Performance Evaluation of Evolutionary Algorithms for Optimal Filter Design

Revna Acar Vural, *Student Member, IEEE,* Tulay Yildirim, *Member, IEEE,*
Tevfik Kadioglu, and Aysen Basargan

***Abstract*—In analog filter design, component values are selected due to manufactured constant values where performing an exhaustive search on all possible combinations of preferred values for obtaining an optimized design is not feasible. The application of evolutionary algorithms (EA) in analog active filter circuit design and optimization is a promising area which is based on concepts of natural selection and survival of the fittest. In this paper, the performances of genetic algorithm, artificial bee colony optimization, and particle swarm optimization, which are nature-inspired EA techniques, are evaluated for active filter design. Each algorithm is applied to two different filter structures and performances of them are also evaluated when filter design is realized with components selected from different manufactured series.**

***Index Terms*—Active filter design, artificial bee colony algorithm, circuit optimization, genetic algorithm, particle swarm optimization.**

I. Introduction

DESPITE the extensive usage of integrated circuits, discrete components are still preferred in analog active filter design. Conventionally, the values of passive components used in the active filters are chosen as equal to each other. This approach simplifies the design procedure but also limits the freedom of design. Moreover, components are assumed to be ideal and have infinite value during analog design process However, discrete components such as resistors and capacitors are produced in approximate logarithmic multiples of a defined number of constant values such as E12 series. There are also E24, E48, E96, and E192 ranges for components of tighter tolerance with 24, 48, 96, and 192 different values within each decade. In order to reduce the costs and make the design more reliable, discrete components are chosen from these industrial series or other possible produced preferred values. Performing an exhaustive search on all possible combinations of preferred values for obtaining an optimized design is not feasible when components are selected from the tighter tolerance series over wide decade range. Therefore, intelligent search methods must be developed that requires short computation time with high accuracy.

The application of evolutionary techniques in filter design automation and optimization is a promising area which is based on concepts of natural selection and survival of the fittest. In the paper, various algorithms were used for optimal component selection or topology optimization for digital and analog filters [1]–[16]. In [1], optimal microwave filter design with arbitrary geometries is aimed to be designed with particle swarm optimization (PSO) and finite element method. In [2], a genetic algorithm (GA)-based stable and simple design method for 2-D recursive digital filters was developed. Coefficient optimization of digital filters with numerous evolutionary approaches were investigated in [3] and [4]. In [5], a detailed comparative study on analog passive filter design with different evolutionary methodologies was presented. An automated passive analog circuit synthesis procedure based on GA was utilized for the simultaneous generation of both the topology and the component value selection in [6]. Component value and topology evolution were also studied in [7] by means of genetic programming (GP) and in [8] by using GP-based tree representation method. Unconstrained and constrained evolutions were applied toward design of analog (inductor, capacitor, resistor) low-pass filter [9]. This paper has been a successful attempt of application of evolutionary strategy method for analog filter design. A robust design paradigm that exploits the open-ended topological synthesis capability of GP was developed in [10] to evolve robust low-pass and high-pass analog passive filters. In [11], a GA-based growing technique for component value optimization of analog passive filters was presented. Component value selections of analog passive and active filters were also investigated in [12]. Sheta [12] explored the advantages of differential evolution over numerical optimization approaches to perform the operation of selecting the best values of circuit elements for various types of band-pass filters. In [13] and [14], component value selection and topology optimization of analog active filter has been performed using GA and adaptive immune GA, respectively. Moreover, some particular analog active filter types were also optimized using evolutionary approaches in the paper. A voltage con-

Manuscript received September 6, 2010; revised November 12, 2010; accepted January 20, 2011. Date of publication October 10, 2011; date of current version January 31, 2012. This work was supported by the Yildiz Technical University Scientific Projects Coordination Department, under Project 29-04-03-KAP01.

R. A. Vural and T. Yildirim are with the Department of Electronics and Communication Engineering, Yildiz Technical University, Istanbul 34349, Turkey (e-mail: racar@yildiz.edu.tr; tulay@yildiz.edu.tr).

