

# Genetic Algorithm for the Design of Optimal IIR Digital Filters

Ranjit Singh<sup>1</sup>, Sandeep K. Arya<sup>2</sup>

<sup>1</sup>Department of Electronics and Communication Engineering, JMIT Radaur, Yamuna Nagar, India; <sup>2</sup>Department of Electronics and Communication Engineering, GJUS&T, Hisar, India.  
Email: [chauhan@jmit.ac.in](mailto:chauhan@jmit.ac.in)

Received January 2<sup>nd</sup>, 2012; revised February 4<sup>th</sup>, 2012; accepted February 11<sup>th</sup>, 2012

## ABSTRACT

This paper presents the design of Optimal Infinite-Impulse Response (IIR) digital filters using Genetic Algorithm (GA). IIR filter is essentially a digital filter with Recursive responses. Since the error surface of digital IIR filters is generally nonlinear and multimodal, global optimization techniques are required in order to avoid local minima. This paper presents heuristic way for the designing IIR filters. GA is a powerful global optimization algorithm introduced in combinatorial optimization problems. The paper finds the optimum Coefficients of IIR digital filter through GA. Design of Lowpass and High pass IIR digital filter is proposed to provide estimate of transition band. It is found that the calculated values are more optimal than *fsda* tool available for the design of filter in MATLAB. The simulation result of the employed examples shows an improvement on transition band and mean-square-error (MSE). The position of pole-zero is also presented to describe stability and results are compared with Simulated Annealing (SA) method.

**Keywords:** Digital Filter; Infinite-Impulse Response (IIR); Genetic Algorithm (GA); Optimization

## 1. Introduction

Over the last few decades the field of Digital Signal Processing (DSP) has grown to important both theoretically and technologically. In DSP, there are two important types of Systems. The first type of systems performs signal filtering in time domain and hence it is known as Digital filters. The second type of systems provide signal representation frequency domain and are known as Spectrum Analyzer. Digital filtering is one of the most powerful tools of DSP. Digital filters are capable of performance specifications that would, at best, be extremely difficult, if not impossible, to achieve with an analog implementation. In addition, the characteristics of a digital filter can be easily changed under software control. Digital filters are classified either as Finite duration impulse response (FIR) filters or Infinite duration impulse response (IIR) filters, depending on the form of impulse response of the system. In the FIR system, the impulse response sequence is of finite duration, *i.e.*, it has a finite number of non zero terms. Digital infinite-impulse-response (IIR) filters can often provide a much better performance and less computational cost than their equivalent finite-impulse-response (FIR) filters and have become the target of growing interest [1-4]. However, because the error surface of IIR filters is usually nonlinear and multimodal, conventional gradient-based design

methods may easily get stuck in the local minima of error surface [4,5]. Therefore, some researchers have attempted to develop design methods based on modern heuristic optimization algorithms such as genetic algorithm (GA) [6-9], simulated annealing (SA), tabu search (TS) [10] etc.

Analytical or simple iterative methods usually lead to sub-optimal designs. Consequently, there is a need of optimization methods (heuristic type) that can be use to design digital filters that would satisfy prescribed specifications. Goldberg presented a detailed mathematical model of Genetic Algorithm [11]. Benvenuto *et al.* (1992) described the salient features of using a simulated annealing (SA) algorithm in the context of designing digital filters with linear phase digital filter. The algorithm is then applied to the design of FIR filter. The result was not impressive. Moreover, it is computationally very expensive. Ahmadi *et al.* (2003) used genetic algorithm to design 1-D IIR filter with canonical-signed-digit coefficients restricted to low-pass filter. Ahmad and Antoniou (2006) explored FIR filters and equalizers through the use of GA. Consequently GAs requires a large amount of computation. Oliveira *et al.* (2007) presented a new approach for designing linear FIR filters by using nonlinear stochastic global optimization based on simulated annealing techniques. Jung *et al.* (2008) found the design

method of a linear phase finite word length finite-duration impulse response (FIR) filter using simulated annealing. Weise and Tang (2011) evaluated the applicability of genetic programming (GP) for the evolution of distributed algorithms. The basic limitation of all the above methods is that they can mainly be used to design FIR digital filters [11-13]. The drawback of preceding design methods is that the computation time is quite long. To test the optimization procedure, the proposed algorithm is implemented in Matlab and results are found to be very encouraging.

