

# OPTIMAL COMPONENT VALUE SELECTION FOR ANALOG ACTIVE FILTER USING DIFFERENTIAL EVOLUTION

<sup>a</sup>Vasundhara, <sup>a</sup>Durbadal MANDAL, <sup>a</sup>Rajib KAR, <sup>b</sup>Sakti PRASAD GHOSHAL

<sup>a</sup>Department of Electronics and Communication Engineering,

<sup>b</sup>Department of Electrical Engineering,

National Institute of Technology, Durgapur, West Bengal, India

Tel.: +91 343 2754390; Fax: +91 343 2547375;

E-mail address: [vdhara2@gmail.com](mailto:vdhara2@gmail.com), [durbadal.bittu@gmail.com](mailto:durbadal.bittu@gmail.com), [rajibkarece@gmail.com](mailto:rajibkarece@gmail.com),  
[spghoshalnitdgp@gmail.com](mailto:spghoshalnitdgp@gmail.com)

*Abstract-* This paper presents an efficient approach of designing analog active filter by selecting its component value with an optimization method known as differential evolution. Differential evolution (DE) is one of the very fast and robust evolutionary algorithms, which has shown to have superior performance for continuous global optimization and uses differential information to guide its search direction. Differential Evolution serves the dual task of efficiently optimizing the component values as well as minimizing the total design error of a 4th order Butterworth low pass active filter. The component values of the Butterworth active filter are designed in such a way so that they are E12 series compatible. Differential Evolution proves itself to be a very good optimizing tool for selecting the components of the analog active filter. The simulation results prove the efficiency of using DE for the design of analog active filter by optimizing the component values as well as design error simultaneously.

Keywords: Analog active filter; Butterworth filter; Differential evolution; Optimization Tool

## I. INTRODUCTION

Filters of some sort are essential to the operation of most electronic circuits. Electronic circuit design must have the ability to develop filter circuits capable of meeting a given set of specifications. A filter can be viewed as a network that alters the amplitude or phase response of any signal with respect to frequency. In ideal case, a filter will not add any new frequency component to any signal neither it will change any existing frequency component of the signal; rather it will only alter the relative amplitudes or phase responses of various frequency components. Filters are often used in electronic systems to emphasize signals in certain frequency ranges and reject signals in other frequency ranges. Such a filter has a gain which is dependent on signal. An analog active filter is a type of electronic filter which uses active components like amplifiers, transistors in addition

to resistors and capacitors. Amplifiers used in the design of an active filter can improve the performance and predictability of the filter without using the expensive inductors [1]. An amplifier prevents the load impedance of the following stage from affecting the characteristics of the filter. An active filter can have complex poles and zeros without using a bulky or expensive inductor. The shape of the response, the Q (quality factor), and the tuned frequency can often be set with inexpensive variable resistors. In some active filter circuits, one parameter can be adjusted without affecting the others [1].

A passive filter consists of only passive elements which include resistors, inductors and capacitors. There are many inherent advantages of active filters over passive filters like no insertion loss, since the amplifier can provide gain as well. Active filter components are more economical than inductors which form the major part of passive filters. Active filters are easily tuned and adjusted over a wide range without altering the desired response. Active filters have good isolation due to their high input impedance and low output impedance. Analog filters have higher dynamic range than digital filters in both ranges of amplitude and frequency.

Analog filters find wide applications in high precision large scale integrated circuits. Therefore it becomes quite necessary to carefully select the parameters of the filter as it can affect the stability of the system and the noise restraining capability. If the parameters of the filters are selected manually by modification, computation and debugging, then it decrease the precision and accuracy of the filter [2-4]. So, parameter selection of the filter has to be done with great precision. In recent times, this area has attracted a lot of attention for researchers. The values of the passive components used in the filter can be chosen equal to each other. Though this simplifies the procedure but limits the flexibility of the design [4]. In order to make the design more reliable and flexible, the passive component values are chosen from a series known as E12 series. In the E12 series each succeeding resistor falls within the  $\pm 10\%$  of the previous value. The commonly used values in the E12 series are  $1\Omega$ ,  $1.2\Omega$ ,

1.5Ω, 1.8Ω, 2.2Ω, 2.7Ω, 3.3Ω, 3.9Ω, 4.7Ω, 5.6Ω, 6.8Ω, 8.2Ω (1 ohm to 8.2 ohm). If the component values of the active filter are chosen either from this series or other possible preferred values, then it reduces computational cost as well as complexity. Some of the passive components are directly chosen from the series and others are rounded off to get the nearest preferred values. In an attempt to round off to the nearest preferred values, it may so happen that the cost criteria of design of the filter is not satisfied. Therefore, it becomes a challenging task to launch an exhaustive search method on all possible combinations of preferred values to obtain an optimized design.

