPMSIDE	0.005720617120792	0.005720614776317
49th-order BP-FIR of original DE	0.005934783442553	0.005939168568020

4.7 Conclusion

The first improved DE algorithm achieved superior performance in the convergence speed, which is due to the decreasing of the computing complexity. To increase the number of successful tries in the selection process of the DE algorithm, half of the population which has worse fitness are evolved for global optimal searching. This improvement can increase the probability of selection of evolved individuals and consequently can increase the performance of the DE algorithm. On the other hand, in the new version of DE algorithm, the individuals with better fitness can remain and survive while the worse ones have less chance to survive. The performance characteristics of the improved DE algorithm have been compared with the original one. Results obtained show that the modified algorithm converges much faster than the original algorithm and need fewer iterations to obtain optimal solution.

Many researchers have tried to improve the performance of the original DE algorithm since the invention of the algorithm. One of the most important problems is that the ability of global optimal searching. Many researchers started to develop several methods to adaptively adopt optimal set of control parameters of the algorithm to decrease the computational time and larger searching space. While DE algorithm finds the optimal point through an organized parallel interacting try and error process, the performance of DE would be improved if the number of unsuccessful tries could be decreased. That was the point in this chapter which is introduced in details. In the improved algorithm, in each generation, all individuals of the population are sorted based on their fitness values.

The improved mutation is introduced to improve the global optimization ability of the DE algorithm, avoid possible trap in a local minimum for dealing with complex problems with high dimension multimodal optimization issues. The local optimization strategy aims to avoid the local extreme point and enhance the local hill-climbing ability in the local search. The performance of MPMSIDE [19] algorithm is compared with the original DE algorithm for designing linear phase digital FIR filter in this chapter. The results show that the improved MPMSIDE [19] algorithm is overall more effective and takes on better searching precision and global optimization ability. The modified algorithm can get better magnitude response in 48 order FIR filter design. According to the results obtained, these modifications can improve the performance of the linear phase digital FIR filter design. What's more, the new algorithm costs much more time to converge which we need to consider in the following job.

Chapter 5 Conclusion and Further Research

5.1 Conclusion

In this thesis, we mainly use PSO and DE algorithms to design different kinds of linear phase digital FIR filters and the implementation of them are discussed. Direct search methods are widely used in applied science and engineering. They are a class of optimization methods which are easy to program, do not require any properties of the function, being optimized and are often claimed to be robust for problems with noisy function values. Hence, when the optimizing function is nonlinear, direct search methods are the methods of choice.

In Chapter 2, we introduced PSO as a method for optimization of continuous nonlinear functions. The method was discovered through simulation of a simplified social model. This paper describes the PSO concept regarding its precursors, briefly reviewing the stages of its development from social simulation to an optimizer. Its ties to artificial life in general, and to fish schooling, and swarming theory in particular. PSO is also related to evolutionary computation and has a connection with both GA and evolutionary programming. These relationships are briefly reviewed in the paper.

PSO is a very simple concept, and paradigms can be implemented in a few lines of computer code. It requires only primitive mathematical operators and is computationally inexpensive regarding both memory requirements and speed. Early testing has found the implementation to be effective with several kinds of problems. The application of the algorithm to the high dimensional question of linear phase digital also proves effective.

DE is an efficient and powerful population-based stochastic search technique for solving optimization problems over continuous space, which has been widely applied in many scientific and engineering fields. And its application is presented in chapter 4. Inspired by the natural evolution of species, DE has been successfully applied to solve numerous optimization problems in diverse fields. DE is an algorithm designed for minimizing a function of real variables. It is extremely robust in locating the global minimum. The overall structure of DE resembles that of PSO. Like the other population set-based direct search methods DE also attempts to guide an initial set S of points to the vicinity of the global minimum through repeated cycles of selection, mutation and crossover and acceptance.

The error analysis technique used in this thesis is called the weighted sum of the maximum deviation (SMD), which is a (For the $p = 2$) method to calculate minimax value of the all the strategies in this paper. Since the objective function is fixed and the maximum number iterations rely on the speed of the algorithm to locate the optimal solution, the performance of these algorithms can be evaluated based on their ability to meet the filter design Specifications.