This Paper is organized as follows: In Section 2, IIR digital filter design aspects are discussed. In section 3, Genetic Algorithm (GA) approach is briefly mentioned. The Genetic Algorithm (GA) related to filter design is proposed in Section 4. The simulation results of designed examples used is briefly described in Section 5. The Conclusion and future scope is described in Section 6.

### 2. IIR Filter Design Issues

Digital filters are classified as Recursive and Non-Recursive filters. The response of Recursive or IIR filters is dependent on one or more of its past output. If such filter subjected to an impulse then its output need not necessarily become zero. This indicates that the system is prone to feedback and instability. The Digital filters have various stages in their design [14,15] as shown in **Figure 1**.

Consider the IIR filter with the input-output relationship governed by:

$$y(k) + \sum_{i=1}^M b_i y(k-i) = \sum_{i=0}^L a_i x(k-i) \quad (1)$$

where  $x(k)$  and  $y(k)$  are the filter's input and output, respectively,  $M (\geq L)$  is the filter order. The transfer function of this IIR filter can be written as:

$$H(z) = \frac{A(z)}{B(z)} = \frac{\sum_{i=0}^L a_i z^{-i}}{1 + \sum_{i=1}^M b_i z^{-i}} \quad (2)$$

These parameters  $a_0, a_1, a_2, \dots, a_L, b_1, b_2, \dots, b_M$  appearing in Equation (1) and Equation (2) are called the filter coefficients. These determine the characteristics of the filter.

Hence, the design of this filter can be considered as an optimization problem of cost function  $J(w)$  stated as the following:

$$\min J(w), \quad (3)$$

where  $w = [a_0, a_1, a_2, \dots, a_L, b_1, b_2, \dots, b_M]$  is filter coefficient vector. The aim is to minimize the cost function  $J(w)$  by adjusting  $w$ . The cost function is usually expressed as the time-averaged cost function defined by:

$$J(w) = \frac{1}{N} \sum_{k=1}^N (d(k) - y(k))^2 \quad (4)$$

where  $d(k)$  and  $y(k)$  are the desired and actual responses of the filter, respectively and  $N$  is the number of samples used for the calculation of cost function.

### 3. Genetic Algorithm

Genetic Algorithms (GA) are stochastic search methods that can be used to search for an optimal solution to the evolution function of an optimization problem. Holland proposed genetic algorithms in the early seventies as computer programs that mimic the natural evolutionary process. De Jong extended the GAs to functional optimization and a detailed mathematical model of a GA was presented by Goldberg in 1975. GAs manipulates a *population* of individuals in each *generation* (iteration) where each individual, termed as the *chromosome*, represents one candidate solution to the problem. Within the population, fit individuals survive to reproduce and their genetic materials are recombined to produce new individuals as *offsprings*. The genetic material is modeled by some data structure, most often a finite-length of attributes. As in nature, *selection* provides the necessary driving

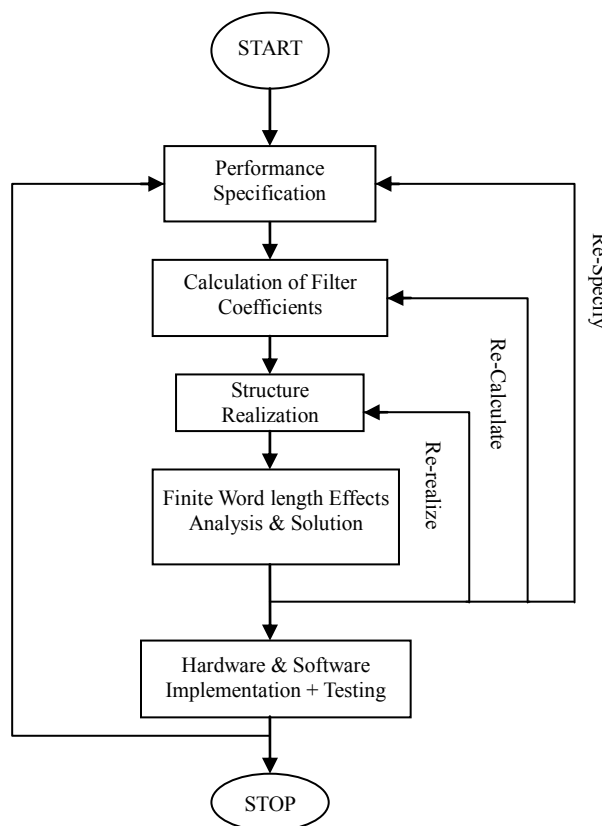


Figure 1. Flow chart of digital filter design.