A few works have already been reported in this field of optimizing the components of the active filters using evolutionary optimization techniques like genetic algorithm [5]. Horrocks *et al.* [6] used the genetic algorithm for the design of analog circuits considering their parasitic effects. An op-amp is also designed using genetic programming in [7]. Genetic algorithm provides a basis for automatic synthesis of analog electronic networks [8]. An automatic circuit for optimisation of analog circuits is presented in [9]. Analog electronic networks are also synthesised using genetic algorithm [10]. In [11], genetic algorithm is presented as a better optimizing tool used for the automatic synthesis of analog circuits. Genetic algorithm is also used for the design of CMOS op-amp [12]. Many sequential and combinational circuits are also designed using genetic algorithms [13]. Different evolutionary algorithms like immune algorithm [14] and Tabu search [15] are also used for the design of analog active filters. Clonal selection method is also employed for the component selection of the Butterworth active LP filter in [16]. In [17], a comparison has been made among various optimization methods used for the design of electronic filter. Yildirim *et al.* [18] have proposed the use of Particle Swarm Optimization [19] method for the design of active analog filter. In this paper another evolutionary optimization method known as Differential Evolution (DE) is used for better optimization of the passive components of the active analog filter. DE is a stochastic, population-based optimisation algorithm introduced by Storn and Price in 1996 [20]. The rest of the paper is arranged as follows. In section II, the Butterworth filter design problem is formulated. Section III briefly discusses about the DE algorithm. Section IV describes the DE based active filter design approach. Section V shows the simulation results obtained using DE algorithm. Finally, section VI concludes the paper.

## II. BUTTERWORTH ACTIVE FILTER

The Butterworth filter is a type of signal processing filter designed to have as flat a frequency

response as possible in the pass band and it is also termed as a maximally flat magnitude filter. The transient response of a Butterworth filter to a pulse input shows moderate overshoot and ringing [9]. The Butterworth implementation ensures flat response (maximally flat) in the pass band and an adequate roll-off. This type of filter is a good ‘all rounder’, simple to realize and is good for applications such as audio processing. In this paper, a low pass (LP) Butterworth active filter is designed with a cut off frequency of 10krad/s ( $w_c$ ) and stop band extends from 20k rad/s to  $\infty$  ( $w_s$ ), where ( $w_s$ ) is the stop band frequency. A fourth order Butterworth LP filter can be designed by cascading two second order blocks [16]. This fourth order Butterworth filter behaves like a voltage controlled voltage source.

The transfer function of the fourth order Butterworth filter can be obtained by cascading two second order filters.

$$H(S) = \frac{W_{c1}^2 \cdot W_{c2}^2}{\left( (S^2 + (W_{c1}/Q_1)S + W_{c1}^2) \times (S^2 + (W_{c2}/Q_2)S + W_{c2}^2) \right)} \quad (1)$$

The circuit of a fourth order Butterworth LP filter is shown below:

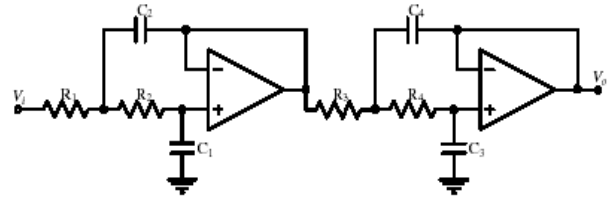


Figure1. Fourth order Butterworth LP filter.