T. Kadioglu is with the National Research Institute of Electronics and Cryptology, TUBITAK, Istanbul 41470, Turkey (e-mail: tevfik.kadioglu@gmail.com).

A. Basargan is with the Department of Electronics and Communication Engineering, Istanbul Technical University, Istanbul 34469, Turkey (e-mail: aysenbasargan@hotmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEVC.2011.2112664

trolled voltage source (VCVS) low-pass Butterworth active filter circuit was designed using clonal selection algorithm (CSA) and results of CSA-based design was compared with results of tabu search (TS)-based, GA-based, and conventional design methods [15]. In [16], a PSO-based component value selection method has been utilized for the optimal design of the same circuit topology used in [15] and less design error was obtained when compared with results of [15]. In [17]–[19], component values of a low-pass state variable active filter (SVF) circuit was selected using GA, TS, and artificial immune algorithm, respectively. However, [17]–[19] used a different calculation of cutoff frequency than regular and practical expression of cutoff frequency statement used in [20] and in this paper. Therefore, despite the fact that [17]–[19] are related to our paper, their results have not been compared here.

In this paper, the applicability of evolutionary algorithms (EAs) in active filter design is investigated by means of accuracy and computation time. A fourth order VCVS Butterworth filter [15] and a second order low-pass SVF structure [21] is designed using GA, ABC algorithm, and PSO. SVF filter is designed using components selected from two different manufactured series in order to investigate the performance of evolutionary methods when tighter tolerances are preferred in component selection. Performance criteria of the filter circuit to be designed constitute the constraints of GA, ABC, and PSO which are described in Section II. In Section III, active filter structures which are selected for demonstrating EA methods are investigated. Section IV describes the conventional design method while the following section explains EA-based filter design in details. EA-based design results which are also validated with simulation program with integrated circuit emphasis (SPICE) simulator are given in Section VI. Section VII concludes with a discussion of simulation results.

II. Evolutionary Algorithms

In artificial intelligence, evolutionary computation (EC) refers to computer-based problem-solving systems that use computational models of evolutionary processes, such as natural selection, survival of the fittest, and reproduction, as the fundamental components of such computational systems [22]. EA is a subset of EC, generic population-based metaheuristic optimization algorithm. EA techniques differ in the implementation details and the nature of the particular applied problem. In this paper, the performances of GA, ABC, and PSO which are nature-inspired EA techniques are evaluated for active filter design. Details of those are introduced in the following sections.

A. Genetic Algorithm

GAs are search algorithms based on the mechanics of natural selection and natural genetics which have been developed by John Holland and his students in the 1970s [23]. GAs combine survival of the fittest among string structures with a structured yet randomized information exchange to form a search algorithm with some of the innovative flair of human search. In every generation, a new set of artificial strings is created using bits of the fittest of the previous generations; an occasional new part is tried for good measure [24]. The process begins with a set of potential solutions or chromosomes (usually in the form of bit strings) that are randomly generated or selected. The entire set of these chromosomes comprises a population. The chromosomes evolve during several iterations or generations. New generations are generated using the crossover and mutation technique. Crossover involves splitting two chromosomes and then combining one half of each chromosome with the other pair. Mutation involves flipping a single bit of a chromosome. The chromosomes are then evaluated using a certain fitness criteria and the best ones are kept while the others are discarded. This process repeats until one chromosome has the best fitness and thus is taken as the best solution of the problem. More details about GAs can be found in [24].