mechanism for better solutions to survive. Each solution is associated with a *fitness* value that reflects how good it is, compared with other solutions in the population [16]. The recombination process is simulated through a *crossover* mechanism that exchanges portions of data strings between the chromosomes. New genetic material is also introduced through *mutation* that causes random alterations of the strings. The frequency of occurrence of these genetic operations is controlled by certain pre-set probabilities. The selection, crossover, and mutation processes as illustrated in **Figure 2** [17-19] constitute the basic GA cycle or generation, which is repeated until some pre-determined criteria are satisfied. Through this process, successively better and better individuals of the species are generated.

With the increasing computing power offered by advancement in integrated circuit technology, the simulation of evolutionary systems is becoming more and more tractable and GAs are being applied to many real world problems including the design of digital filters.

#### 4. GAs and Filter Design

The error surface of digital infinite-impulse response (IIR) filters is generally nonlinear and multimodal, so global optimization techniques are required in order to avoid local minima. In designing IIR digital filter, the values of  $a_i$  and  $b_i$  must be such that the magnitude response of the filter approximates a desired characteristic while preserving the stability of the designed filter. Relatively little work has been published so far on GAs applied to analogues filters [20]. A number of practical issues are important in analogues filter design [21-25]. One of them is the choice of component values. A conventional form of GA is used to perform the design of IIR digital filter. Genetic Algorithm based Infinite Impulse response (GAIIR) digital filter is implemented by selection, crossover and mutation. The selected random number in range [0.17, 0.76] has been found satisfactory. Fitness is evaluated in the normalized frequency range [0, 1] over a uniform grid of frequency point. Selection is the process of choosing structures for the next generation from the structures in the current generation. In the design of filter the Selection function used is the “*Stochastic Universal*

*Sampling*”, which allocate to each individual a portion of the wheel proportional to the individual’s fitness. Crossover is the process of generating a child from two parents by taking a part from one of the parents and replaces it with the corresponding part from the second parent and vice versa. IIR Filter designing is performed by *double point crossover* between pairs of individuals and returns the current generation after mating. Mutation is a change done on some of the children resulted from the crossover process by flipping the value of one of the bits randomly. The benefit of such operation is to restore the lost genetic values when the population converges too fast. The filter coefficients were encoded in terms of 16 bit binary string with initial Crossover probability and Mutation probability of 0.8 and 0.02 respectively, and the population size of 50 was assumed. The processes as illustrated above constitute the basic pseudo code of GA for IIR shown in **Figure 3**, which is repeated until some pre-determined criteria are satisfied.

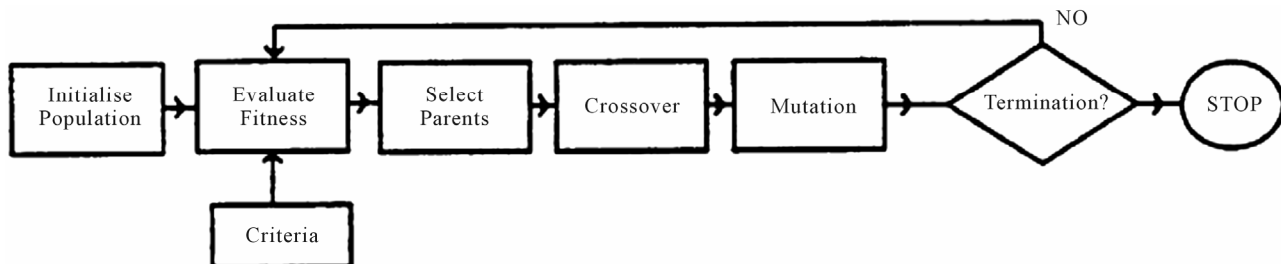
The GAIIR produces filter coefficients that satisfy both magnitude and phase templates. The fitness value of a solution  $i$  in the population is determined by using fitness formula given as:

$$fit(i) = \frac{1}{k + J(w)_i} \quad (5)$$

where  $J(w)$  is the cost function value computed for  $i$  and  $k$  is the number of poles outside the unit circle.