The cut-off frequency and the quality factor for both the cascaded filters can be written as:

$$W_{c1} = \frac{1}{\sqrt{R_1 R_2 C_1 C_2}} \quad (2)$$

$$W_{c2} = \frac{1}{\sqrt{R_3 R_4 C_3 C_4}} \quad (3)$$

$$Q_1 = \frac{\sqrt{R_1 R_2 C_1 C_2}}{R_1 C_1 + R_2 C_1} \quad (4)$$

$$Q_2 = \frac{\sqrt{R_3 R_4 C_3 C_4}}{R_3 C_3 + R_4 C_3} \quad (5)$$

The transfer function of a fourth order Butterworth LP filter, according to Figure 1, can be written as:

$$H(S) = H_1(S) \cdot H_2(S) \quad (6)$$

where

$$H_1(S) = \frac{1}{(S^2 R_1 R_2 C_1 C_2 + S(R_1 C_1 + R_2 C_1) + 1)}$$

$$\text{and } H_2(S) = \frac{1}{(S^2 R_3 R_4 C_3 C_4 + S(R_3 C_3 + R_4 C_3) + 1)}$$

where  $W_{c1}$  is the cut-off frequency of the first second order filter used in the design of the Butterworth filter;  $W_{c2}$  is the cut-off frequency of the second filter used for cascading. The cut-off frequency of each second order filters used for cascading is 10krad/sec. Quality factor or Q factor is a dimensionless parameter that describes how under-damped an oscillator or resonator is, or equivalently, characterizes a resonator's bandwidth relative to its centre frequency. Higher Quality factor states relatively lower rate of energy loss as compared to its stored energy, i.e., oscillations will die out slowly. So, quality factor is an important parameter of any filter. The quality factors  $Q_1$  and  $Q_2$  are 1/0.7654 and 1/1.8478, respectively, which are determined from the table of the LP second orders. In traditional method of filter design the values of all the resistors are taken as 1 ohm each and the cut-off frequency and the quality factor are set and then the values of all the capacitors are determined from (4) and (5).

But in this paper, component values of the filter are determined via evolutionary optimization method and all those component values are E12 series compatible. So in order to make the resistor values to lie within a certain range, their values are multiplied by a certain factor and to make the capacitor values series compatible, all the capacitor values are divided by the same factor.

The main aim of this paper is to carefully select the components of the filter and to minimise the cost function simultaneously. The cost function is created by taking the components of the active filter only [16]. The total cost function, which is also the design error, is the summation of cut-off frequency deviation ( $\Delta W$ ) and quality factor deviation ( $\Delta Q$ ) as given in (11).

$$\Delta W = \frac{(|W_{c1} - W_c| + |W_{c2} - W_c|)}{W_c} \quad (7)$$

$$\Delta Q = \left( \left| Q_1 - \frac{1}{0.7654} \right| + \left| Q_2 - \frac{1}{1.8478} \right| \right) \quad (8)$$

$W_c = 10\text{k rad/sec}$

In terms of the components of the filter the frequency deviation parameter can be written as:

$$\Delta W = \frac{\left( \left| \frac{1}{\sqrt{R_1 R_2 C_1 C_2}} - W_c \right| + \left| \frac{1}{\sqrt{R_3 R_4 C_3 C_4}} - W_c \right| \right)}{W_c} \quad (9)$$

The quality factor deviation parameter can be written as:

$$\Delta Q = \frac{\left( \left| \frac{\sqrt{R_1 R_2 C_1 C_2}}{R_1 C_1 + R_2 C_1} - \frac{1}{0.7654} \right| + \left| \frac{\sqrt{R_3 R_4 C_3 C_4}}{R_3 C_3 + R_4 C_3} - \frac{1}{1.8478} \right| \right)}{1} \quad (10)$$

The total cost function can be written as (11)

$$CF = (0.5 \Delta W + 0.5 \Delta Q) \quad (11)$$

The total cost function is to be minimised by the DE algorithm and while doing so, the component values which give the minimum error are selected and if required are rounded off to the nearest preferred values so as to make them E12 series compatible.