B. Artificial Bee Colony Optimization

ABC algorithm, proposed by Karaboga [25] for real parameter optimization, is a recently introduced optimization algorithm and simulates the foraging behavior of the bee colony. In ABC algorithm, the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. First of all, the food source positions are randomly initialized as $x_i (i = 1, \ldots, SN)$ where SN is the maximum number of the food sources. Each employed bee, whose total number equals to the number of food sources, produces a new food source in her food source site given as follows:

$$v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}) \tag{1}$$

where φ_{ij} is a uniformly distributed real random number within the range $[-1, 1]$, k is the index of the solution chosen randomly from the colony, and j is the index of the dimension of the problem. After producing v_{ij}, this new solution is compared to x_{ij} solution and the employed bee exploits a better source while each onlooker bee whose total number is equal to the number of employed bees selects a food source with the probability given as follows:

$$p_i = \frac{fit_i}{\sum_{j=1}^{SN} fit_j} \tag{2}$$

where fit_i is the fitness of the solution x_{ij} and produces a new source in selected food source site by (1). After all onlookers are distributed to the sources, sources are checked whether they are to be abandoned. The employed bee associated with the abandoned source becomes a scout and makes random search in problem domain by (3). The best food source found so far has been memorized and the production steps are repeated until the stopping criterion is met [26] as follows:

$$x_{ij} = x_j^{\min} + (x_j^{\max} - x_j^{\min}) * rand. \tag{3}$$

C. Particle Swarm Optimization

PSO is an EC method based on the social behavior, movement, and intelligence of swarms searching for an optimal location in a multidimensional search area which has been developed by Eberhart [27]. The approach uses the concept of population and a measure of performance similar to the fitness value used with EAs. Population consists of potential solutions called particles. Each particle is initialized with a random position value. In each iteration, the fitness function is evaluated by taking the current position of the particle in the solution space and the two best values (p_{best}, g_{best}). Personal best value, p_{best}, is the location of the best fitness value obtained so far by the particle. Global best value, g_{best}, is the location of the best fitness value achieved so far considering all the particles in the swarm [27]–[29].

In particle population matrix, containing N number of particles, ith particle with a feature number of D is denoted as $\mathrm{x_i} = [\mathrm{x_{i1}}, \mathrm{x_{i2}}, \ldots, \mathrm{x_{iD}}]$. For each iteration, the velocity and the position vector of the ith particle in $N \times D$ dimension of the search space are updated as follows:

$$v_{id}^{k+1} = w.v_{id}^{k} + c_1.rand_1^k.(pbest_{id}^k - x_{id}^k) + c_2.rand_2^k.(gbest_d^k - x_{id}^k) \quad (4)$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1}. \quad (5)$$

Here, the range of i, d, and k indices are defined as $\{1 \ldots N\}$, $\{1 \ldots D\}$, and $\{1 \ldots max\_iteration\_number\}$, respectively. The acceleration factors c_1 and c_2 indicates the relative attraction toward p_{best} and g_{best}, respectively. Following $rand_1$ and $rand_2$ are random numbers uniformly distributed between zero and one. Inertia weight parameter, w, controls the tradeoff between the global and the local search capabilities of the swarm. Initially, w should be chosen as less than one and should be decreased linearly in each iteration.

PSO algorithm used in this paper has been built up for the global best (g_{best}) PSO model. The g_{best} model is chosen since it converges faster than local best (l_{best}) PSO [22]. This is due to the larger particle connectivity of g_{best} PSO. Each particle can interact with every other one in the swarm and can be attracted to the best position obtained by any other particle.

III. Low-Pass Analog Active Filters

Analog active filters are one of the key components in mixed-signal circuit designs and are widely used in separation of signals according to frequency bands, frequency selection decoding, estimation of a signal from noise, demodulation of signals, and amplifying elements [30]. Analog active filters are comprised of operational amplifiers (op-amp), with resistors and capacitors in their feedback loops, to synthesize the desired filter characteristics. They can have high input impedance, low output impedance, and virtually any arbitrary gain. Possibly, their most important attribute is that they lack inductors, thereby reducing the problems associated with those components. Still, the problems of accuracy and value spacing also affect capacitors, although to a lesser degree. Performance at high frequencies is limited by the gain-bandwidth product of the amplifying elements, but within the amplifier's operating frequency range, op-amp-based active filter can achieve very good accuracy, provided that low-tolerance resistors and capacitors are used. Analog active filters are characterized by four basic properties: the filter type (low-pass, high-pass, bandpass, and others), the passband gain (generally all the filters have unity gain in the passband), the cutoff frequency (the point where the output level has fallen by 3 dB from the maximum level within the passband), and the quality factor Q (determines the sharpness of the amplitude response curve). A low-pass filter is a filter that passes low-frequency signals but attenuates signals with frequencies higher than the cutoff frequency. Low-pass analog active filters are widely used in biomedical instrumentation amplifiers, telecommunications, and radio frequency systems [30], [31]. Two of the low-pass analog active filter topologies used in this paper is examined in the following sections.