#### 5. Results and Discussion

Simulation studies are carried for well known IIR filters (see Appendix), which has been used by many authors as a “benchmark filter” for comparison purpose [26,27]. The magnitude response of first example is shown in **Figure 4** in which a zoomed in curve for transition band is included. It is observed from the responses that the gain is  $-62.61$  at  $0.9746$  while using GA, which occurs at  $0.9868$  in case of general. The Phase response is shown in **Figure 5** in which it is clearly seen that the response is identical to *fda* tool. In **Figure 6**, we have summarized Pole-Zero behavior of Low pass filter. It can be seen that the poles-zeros location of designed filter falls within unit circle. This shows that the designed filter is also stable



**Figure 2.** Basic genetic algorithm cycle.

```

Procedure GA
Initialization;
while condition not met
while condition not met
    Evaluate fitness
    prune population
    Selection
    Crossover
    Mutation
end while
end procedure
    
```

Figure 3. Pseudo code of genetic algorithm.

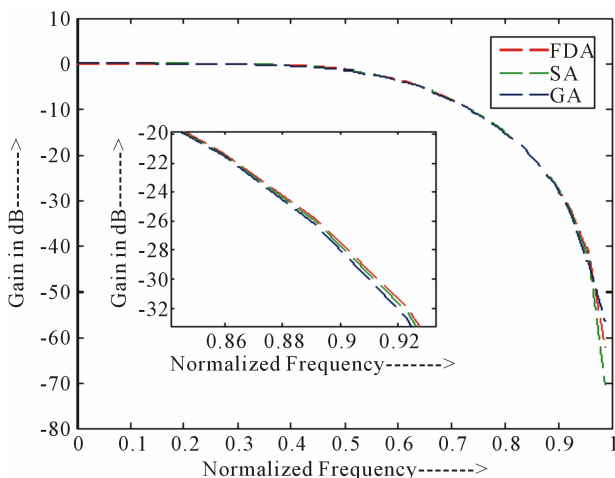


Figure 4. Magnitude response of lowpass filter.

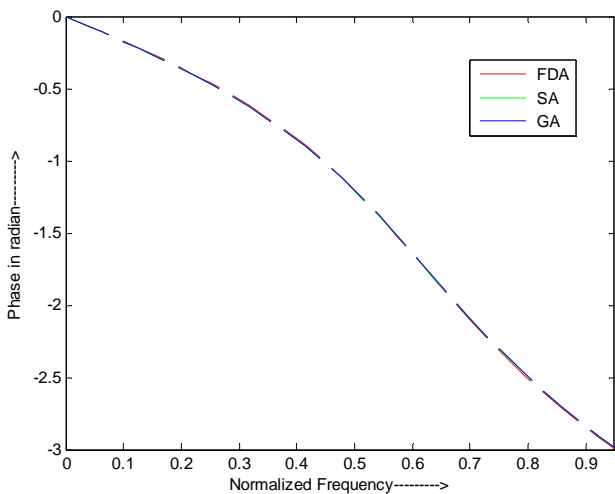


Figure 5. Phase response of lowpass filter.

using proposed method. The resulting coefficients of the low-pass filter are shown in Table 1. A comparison of coefficients is done with the *fda* tool and SA method with minimum mean-square-error (MSE).

The Figure 7 illustrates the magnitude response of second example in which a zoomed in curve for transition band is included. From this, we have observed that