### III. DIFFERENTIAL EVOLUTION

The concept of DE was first proposed by Storn and Price in 1995 [20]. The crucial idea behind DE algorithm is a scheme for generating trial parameter vectors and adds the weighted difference between two population vectors to a third one. Like any other evolutionary algorithm, DE algorithm aims at evolving a population of  $N_p$ , D-dimensional parameter vectors, so-called individuals, which encode the candidate solutions, i.e.,

$$\vec{x}_{i,g} = \{x_{1,i,g}, x_{2,i,g}, \dots, x_{D,i,g}\} \quad (12)$$

where  $i = 1, 2, 3, \dots, N_p$ . The initial population (at  $g=0$ ) should cover the entire search space as much as possible by uniformly randomizing individuals within the search constrained by the prescribed minimum and maximum parameter bounds:

$$\vec{x}_{\min} = \{x_{1,\min}, \dots, x_{D,\min}\} \text{ and } \vec{x}_{\max} = \{x_{1,\max}, \dots, x_{D,\max}\} \quad (13)$$

For example, the initial value of the  $j$ th parameter of the  $i$ th vector is

$$x_{j,i,0} = x_{j,\min} + rand(0,1) * (x_{j,\max} - x_{j,\min}) \quad (14)$$

The random number generator,  $rand(0,1)$ , returns a uniformly distributed random number in the range [0,1]. After initialization, DE enters a loop of evolutionary operations: mutation, crossover, and selection.

a) *Mutation:*

Once initialized, DE mutates and recombines the population to produce new population. For each trial vector  $x_{i,g}$  at generation  $g$ , its associated mutant vector  $\vec{v}_{i,g} = \{v_{1,i,g}, v_{2,i,g}, \dots, v_{D,i,g}\}$  can be generated via certain mutation strategy. Five most frequently used mutation strategies in the DE codes are listed as follows:

$$"DE / rand / 1": \vec{v}_{i,g} = \vec{x}_{r_1,g} + F(\vec{x}_{r_2,g} - \vec{x}_{r_3,g}) \quad (15)$$

$$"DE / best / 1": \vec{v}_{i,g} = \vec{x}_{best,g} + F(\vec{x}_{r_1,g} - \vec{x}_{r_2,g}) \quad (16)$$

$$"DE / rand - to - best / 1": \vec{v}_{i,g} = \vec{x}_{i,g} + F(\vec{x}_{best,g} - \vec{x}_{i,g}) + F(\vec{x}_{r_1,g} - \vec{x}_{r_2,g}) \quad (17)$$

$$"DE / best / 2": \vec{v}_{i,g} = \vec{x}_{best,g} + F(\vec{x}_{r_1,g} - \vec{x}_{r_2,g}) + F(\vec{x}_{r_3,g} - \vec{x}_{r_4,g}) \quad (18)$$

$$"DE / rand / 2": \vec{v}_{i,g} = \vec{x}_{r_1,g} + F(\vec{x}_{r_2,g} - \vec{x}_{r_3,g}) + F(\vec{x}_{r_4,g} - \vec{x}_{r_5,g}) \quad (19)$$

The indices  $r_1, r_2, r_3, r_4, r_5$  are mutually exclusive integers randomly chosen from the range  $[1, N_p]$ , and all are different from the base index  $i$ . These indices are randomly generated once for each mutant vector. The scaling factor  $F$  is a positive control parameter for scaling the difference vector.  $x_{best,g}$  is the best individual vector with the best fitness value in the population at generation 'g'.

b) *Crossover*

To complement the differential mutation search strategy, crossover operation is applied to increase the potential diversity of the population. The mutant vector  $v_{i,g}$  exchanges its components with the target vector  $x_{i,g}$  to generate a trial vector:

$$\vec{u}_{i,g} = \{u_{1,i,g}, u_{2,i,g}, \dots, u_{D,i,g}\}$$

In the basic version, DE employs the binomial (uniform) crossover defined as

$$u_{j,i,g} = \begin{cases} v_{j,i,g} & \text{if } (rand_{i,j}(0,1) \leq C_r \text{ or } j = j_{rand}) \\ x_{j,i,g} & \text{otherwise} \end{cases} \quad (20)$$

where  $j=1, 2, \dots, D$ .

The crossover rate  $C_r$  is user-specified constant within the range  $[0, 1]$ , which controls the fraction of parameter values copied from the mutant vector. The parameter  $j_{rand}$  is a randomly chosen integer in the range  $[1, D]$ . The binomial crossover operator copies the  $j^{th}$  parameter of the

mutant vector  $\vec{v}_{i,g}$  to the corresponding element in the trial vector  $\vec{u}_{i,g}$  if  $rand_{i,j}(0,1) \leq C_r$  or  $j = j_{rand}$ . Otherwise, it is copied from the corresponding target vector  $x_{i,g}$ .