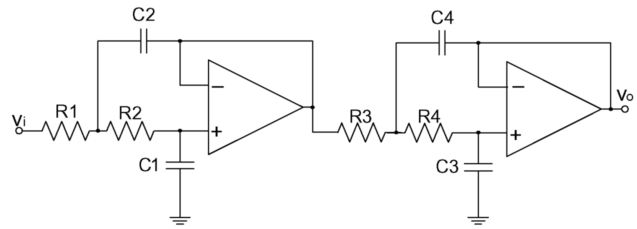

Fig. 1. Butterworth fourth order VCVS low-pass filter [15].

A. Butterworth Filter

Butterworth filters are termed maximally flat-magnitude-response filters, optimized for gain flatness in the passband. The transient response of a Butterworth filter to a pulse input shows moderate overshoot and ringing [21]. Ideally passband extends from 0 to ω_c and stopband extends from ω_s to ∞ where ω_c is the cutoff frequency and ω_s is the stopband frequency. Here, a low-pass Butterworth filter is of concern with passive components and op-amp structures. In order to make a true comparison with the results of [15], the same filter topology is used in this paper. The fourth order VCVS low-pass Butterworth filter can be realized by cascading two second order blocks, its transfer function is given in (6), where ω_{c1} and ω_{c2} is the cutoff frequency of two second order filters and Q_1 and Q_2 is the quality factor of two second order filters as follows:

$$H(s) = \frac{\omega_{c1}^2}{s^2 + \frac{\omega_{c1}}{Q_1}s + \omega_{c1}^2} x \frac{\omega_{c2}^2}{s^2 + \frac{\omega_{c2}}{Q_2}s + \omega_{c2}^2}. \quad (6)$$

The circuit of the fourth order VCVS low-pass Butterworth filter is shown in Fig. 1. According to the circuit in Fig. 1, the transfer function of this filter can be obtained as follows:

$$\begin{aligned} H(s) = {} & \frac{1}{s^2 R_1 R_2 C_1 C_2 + s(R_1 C_1 + R_2 C_1) + 1} \\ & \times x \frac{1}{s^2 R_3 R_4 C_3 C_4 + s(R_3 C_3 + R_4 C_3) + 1}. \end{aligned} \quad (7)$$

According to (6) and (7), definitions of cutoff frequency and quality factor with respect to the circuit components are given in (8) and (9) as follows:

$$\omega_{c1} = \frac{1}{\sqrt{\mathrm{R_1 R_2} C_1 C_2}}, \omega_{c2} = \frac{1}{\sqrt{\mathrm{R_3 R_4} C_3 C_4}} \quad (8)$$

$$Q_1 = \frac{\sqrt{\mathrm{R_1 R_2} C_1 C_2}}{R_1 C_1 + R_2 C_1}, Q_2 = \frac{\sqrt{\mathrm{R_3 R_4} C_3 C_4}}{R_3 C_3 + R_4 C_3}. \quad (9)$$

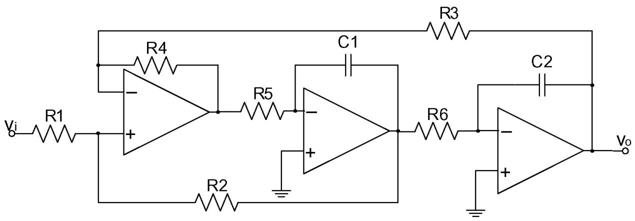

Fig. 2. Second order state variable low-pass filter [21].