Table 5.1 Simulation Results for PSO, PM, and DE

		Stopband peak error	Passband peak error
12th-order LP-FIR	PSO	0.151279546816109	0.150398090335351
	PM	0.150109047269449	0.150564579975670
	DE	**0.147463170638424**	**0.147463170638891**
24th-order LP-FIR	PSO	0.046248431572409	0.053365558441687
	PM	0.045264759603219	0.045062625148867
	DE	**0.043536322158584**	**0.043536385497003**
36th-order LP-FIR	PSO	0.016303333542335	0.016968828284386
	PM	0.014770025669699	0.014841449760834
	DE	**0.014029908627723**	**0.014029882185380**
48th-order LP-FIR	PSO	0.0073759340395515	0.0073776557844607
	PM	0.005052454212868	0.005052454212868
	DE	**0.004729204159408**	**0.004729648325497**
12th-order HP-FIR	PSO	0.149579872844649	0.152893741918110
	PM	0.150109047269449	0.150564579975670
	DE	**0.147463199395232**	**0.147463252897533**
24th-order HP-FIR	PSO	0.045813959055968	0.045990750663258
	PM	0.045264759603219	0.045062625148867
	DE	**0.044509573845163**	**0.044615847837733**
36th-order HP-FIR	PSO	0.015911927701090	0.016021904562858
	PM	0.014770025669699	0.014841449760834
	DE	**0.014281709927772**	**0.014345989865743**
48th-order HP-FIR	PSO	0.005342504707092	0.005322669623213
	PM	0.005052454212868	0.005052454212868
	DE	**0.004729204159408**	**0.004729648325497**
12th-order BP-FIR	PSO	0.208422639560898	0.186476983648942
	PM	0.193736337871682	0.191302184150198
	DE	**0.186688041227229**	**0.186687974413917**
24th-order BP-FIR	PSO	0.061758283413739	0.063718444449077
	PM	0.064326791198981	0.061879888338857
	DE	**0.058327396074577**	**0.058332583584781**
36th-order BP-FIR	PSO	0.037339573407747	0.022652997836862

	PM	0.023149119527607	0.021455863219901
	DE	**0.019501477342182**	**0.019499285954434**
48th-order BP-FIR	PSO	0.006662125873949	0.006669725210366
	PM	0.008614167159237	0.007612756825124
	DE	**0.006254411520480**	**0.006255115566308**
12th-order BS-FIR	PSO	0.107590998821996	0.163215646048442
	PM	0.146721699974909	0.148091298956491
	DE	**0.134842534921531**	**0.134842532491527**
24th-order BS-FIR	PSO	0.060703136028247	0.059849388853956
	PM	0.063103005793731	0.061522516533729
	DE	**0.054747732174394**	**0.054747770363342**
36th-order BS-FIR	PSO	0.020408839055092	0.023572479676884
	PM	0.028512470309936	0.027929164489408
	DE	**0.012053682650155**	**0.012078800209884**
48th-order BS-FIR	PSO	0.007381528454473	0.007252200677178
	PM	0.009547790487259	0.009683806950184
	DE	**0.005163127949613**	**0.005174041464594**
25th-order LP-FIR	PSO	0.041201848758123	0.041827017886749
	DE	**0.037769141721847**	**0.037769141721847**
49th-order LP-FIR	PSO	0.006352918120974	0.006352918489874
	DE	**0.005303839450343**	**0.005202975378994**
25th-order BP-FIR	PSO	0.048768630452890	0.048847523383255
	DE	**0.047106598726385**	**0.047106598726385**
49th-order BP-FIR	PSO	0.007445696141175	0.006589674897734
	DE	**0.005934783442553**	**0.005939168568020**

Experiments indicate PSO can converge to the global optima with less time consumption. And in almost all the designs, the ripple errors achieved are similar to Parks-McClellan algorithm. In details, PM performs much better in designing LP and HP filters, but PSO outperforms PM in the other two kinds of BP and BS filters. PSO only outperforms the others in the design of 12th-order BP-FIR.In summary, all algorithms tested for the continuous problem was able to minimize the objective function to a reasonable point, PSO and PM can meet the specification in the continuous domain with the negligible discrepancy. DE performs best regarding execution speed and accuracy than ACO. DE shows a great advantage in these designs.