c) *Selection*

To keep the population size constant over subsequent generations, the next step of the algorithm calls for the selection to determine whether the target or the trial vector survives to the next generation i.e., at  $g=g+1$ . The selection operation is described as (24).

$$\vec{x}_{i,g+1} = \begin{cases} \vec{u}_{i,g} & \text{if } f(\vec{u}_{i,g}) \leq f(\vec{x}_{i,g}) \\ \vec{x}_{i,g} & \text{otherwise} \end{cases} \quad (21)$$

where  $f(x)$  is the objective / error fitness function to be minimized. So, if the new vector yields an equal or lower value of the objective function, it replaces the corresponding target vector in the next generation; otherwise the target is retained in the population. Hence the population either gets better (with respect to the minimization of the objective function) or remains the same in fitness status, but never deteriorates.

Proper selection of control parameters is very important for algorithm's success and performance. The optimal control parameters are problem-specific. Therefore, the set of control parameters that best fit each problem have to be chosen carefully. Values of  $F$  lower than 0.5 may result in premature convergence, while values greater than 1 tend to slow down the convergence speed. Large populations help in maintaining diverse individuals, but also slow down convergence speed. In order to avoid premature convergence,  $F$  or  $N_p$  should be increased or  $C_r$  should be decreased. Larger values of  $F$  result in larger perturbations and better probabilities to escape from local optima, while lower  $C_r$  preserves more diversity in the population, thus avoiding local optima.

#### IV. DE BASED ACTIVE FILTER DESIGN

The initial population is formed by the values for the resistors and capacitors lying in the range of E12 series. Each component used in the design is chosen to take value in the range of  $10^3$  to  $10^6 \Omega$  for a resistor and  $10^{-9}$  to  $10^{-6} F$  for a capacitor.

If the values of the resistors and capacitors lie outside this range, then they must be discarded as now they would not be compatible with the E12 series.

Each row of the population matrix can be initialised as a vector:

$$[p, q, r, s, t, u, v, w, a, b, c, d, e, f, g, h] \quad (22)$$

From these elements of each row vector of the population matrix, the values of the resistors and capacitors can be calculated as (23).

$$R_1 = p * 100 * (10^a) \Omega \quad R_2 = q * 100 * (10^b) \Omega$$

$$R_3=r*100*(10^c) \Omega \quad R_4=s*100*(10^d) \Omega$$

$$C_1=t*100*(10^e) \text{ pF} \quad C_2=u*100*(10^f) \text{ pF}$$

$$C_3=v*100*(10^g) \text{ pF} \quad C_4=w*100*(10^h) \text{ pF} \quad (23)$$

After each and every iteration the elements of the population matrix should be checked whether these are E12 series compatible or not by checking the following criteria:

$$0.1 < p, q, r, s, t, u, v, w < 0.82 \quad \text{and} \quad 2 < a, b, c, d, e, f, g, h < 4; \quad (24)$$

So, by following all the above steps properly and also rounding off to the nearest preferred values, it can be ensured that after optimization, all the components values will be lying within the E12 series range

## V. SIMULATION RESULTS

This section presents the simulation results. The algorithm was iterated many times for getting the desired results. DE is used as the optimization tool for carrying out successive iterations for getting the desired result.

Table 1 shows the comparison of the component values and the cost functions obtained by conventional, other evolutionary optimization methods and the DE algorithm. From Table 1 it is observed that the design error obtained by DE algorithm is the least as compared to all other evolutionary and conventional methods. The design error or the cost function obtained by the DE algorithm is  $1.7814e-004$  as compared to  $0.0088$  obtained by PSO for the ideal case. If the component values obtained by the DE method do not fit into the E12 series range, then they are rounded off to the nearest preferred values. When the component values are rounded off to the nearest value for the E12 series compatibility, then also the value of cost function or design error obtained by DE is less than that of PSO. For the rounding case the design error of DE is  $0.0052$  as compared to  $0.0076$  for PSO,  $0.00789$  obtained by CSA,

$0.01817$  by GA and  $0.0277$  obtained by TS. From Table 1, it is observed that the nearest preferred values of the resistors and capacitors obtained by DE can be realised as  $R_1$  is the series combination of  $1.5 \text{ K}\Omega$  and  $100 \Omega$ ,  $R_3$  is the series combination of  $1 \text{ K}\Omega$  and  $180 \Omega$ .  $C_1$  is the parallel combination of  $27 \text{ nF}$  and  $2.2 \text{ nF}$ ,  $C_2$  is obtained from  $180 \text{ nF}$  and  $33 \text{ nF}$ ,  $C_3$  is the parallel combination of  $82 \text{ nF}$  and  $2.7 \text{ nF}$  and  $C_4$  is the parallel combination of  $82 \text{ nF}$  and  $18 \text{ nF}$ . Thus it is observed from Table 1 that DE proves itself to be superior to other evolutionary as well as conventional algorithms for analog active filter design.