The fourth order Butterworth filter is implemented by cascading two second order ones, in this paper and as in [15], their cutoff frequency ω_{c1} and ω_{c2} are the same as 10 k rad/s, their quality factors Q_1 and Q_2 are 1/0.7654 and 1/1.8478, respectively, where quality factor values are determined from the table of low-pass second order factors.

B. State Variable Filter

A state variable filter (SVF) realizes the state-space model directly. The instantaneous output voltage of one of the integrators corresponds to one of the state-space model's state variables. SVF can produce simultaneous low-pass, high-pass, and bandpass outputs from a single input. A second order SVF is illustrated in Fig. 2 and is well described in [21]. The low-pass output is assumed here to be the desired output.

The response of a second order low-pass circuit is specified by the passband gain (H_{SVF}), the cutoff frequency ($\omega_{SVF} = 2\pi f_{SVF}$), and the selectivity factor (Q_{SVF}). These quantities are given in terms of passive component values as follows:

$$H_{SVF} = \frac{R_2(R_3 + R_4)}{R_3(R_1 + R_2)}, \omega_{SVF} = \sqrt{\left(\frac{R_4}{R_3}\right)\left(\frac{1}{C_1 C_2 R_5 R_6}\right)} \tag{10}$$

$$Q_{SVF} = \frac{R_3(R_1 + R_2)}{R_1(R_3 + R_4)}\sqrt{\frac{C_1 R_4 R_5}{C_2 R_3 R_6}}. \tag{11}$$

The specification chosen here is $\omega_{SVF} = 10$ k rad/s ($f_{SVF} = 10\,000/(2^*\pi) = 1591.55$ Hz) and $Q_{SVF} = 0.707$ for reduced peak on low-pass response. The passband gain *(H)* is not very critical in most applications since it can be compensated by other cascaded analogue circuits. In the conventional design procedure [21], H is fixed at some value; however, for EA methods, it is unconstrained.

IV. Conventional Design Method

Conventional methods make all resistors equal to a normalized value of unity (1Ω), and then set the cutoff frequency of two second order Butterworth filters ω_{c1}, ω_{c2} to 10 k rad/s, and quality factor of two second order Butterworth filters Q_1, Q_2 to 1/0.7654 and 1/1.8478, respectively. Similarly, all resistors are equalized to a normalized value and set cutoff frequency of second order state variable filter ω_{SVF} to 10 k rad/s and quality factor Q_{SVF} to 0.707. Thus, the values of four capacitors can be obtained according to (8) and (9) for VCVS Butterworth filter and two capacitors according to (10) and (11) for state variable filter. As the values of components may not in the feasible range, for an exact design, a sensible way is first to multiply all values of resistors by a reasonable factor to make their values in the middle of the range. At the same time, all the capacitor values must be divided by the same factor [21]. For a design procedure where manufactured components are chosen, the exact component values are rounded to the nearest preferred values which will increase the total design error.

V. EA-Based Active Filter Design

In order to investigate the usage of EAs in active filter circuit design and to compare with previously used methods, two different low-pass analog active filter circuits given in Figs. 1 and 2 are selected. By establishing design criteria and design parameters to EA and satisfying desired constraints, the optimal circuit structure was aimed to be designed by the algorithm. Design problem has been introduced by composing an equation consists of design parameters as a cost function (CF). In the beginning of the algorithm, a certain range was determined for design parameters by human designer. EA should minimize the given CF and obtain design criteria and design parameter values for the given range which gives minimum CF value.