The optimization of such a high-dimensional non-linear FIR filter design function is a difficult problem occurring in engineering applications. We also have tried several versions of DE methods to solve this problem. From the comparison in the previous section, it is quite clear that the new algorithms outperform the original DE in the convergence speed and ripple performance. What makes the improved DE algorithms more robust than the original one is the feature that not all points in the set S are randomly updated for each generation. In this way, DE has the potential to explore widely with the help of the existing points of the current generation.

Table 5.2 The comparison of MPMSIDE [19] and the standard DE algorithm

	Passband Ripple Error	Stopband Ripple Error
48th-order LP-FIR of MPMSIDE	**0.004727419639804**	**0.004727412417021**
48th-order LP-FIR of original DE	0.004770844291314	0.004778773524617
48th-order HP-FIR of MPMSIDE	**0.004282558482891**	**0.004282559414818**
48th-order HP-FIR of original DE	0.004729204159408	0.004729648325497
48th-order BP-FIR of MPMSIDE	**0.006232409361277**	**0.006232674958691**
48th-order BP-FIR of original DE	0.006254411520480	0.006255115566308
48th-order BS-FIR of MPMSIDE	**0.004986279048719**	**0.004987331167942**
48th-order BS-FIR of original DE	0.005163127949613	0.005174041464594
49th-order LP-FIR of MPMSIDE	**0.004190352585071**	**0.004190070750841**
49th-order LP-FIR of original DE	0.005303839450343	0.005202975378994
49th-order BP-FIR of MPMSIDE	**0.005720617120792**	**0.005720614776317**
49th-order BP-FIR of original DE	0.005934783442553	0.005939168568020

Table 5.2 shows the simulation results of the original DE algorithm and MPMSIDE. Almost in all of the experiments, MPMSIDE shows a better performance in the stopband and passband ripple error. In the MPMSIDE algorithm, the total population is divided into three part according to their fitness values: best population, worst population and general population according the fitness value. Linear phase FIR design is a kind of high dimension multimodal optimization problems. The combination of the knowledge from different

mutation strategies not only enhances the global optimization ability of the DE algorithm and increases the possibility to explore more deeply in the search space for better solution but also helps avoid the local extreme point and improve ability in the local search. The results show that the MPMSIDE algorithm is overall more effective and takes on better searching precision.

5.2 Further research

Genetic algorithm, particle swarm algorithm, differential evolution algorithm are all branches of the evolutionary algorithm which has been proved relatively efficient in high-dimensional non-linear problems. Many scholars have studied these algorithms through continuous improvement in improving the performance of the algorithm and expanding the application areas. It is necessary to discuss these algorithms characteristics, for specific applications and adapt to the ability to different problems for use will be very meaningful work.

Through experiments and literature analysis, PSO and DE algorithms for some of the indicators are summarized as follows:

(1) Parameter setting problem.

DE algorithm has two main parameters to adjust, and the parameter settings on the results of the impact are not obvious, it is easier to use. Considering the PSO algorithm parameters, different parameter settings on the final result is relatively large. So in actual use, constantly adjust sometimes increases the difficulty of the algorithm. Self-adaptive method is discussed in this paper, and further research in this aspect should be focused in the following work

(2) Computational complexity:

Computational complexity has a clear influence on the computational time of this problem regardigng the results achieved above. It would be of theoretical and practical importance to reduce the computational complexity of the proposed algorithm.

(3) Coding standard

PSO and DE are both using real coding while GA uses binary coding. In recent years, many scholars have applied the integer encoding in PSO and DE algorithms, especially 0-1 nonlinear optimization problem. It is a potential direction for improving the performance of the linear phase digital FIR filter design.

(4) High-dimensional issue.