Figure 2 shows the convergence profile obtained for the optimisation of the component values by DE. The Design Error values are plotted against the number of iteration cycles to get the convergence profiles for the optimization technique. The convergence profile shows that DE converges in almost 50 iterations. It is observed from Figure 2 that DE converges to a much lower design error or function cost in very less number of iterations also and hence proves itself to be very economical in terms of convergence speed and the quality of the solution for analog active filter design.

The amplitude response of the Butterworth LP filter designed by DE as realised with all E12 compatible values of the resistors and capacitors is shown in Figure 3. The design of the Butterworth filter as realised with E24 and E96 compatible values of resistors and capacitors is simulated with the Spice simulator using LM380 opamp model. As shown in Figure 3, the Butterworth filter designed by DE provides a maximally flat response in the pass band and a cut-off frequency of  $10.03104 \text{ krad/sec}$ .

## VI. CONCLUSION

In this paper a fourth order analog Butterworth low pass active filter is designed by using Differential Evolution. Differential Evolution proves itself to be a very efficient optimization tool for optimizing the component values of the Butterworth filter. The design error also comes out to be very less in the case of Differential Evolution as compared to other methods and the component values are also E12 series compatible. Thus Differential Evolution proves itself to be an efficient optimization tool for optimizing the components of analog active filter and hence achieving very less value of design error as well.

Table 1 Comparison of the component values and the cost functions among DE and other algorithms

Parameters	Conventional[16]		TS[16]	GA[16]	CSA[16]	PSO[18]		DE	
	Ideal	Nearest				Ideal	Nearest	Ideal	Nearest
R <sub>1</sub>	1KΩ	1KΩ	27 KΩ	4.7 KΩ	4.7 KΩ	4.431 KΩ	4.58 KΩ	<b>1.6093 KΩ</b>	<b>1.6 KΩ</b>
R <sub>2</sub>	1KΩ	1KΩ	270Ω	1.8 KΩ	4.7 KΩ	4.842 KΩ	4.7 KΩ	<b>1.009 KΩ</b>	<b>1 KΩ</b>
R <sub>3</sub>	1KΩ	1KΩ	220 KΩ	100 KΩ	270 Ω	1.09 KΩ	1.1 KΩ	<b>1.1767 KΩ</b>	<b>1.18 KΩ</b>
R <sub>4</sub>	1KΩ	1KΩ	820Ω	4.7 KΩ	27 KΩ	1.023 KΩ	1 KΩ	<b>1.0035 KΩ</b>	<b>1 KΩ</b>
C <sub>1</sub>	38.27nF	39 nF	2.7 nF	12 nF	8.2 nF	8.25 nF	8.2 nF	<b>29.233 nF</b>	<b>29.2 nF</b>
C <sub>2</sub>	26.13 nF	0.27 μF	0.47 μF	0.1 μF	56 nF	56.444 nF	56 nF	<b>210.68 nF</b>	<b>213 nF</b>
C <sub>3</sub>	92.39 nF	0.1 μF	82 nF	1.8 nF	6.8 nF	87.635 nF	87.6 nF	<b>84.697 nF</b>	<b>84.7 nF</b>
C <sub>4</sub>	0.2613μF	0.1 μF	0.68 nF	12 nF	0.2 μF	102.33 nF	102.2 nF	<b>99.986 nF</b>	<b>100 nF</b>
Design Error/Cost Function	0	0.03788	0.02777	0.01817	0.00789	0.00088	0.0076	<b>1.7814e-004</b>	<b>0.0052</b>