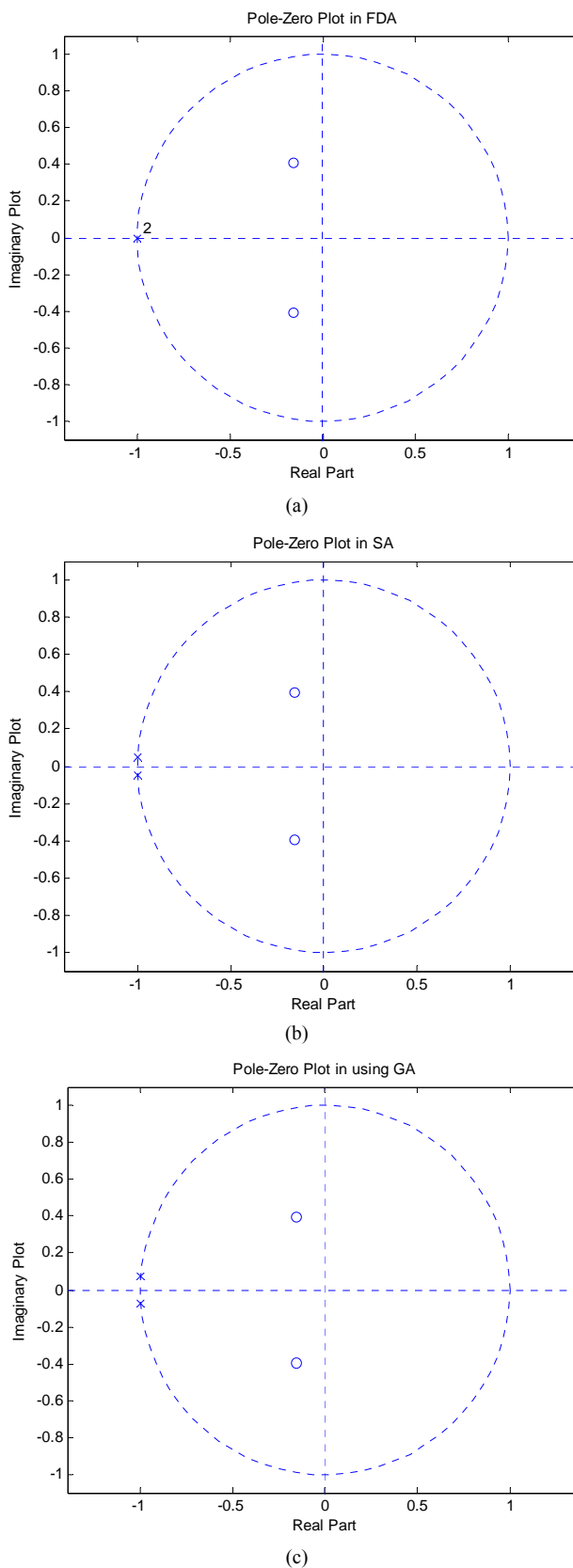
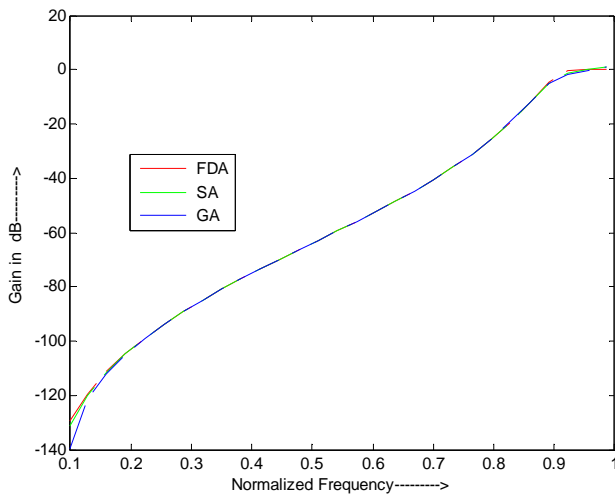


Figure 6. Pole-Zero plot lowpass filter. (a) Using *fda*; (b) Using SA; (c) Using GA.

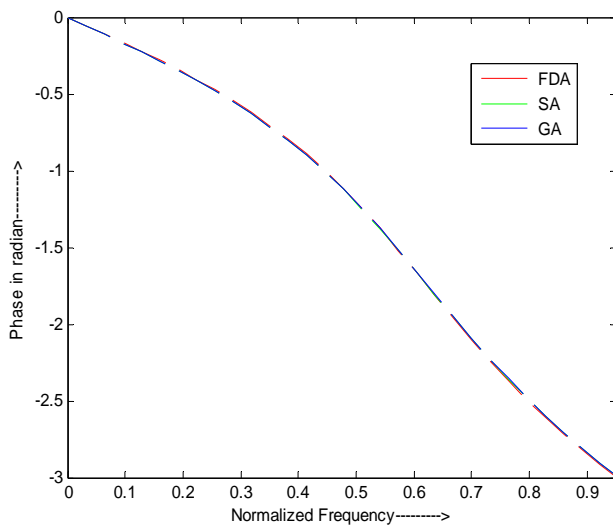
gain of  $-54.41$  occur at  $0.3446$  which changes to  $0.4546$  through GA. The phase response of designed high pass filter is shown in **Figure 8**. **Figure 9** summarized Pole-Zero position of High pass filter using *fda*, SA and GA. It is clearly seen that the poles and zeros placement gives us stable high pass filter. Similarly, **Table 2** gives the coefficients of High-pass filter under same specifications with the evaluation of MSE. The designed example is compared with the traditional method. On comparison, it is found that the proposed algorithm gives optimal coefficients for High pass filter.