In this paper, GA, PSO, and ABC algorithms are utilized for EA-based active filter design and performances of those are evaluated by means of computation time and accuracy. The aim is to estimate the preferred values of resistors and capacitors of the selected circuit with minimum design error. Each component used in filter design tasks was chosen to take value in the uttermost range of 10^3–10^6 Ω for the resistors and 10^{-9}–10^{-6} F for capacitors. Values outside these ranges were judged to lead to unwanted practical effects such as stray capacitance effects or large signal currents [17], [31].

For both filter design tasks, we investigated the performance of EAs by varying own parameters in order to obtain the minimum total error value. Considering GA method, it was initiated with 15, 20, and 30 chromosomes with each comprised of eight genes in population. "Parent" genes were selected with a roulette wheel selection. "Child" genes were generated using random single-point crossover applied with a probability of 0.5, 0.63, and 0.8 and mutation was applied to each bit in the gene with a probability of 0.01, 0.07, and 0.15. Uppermost number of iterations was determined as 10 000. PSO method was applied to filter design tasks such that initial population matrix size was $N \times 16$, where row number of N (N = 5, 10, 15, respectively) indicates the number of particles in the population and column number of 16 is the dimension of particle vector which is given in (12). ABC optimization utilizes with a food number of 5, 50, and 500 with equal numbers of employer and onlooker bees. Maximum search limit (SL) is defined as 10 100 and 1000 cycles and maximum iteration number is 10 000 for both filter design tasks where each of eight components is represented with two parameters as similar to PSO method as follows:

$$\begin{bmatrix} a_1\ al_1\ ; b_1\ bl_1\ c_1\ cl_1\ d_1\ dl_1\ e_1\ el_1\ f_1\ fl_1\ g_1\ gl_1\ h_1\ hl_1 \\ \vdots \\ \vdots \\ a_{10}\ al_{10}\ b_{10}\ bl_{10}\ c_{10}\ cl_{10}\ d_{10}\ dl_{10}\ e_{10}\ el_{10}\ f_{10}\ fl_{10}\ g_{10}\ gl_{10}\ h_{10}\ hl_{10} \end{bmatrix}_{10\times 16}. \tag{12}$$

R1	R2	R3	R4	C1	C2	C3	C4

Fig. 3. Component values in a chromosome for Butterworth filter.

The previous paragraph explains the common characteristics of each method used in both filter design tasks. However, since circuit topologies differ in the following filter design tasks, EA methods used in this paper are utilized for each design task. In the following, design error definition for each filter structure and related component representation method for each EA technique is presented.

A. Butterworth Filter Design

In order to make a true comparison with previous methods in [15], same error criterion with [15] is used here. Design error in (13) is the summation of cost function error of cutoff frequency (CF_{ω}) and quality factor (CF_Q). Those definitions are given in (14) as follows:

$$Error_{total-BF} = 0.5CF_{\omega} + 0.5CF_Q \tag{13}$$

$$CF_{\omega} = \frac{|\omega_{c1} - \omega_c| + |\omega_{c2} - \omega_c|}{\omega_c},$$
$$CF_Q = \left|Q_1 - \frac{1}{0.7654}\right| + \left|Q_2 - \frac{1}{1.8478}\right| \tag{14}$$

In order to introduce the Butterworth filter design task to GA, PSO, and ABC, a CF which includes values of discrete components ($R_{1\ldots4}$, $C_{1\ldots4}$) as design parameters is constituted as given in (15). Here, ω_c is chosen as 10 k rad/s. GA, PSO, and ABC should obtain the minimum value of the CF, and the preferred values of design parameters that minimize CF as follows:

$$Error_{total} = \left(\begin{array}{l} 0.5 \dfrac{\left|\frac{1}{\sqrt{R_1R_2C_1C_2}} - \omega_c\right| + \left|\frac{1}{\sqrt{R_3R_4C_3C_4}} - \omega_c\right|}{\omega_c} \\ +0.5\left(\left|\frac{\sqrt{R_1R_2C_1C_2}}{R_1C_1+R_2C_1} - \frac{1}{0.7654}\right| + \left|\frac{\sqrt{R_3R_4C_3C_4}}{R_3C_3+R_4C_3} - \frac{1}{1.8478}\right|\right) \end{array} \right). \tag{15}$$

The right side of (15) would constitute the CF which EA techniques would minimize. It is desired to obtain the exact values of design parameters ($R_{1\ldots4}$, $C_{1\ldots4}$) which equate CF to a very close value to zero. In each decade, any of 12 preferred values can be taken according to standard E12 series within the range of 10^3–$10^6\ \Omega$ for resistors and 10^{-9}–10^{-6} F for capacitors.