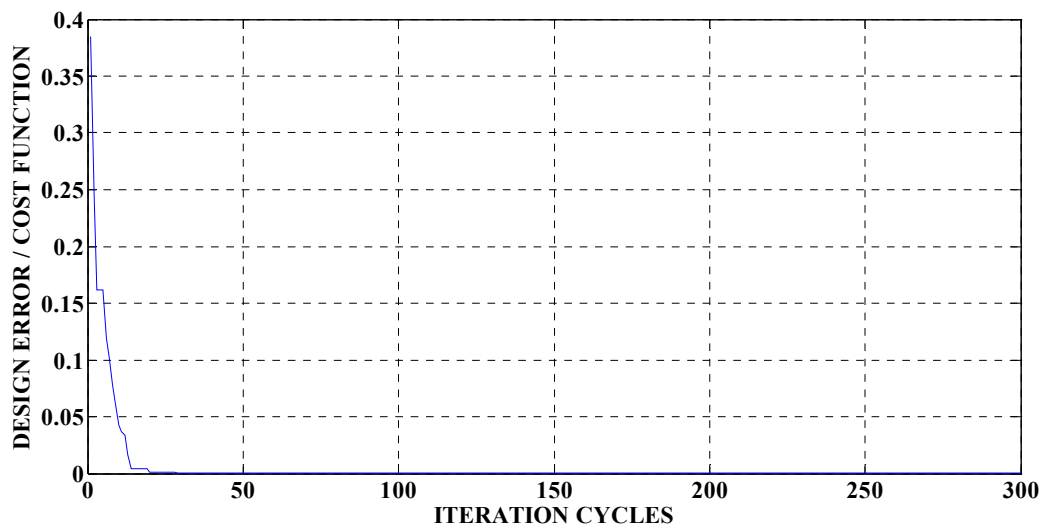


Figure 2 Convergence profile for DE.

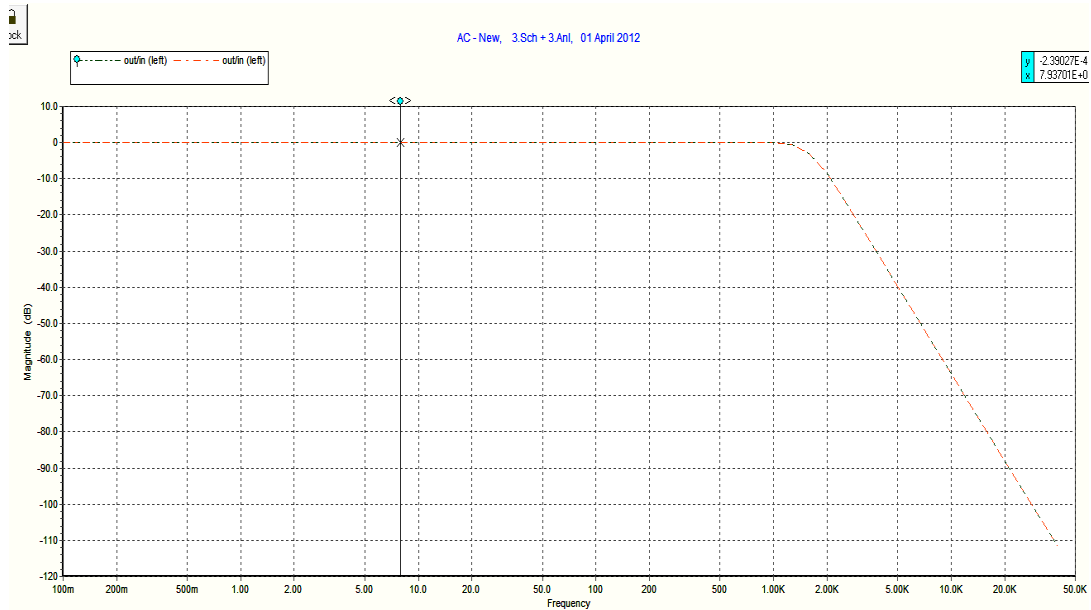


Figure 3 Amplitude response of Butterworth 4th order VCVS low-pass filter designed by DE method.