**6. Conclusion**

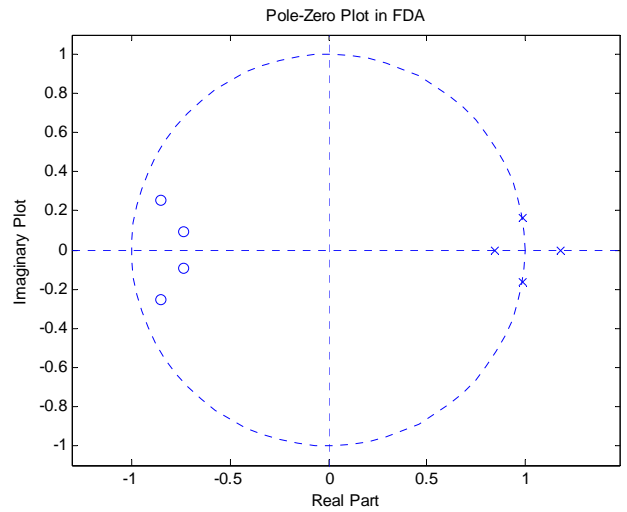
It is concluded that Genetic Algorithm is a global optimization technique for IIR digital filters, and the benefits of GA for designing digital filter have been studied. The simulation results show that GA has better, or at least



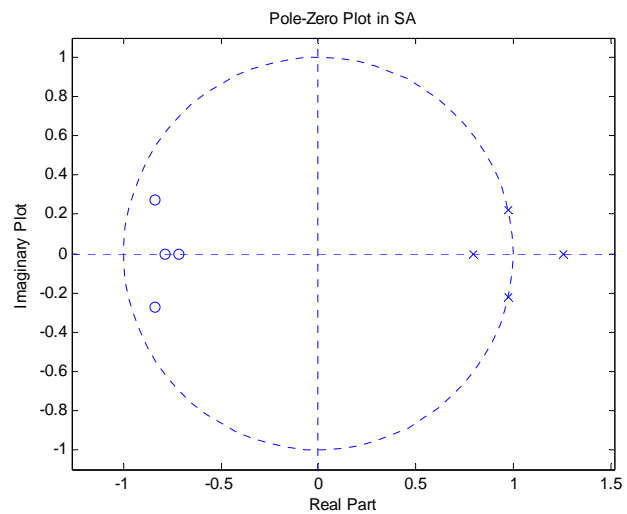
**Figure 7. Magnitude response of highpass filter.**



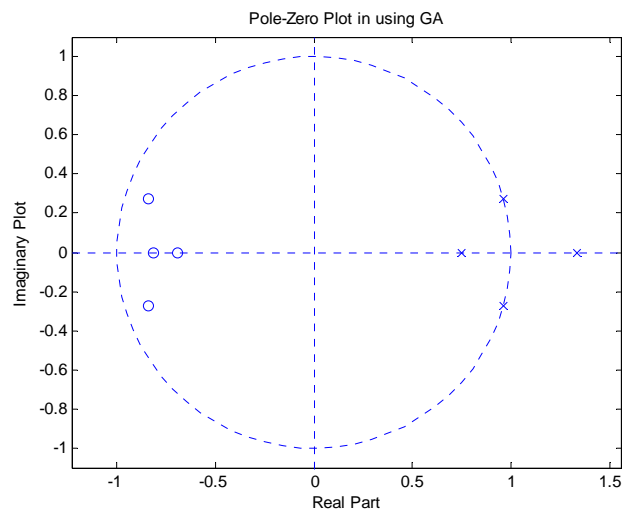
**Figure 8. Phase response of highpass filter.**



(a)



(b)



(c)

**Figure 9. Pole-Zero plot highpass filter. (a) Using *fda*; (b) Using SA; (c) Using GA.**

**Table 1. Coefficients of low-pass filter designed with MSE.**

Name of Method	Order of filter	Coeff. of Numerator	Coeff. of Denominator	MSE
<i>fda</i> tool	2	0.3752, 0.7504, 0.3752	1.0, 0.3120, 0.1888	0.3296
<i>SA</i> method	2	0.3749, 0.7489, 0.3749	1.0, 0.3109, 0.1790	0.3284
Proposed method	2	<b>0.3739, 0.7459, 0.3739</b>	<b>1.0, 0.3116, 0.1789</b>	<b>0.3275</b>