PSO and DE can solve this problem very well, especially the DE algorithm, the convergence speed is very fast, and the result is very accurate. But the actual order of digital FIR filters could be much larger than those in this paper. Through experiments, we could find the result will be decreased along with the increasing of the order. In the practical problem, because of the very high dimensionality of the vector, it is very important to deal with the higher dimensional problem.

References

[1] A. Lee, M. Ahmadi, G. A. Jullien, W. C. Miller, and R. S. Lashkari, “Digital filter design using genetic algorithm,” In *Proceedings of Advances in Digital Filtering and Signal Processing, 1998 IEEE Symposium on,* pp. 34-38, 1998.

[2] Najjarzadeh, Meisam, and A. Ayatollahi, “FIR digital filters design: particle swarm optimization utilizing LMS and minimax strategie,” In *Proceedings of Signal Processing and Information Technology. IEEE International Symposium on*, pp. 129-132, 2008.

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

[4] P. C. Fourie, and A. A. Groenwold, “The particle swarm optimization algorithm in size and shape optimization,” *Structural and Multidisciplinary Optimization,* vol. 23, no. 4, pp. 259-267, 2002.

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

[6] N. Benvenuto, M. Marchesi and A. Uncini, “Applications of simulated annealing for the design of special digital filters,” In *Proceedings of IEEE Transactions on Signal Processing*, vol. 40, no. 2, pp. 323-332, February 1992.

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

[8] C. Sammut, and G. Webb. *Encyclopedia of machine learning*. New York: Springer, 2011.

[9] J. Kennedy. “Particle swarm optimization,” *Encyclopedia of machine learning*. Springer US, pp. 760-766, 2011.

[10] H. A. Abbass, “The self-adaptive pareto differential evolution algorithm.” In *Proceedings of Evolutionary Computation, 2002. CEC'02. In Proceedings of the 2002 Congress, IEEE Transactions on,* vol. 1, pp. 831-836, 2002.

[11] H. K. Kwan, *Optimization Methods for Digital Filter Design*, Edition 1.1, dfisp.org, ISBN: 9780993670794, 19 February 2016.

[12] R. Kar, D. Mandal, S. Mondal, and S. P. Ghoshal, “Craziness based particle swarm optimization algorithm for FIR band stop filter design,” *Swarm and Evolutionary Computation,* vol. 7, pp. 58-64, 2012.

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

[14] 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.

[15] J. Liu and J. Lampinen, "A fuzzy adaptive differential evolution algorithm." *Soft Computing,* vol. 9, no.

6, pp. 448-462, 2005.

[16] M. M. Ali, and A. Törn, "Population set-based global optimization algorithms: some modifications and numerical studies," *Computers and Operations Research*, vol. 31, no. 10, pp. 1703-1725, 2004.

[17] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems," In *Proceedings of Evolutionary Computation*, *IEEE Transactions On,* vol. 10, no. 6, pp. 646-657, 2006.

[18] A. Shamekhi, "An improved differential evolution optimization algorithm," *International Journal of Research and Reviews in Applied Sciences*, vol. 15, no. 2, pp. 132-145, 2013.

[19] C. Song, and Y. Hou, "An Improved Differential Evolution Algorithm for Solving High Dimensional Optimization Problem," *International Journal of Hybrid Information Technology*, vol. 8, no. 10, pp. 177-186, 2015.

[20] X. Y. Zhang and H. B. Duan, "An improved constrained differential evolution algorithm for unmanned aerial vehicle global route planning," *Applied Soft Computing*, vol. 26, pp. 270-284, 2015.

[21] K. Mullen, D. Ardia, D. Gil, D. Windover, and J. Cline, "DEoptim: An R package for global optimization by differential evolution," *Journal of Statistical Software*, vol. 40, no. 6, 2009.

[22] J. Ababneh, and M. Bataineh, "Linear phase FIR filter design using particle swarm optimization and genetic algorithms," *Digital Signal Processing*, vol. 18, no. 4, pp. 657-668, 2008.