#### REFERENCES

- [1] Don Lancaster, Active-Filter Cookbook, Howard W. Sams and Co., 1975 ISBN 0-672-211168-8 pp. 8-10.
- [2] Rolf Schaumann, Mac E. Van Valkenburg, "Design of Analog Filters", Oxford University Press, 2001.
- [3] R.L.Geiger and E. Sanchez-Sinencio, "Active filter design using operational transconductance amplifiers: a tutorial," IEEE Circuits Devices Mag, vol. 1, no.1, pp. 20-32, March 1985.
- [4] E. Sanchez-Sinencio and J. Silva-Martinez, "CMOS transconductance amplifiers, architectures and active filters: a tutorial", IEEE Proc. Circuits, Devices Syst vol.147, no.1, pp. 3-12, February 2000.
- [5] Horrocks, D.H., Spittle, M.C. "Component Value Selection for Active Filter Using Genetic Algorithms", In: Proc. IEE/IEEE Workshop on Natural Algorithms in Signal Processing, Chelmsford, UK, vol. 1, pp. 13/1-13/6 (1993).
- [6] Horrocks, D. H., Khalifa, Y. M. A., "Genetic Algorithm Design of Electronic Analogue Circuits Including Parasitic Effects", 1st On-line Workshop on Soft Computing (WSC1), Special Session on Evolutionary Electronics, August 1996, pp19-30.
- [7] Bennett 111, F. H., Koza, J. R., Andre, D., Keane, M. A., " Evolution of a 60 Decibel Op Amp Using Genetic Programming", Proceedings of t h e First International Conference on Evolvable Systems (ICES96), LNCS 1259, Tsukuba, Japan, October, 1996 pp. 455-469.
- [8] Grimbleby, J.B., "Automatic Analogue Network Synthesis Using Genetic Algorithms", Proceedings of the First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems (GALESIAS - 95), pp. 53-58, UK, 1995.
- [9] Wojcikowski, M.; Glinianowicz, J.; Bialko, M, " System for optimisation of electronic circuits using genetic algorithm", Proceedings of the Third IEEE International Conference On Electronics , Circuits and Systems(ICECS),1996,vol 1,pp. 247-250.
- [10] Grimbleby, J.B., "Automatic Synthesis Of active electronic networks using genetic algorithms", Second international Conference On Genetic Algorithms in Engineering Systems: Innovations and Applications 1997,(GALESIA)97, pp. 103-107.
- [11] Salazar,F.A; Mesquita, A, " Synthesis of analog circuits using evolutionary hardware," Sixth Brazillian Symposium on Neural Networks 2000, pp. 101-106.
- [12] Kruiskamp, W. and Leenaerts, D., "DARWIN: CMOS opamp synthesis by means of genetic algorithm", Proceedings of the 32nd Design Automation Conference, New York, NY: Association for Computer Machinery, 1995, pp. 698-703.
- [13] Miller, J. F., Thomson, P., "Combinational and Sequential Logic Optimization Using Genetic Algorithms", Proceedings of t h e First IEE/IEEE International Conference on

- Genetic Algorithms in Engineering Systems (G A L E SIA - 95), UK, 1995, pp. 34-38.
- [14] Kalinli, A. "Optimal Circuit Design Using Immune Algorithm", ICARIS 2004, LNCS 3239, pp. 42-52, 2004.
- [15] Kalinli A., "Component Value Selection for Active Filters Using Parallel Tabu Search Algorithm", AEU Int. Journal of Electronics and Communications, 60, pp. 85-92, 2006.
- [16] Min Jiang, Zhenkun Yang, Zhaohui Gan, "Optimal Components Selection for Analog Active Filters Using Clonal Selection Algorithms", In Lecture Notes in Computer Science 4681, ICIC (1)2007 pp: 950-959, Springer-Verlag, 2007.
- [17] R.S. Zebulum, M.A. Pacheco, M. Vellasco, "Comparison of Different Evolutionary Methodologies Applied to Electronic Filter Design," IEEE Int. Conf. on Evolutionary Computation, Piscataway, NJ: IEEE Press, 1998, pp. 434-439.
- [18] R.A. Vural, T Yildirim, "Component value selection for analog active filter using particle swarm optimization," Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on, vol. 1, 2010, pp. 25-28.
- [19] J. Kennedy and R.C. Eberhart, "Particle swarm optimization," IEEE International Conference on Neural Networks, Piscataway, NJ, 1995, Vol. 4, pp. 1942-1948.
- [20] R. Storn, and K. Price, "Differential evolution- a simple and efficient adaptive scheme for global optimization over continuous spaces", Technical Report, International Computer science Institute, Berkley, pp. 1-12, 1995.