**Table 2. Coefficients of high-pass filter designed with MSE.**

Name of Method	Order of filter	Coeff. of Numerator	Coeff. of Denominator	MSE
<i>fda</i> tool	4	0.0004218, -0.0016876, 0.0025313, -0.0016876, 0.0004218	1.000, 3.1778, 3.8549, 2.1073, 0.4370	3.0590
<i>SA</i> method	4	0.0004215, -0.0016876, 0.0025311, -0.0016876, 0.0004215	1.000, 3.1773, 3.8541, 2.1071, 0.4363	3.0580
Proposed method	4	<b>0.0004213, -0.0016877, 0.0025301, -0.0016877, 0.0004213</b>	<b>1.000, 3.1769, 3.8539, 2.1069, 0.4359</b>	<b>3.0574</b>

equivalent, global search ability and convergence speed than others. The examples demonstrate the optimization and versatility of the proposed approach. The performance of proposed method has been compared with *fda* tool and over Simulated Annealing (SA) method. It is observed that the value of MSE calculated using GA is 0.3275 and 3.0574 in case of lowpass and highpass respectively, is less as compared to other two heuristic approaches. Thus it is believed that the proposed algorithm is capable of quick and high performance. The proposed method can be extended to arbitrary magnitude response specifications and multiband. Further, the other Evolutionary algorithm can be discussed for the design of IIR filters.

## REFERENCES

- [1] V. K. Ingle and J. G. Proakis, "Digital Signal Processing Using MATLAB," Thomson Books, New Delhi, 2004.
- [2] J. G. Proakis and D. G. Manolakis, "Digital Signal Processing: Principles, Algorithms, and Applications," 4th Edition, Pearson Education, Inc., New Delhi, 2007.
- [3] P. Tarasewich and P. R. McMullen, "Swarm Intelligence," *Communication of the ACM*, Vol. 45, No. 8, 2002, pp. 62-67.
- [4] L. Y. Cao, "Practical Issues in Implementing a Single-Pole Low-Pass IIR Filter," *IEEE Signal Processing Magazine*, November 2010, pp. 114-117.
- [5] J. Skaf and P. B. Stephen, "Filter Design with Low Complexity Coefficients," *IEEE Transactions on Signal processing*, Vol. 56, No. 7, 2008, pp. 3162-3170. [doi:10.1109/TSP.2008.919386](https://doi.org/10.1109/TSP.2008.919386)
- [6] R. J. Vaccaro and B. F. Harrison, "Optimal Matrix-Filter Design," *IEEE Transactions on Signal processing*, Vol. 44, No. 3, 1996, pp. 705-710. [doi:10.1109/78.489044](https://doi.org/10.1109/78.489044)
- [7] X. Zhang and H. Iwakura, "Design of IIR Digital Filters based on Eigen Value Problem," *IEEE Transactions on Signal processing*, Vol. 44, No. 6, 1996, pp. 1325-1319. [doi:10.1109/78.506600](https://doi.org/10.1109/78.506600)
- [8] F. Argenti and E. Del Re, "Design of IIR Eigen Filters in the Frequency Domain," *IEEE Transactions on Signal processing*, Vol. 46, No. 6, 1998, pp. 1694-1700. [doi:10.1109/78.678495](https://doi.org/10.1109/78.678495)
- [9] X. Yao, Y. Liu and G. M. Lin, "Evolutionary Programming Made Faster," *IEEE Transactions on Evolutionary Computation*, Vol. 3, No. 2, 1999, pp. 83-102.
- [10] N. Benvenuto and M. Marchesi, "Applications of Simulated Annealing for the Design of Digital Filters," *IEEE Transactions on Signal Processing*, Vol. 40, No. 2, 1992, pp. 323-331. [doi:10.1109/78.124942](https://doi.org/10.1109/78.124942)
- [11] K. S. Tang, K. F. Man and S. Kwong, "Design and Optimization of Digital Filter Structure Using Genetic Algorithm," *IEEE Transactions on Industrial Electronics*, Vol. 45, No. 3, 1998, pp. 481-489. [doi:10.1109/41.679006](https://doi.org/10.1109/41.679006)
- [12] K. D. Abdesselam, "Design of Stable, Causal, Perfect Reconstruction, IIR Uniform DFT Filters," *IEEE Transactions on Signal Processing*, Vol. 48, No. 4, 2000, pp. 1110-1117. [doi:10.1109/78.827544](https://doi.org/10.1109/78.827544)
- [13] C. C. Tseng and S. C. Pei, "Stable IIR Notch Filter Design with Optimal Pole Placement," *IEEE Transactions on Signal Processing*, Vol. 49, No. 11, 2001, pp. 2673-2681. [doi:10.1109/78.960414](https://doi.org/10.1109/78.960414)
- [14] L. Liang, M. Ahmadi, M. Ahmed and K. Wallus, "Design of Canonical Signed Digital Filters Using Genetic Algorithms," *IEEE Transaction on Signal Processing*, Vol. 3, No. 1, 2003, pp. 2043-2047.
- [15] S. U. Ahmad and A. Antoniou, "Design of Digital Filters Using Genetic Algorithms," *IEEE Transaction on Signal Processing*, Vol.1, No. 1, 2006, pp. 1-9.
- [16] J. E. Cousseau, S. Werner and P. D. Donate, "Factorized All-Pass Based IIR Adaptive Notch Filters," *IEEE Transactions on Signal Processing*, Vol. 55, No. 11, 2007, pp. 5225-5236.
- [17] B. W. Jung, H. J. Yang and J. Chun, "Finite Word length Digital Filter Design Using Simulated Annealing," *IEEE Transactions on Signal Processing*, Vol. 15, No. 5, 2008,