[23] M. A. Potter, and K. A. De Jong, "A cooperative coevolutionary approach to function optimization," In *Proceedings of International Conference on Parallel Problem Solving from Nature,* pp. 249-257, 1994.

[24] G. Liu, Y. Li, and G. He, "Design of digital FIR filters using differential evolution algorithm based on reserved genes," In *Proceedings of IEEE congress on evolutionary computation*, pp. 1-7, 2010.

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

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

[27] W. Fang, J. Sun, W. Xu, and J. Liu, "FIR digital filters design based on quantum-behaved particle swarm optimization," In *Proceedings of First International Conference on Innovative Computing, IEEE Information and Control-Volume I (ICICIC'06) on*, vol. 1, pp. 615-619, 2006.

[28] K. Price and R. Storn, "Differential evolution: a simple evolution strategy for fast optimization," *Dr. Dobb's Journal*, vol. 22, no. 4, pp. 18-24, 1997.

[29] A. Sarangi, R. K. Mahapatra, and S. P. Panigrahi, "RETRACTED: DEPSO and PSO-QI in digital filter design," *Expert Systems with Applications,* vol. 38, no. 9, pp. 10966-10973, 2011.

[30] Z. Zhao and H. Gao, “FIR digital filters based on cultural particle swarm optimization,” In *Proceedings of Proc. 2009 International Workshop on Iriformation Security and Application (IWISA 2009), Qingdao, China*, pp. 252-255. 2009.

[31] D. Mandal, R. Kar, and S. P. Ghoshal, “Digital FIR filter design using fitness based hybrid adaptive differential evolution with particle swarm optimization,” *Natural Computing*, vol. 13, no. 1, pp. 55-64, 2014.

[32] J. Zhang, D. Jia, and K. Li, “FIR digital filters design based on Chaotic mutation particle swarm optimization,” In *Proceedings of Audio, Language and Image Processing. ICALIP 2008 IEEE International Conference on*, pp. 418-422, 2008.

[33] Y. Lim, and S. Parker, “Discrete coefficient FIR digital filter design based upon an LMS criteria,” In *Proceedings of IEEE Transactions on Circuits and Systems*, vol. 30, no. 10, pp. 723-739, 1983

[34] N. Karaboga, “Digital IIR filter design using differential evolution algorithm,” *EURASIP Journal on Applied Signal Processing*, pp. 1269-1276, 2005.

[35] 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

[36] G. Liu, Y. Li, and G. He, “Design of digital FIR filters using differential evolution algorithm based on reserved genes.” In *Proceedings of evolutionary computation*, *IEEE congress on,* pp. 1-7, 2010.

[37] N. Karaboga, and B. Cetinkaya, “Performance comparison of genetic and differential evolution algorithms for digital FIR filter design,” In *Proceedings of International Conference on Advances in Information Systems*, pp. 482-488. Springer Berlin Heidelberg, 2004.

[38] R. M. Mersereau, and D. E. Dudgeon, “Two-dimensional digital filtering,” In *Proceedings of the IEEE*, vol. 63, no. 4, pp. 610-623, 1975.

[39] A. K. Qin, and P. N. Suganthan, “Self-adaptive differential evolution algorithm for numerical optimization,” In *Proceedings of 2005 IEEE congress on evolutionary computation on,* vol. 2, pp. 1785-1791, 2005.

[40] A. K. Qin, V. L. Huang, and P. N. Suganthan, “Differential evolution algorithm with strategy adaptation for global numerical optimization,” In *Proceedings of Evolutionary Computation, IEEE transactions on*, vol. 13, no. 2, pp. 398-417, 2009.

Vita Auctoris

NAME:	Wei Zhong
PLACE OF BIRTH:	Yantai, Shandong, China
YEAR OF BIRTH:	1988
EDUCATION:	Harbin Engineering University, Harbin, Heilongjiang, China. Bachelor Degree, Information, and Telecommunication Engineering 2008-2012; Master of Engineering, Signal and Information Processing, 2012-2014
	University of Windsor, Windsor ON, Canada. Master of Applied Science, Electrical and Computer Engineering, 2015-2016.