- pp. 546- 550.
- [18] C. H. Dai, W. R. Chen and Y. F. Zhu, "Seeker Optimization Algorithm for Digital IIR Filter Design," *IEEE Transaction on Evolutionary Computation*, Vol. 57, No. 5, 2010, pp. 1710-1718.
- [19] D. E. Goldberg, "Genetic Algorithm in Search, Optimization and Machine Learning, Pearson Education," Low Price Edition, Delhi, 2005.
- [20] W. X. Zheng, "Adaptive Filter Design Subject to Output Envelop Constraints and Bounded Input Noise," *IEEE Transaction on Circuit & Systems-II Analog & Digital Signal Processing*, Vol. 50, No. 12, 2003, pp. 1023-1027.
- [21] Y. R. Zhou and J. He, "A Runtime Analysis of Evolutionary Algorithms for Constrained Optimization Problems," *IEEE Transactions on Evolutionary Computation*, Vol. 11, No. 5, 2007, pp. 608-620.  
[doi:10.1109/TEVC.2006.888929](https://doi.org/10.1109/TEVC.2006.888929)
- [22] N. Benvenuto, M. Marchesi and A. Uncini, "Applications of Simulate Annealing for Design of Digital Filter," *IEEE Transactions on Signal Processing*, Vol. 40, No. 2, 1992, pp. 323-332. [doi:10.1109/78.124942](https://doi.org/10.1109/78.124942)
- [23] J. Skaf and P. Boyd Stephen, "Filter Design with Low Complexity Coefficients," *IEEE Transactions on Signal Processing*, Vol. 56, No. 7, 2008, pp. 3162-3170.  
[doi:10.1109/TSP.2008.919386](https://doi.org/10.1109/TSP.2008.919386)
- [24] T. Weise and K. Tang, "Evolving Distributed Algorithms with Genetic Programming" *IEEE Transactions on Evolutionary Computation*, 2011, pp. 1-24.
- [25] A. Antoniou, "Digital Filters Analysis, Design and Application," Tata Mcgraw-Hill Edition, New Delhi, 2005.
- [26] R. S. Chauhan and S. K. Arya, "An Optimal Design of FIR Digital Filter Using Genetic Algorithm," *Lecture Notes in Computer Science (LNCS) Springer-Verlag Berlin Heidelberg*, Vol. 168, No. 1, 2011, pp. 51-56.
- [27] H. Ali, A. Doucet and D. I. Amshah, "GSR: A New Genetic Algorithm for Improving Source and Channel Estimates," *IEEE Transactions on Circuits and Systems-I*, Vol. 54, No. 5, 2007, pp. 1088-1098.  
[doi:10.1109/TCSI.2007.893507](https://doi.org/10.1109/TCSI.2007.893507)

## Appendix

Example 1: In the first example, design a low-pass filter with following specifications: Pass/Stop band ripples 1 dB/15 dB, and band edges 200 Hz/400 Hz and a sampling frequency of 1000 Hz.

Example 2: This example is taken for design of a high-pass filter with following specifications: Pass/Stop band ripples 1 dB/75 dB, and band edges 700 Hz/300 Hz and a sampling frequency of 1500 Hz.