1) *Component Representation for GA:* The values of resistors and capacitors constitute the dimension of the chromosome where each chromosome is comprised of eight genes as given in Fig. 3. Each gene is binary coded 4 bits, representing the resistor and capacitor values compatible with E12 series. As a result, considering VCVS Butterworth active filter design task, GA utilizes chromosomes with 32 bits.

2) *Component Representation for PSO and ABC:* The values of capacitors and resistors constitute the dimension of particle vector. Since the probable values vary from three decade range, a coding scheme is used as follows:

$$\begin{array}{ll} R_1 = a \times 100 \times 10^{a1}(\Omega), & R_2 = b \times 100 \times 10^{b1}(\Omega) \\ R_3 = c \times 100 \times 10^{c1}(\Omega), & R_4 = d \times 100 \times 10^{d1}(\Omega) \\ C_1 = e \times 100 \times 10^{e1}(\text{pF}), & C_2 = f \times 100 \times 10^{f1}(\text{pF}) \\ C_3 = g \times 100 \times 10^{g1}(\text{pF}), & C_4 = h \times 100 \times 10^{h1}(\text{pF}). \end{array} \tag{16}$$

Since each resistor should take an E12 serial value in the range of 10^3–$10^6\ \Omega$, the design constraint for resistors given in (17) must be satisfied. Similarly, each capacitor should take E12 serial value in the range of 10^{-9}–10^{-6} F. If capacitor values are defined in picofarads (pF) then design constraint for capacitors given in (18) would be valid as follows:

$$0.1 \le a, b, c, d \le 0.822 \le a1, b1, c1, d1 \le 4 \tag{17}$$

$$0.1 \le e, f, g, h \le 0.822 \le e1, f1, g1, h1 \le 4. \tag{18}$$

B. State Variable Filter Design

Design error in (19) is the summation of cost function error of cutoff frequency (CF_{ω}) and quality factor (CF_Q). Those definitions are given in (20). Since passband gain is unconstrained, it is not included to design error equation as follows:

$$Error_{total-SVF} = 0.5CF_{\omega} + 0.5CF_Q. \tag{19}$$

$$CF\omega = \frac{|\omega_{SVF} - \omega_0|}{\omega_0}, CF_Q = \frac{|Q_{SVF} - Q|}{Q} \tag{20}$$

In order to introduce the SVF design task to GA, PSO, and ABC, a CF which includes values of discrete components ($R_{1\ldots6}$, $C_{1,2}$) as design parameters is constituted as given in (21). Here, ω_0 and Q are set to 10 k rad/s and 0.707, respectively. GA, PSO, and ABC should obtain the minimum value of the CF, and the preferred values of design parameters that minimize CF as follows:

$$Error_{total} = 0.5\frac{\left|\sqrt{\left(\frac{R_4}{R_3}\right)\left(\frac{1}{C_1C_2R_5R_6}\right)} - \omega_0\right|}{\omega_0} + 0.5\frac{\left|\frac{R_3(R_1+R_2)}{R_1(R_3+R_4)}\sqrt{\frac{C_1R_4R_5}{C_2R_3R_6}} - Q\right|}{Q}. \tag{21}$$

The right side of (21) constituted the CF which PSO algorithm would minimize. It is desired to obtain the exact values of design parameters ($R_{1\ldots6}$, $C_{1,2}$) which equate CF to a very close value to zero. In each decade, any of 24 and 96 preferred values can be taken according to standard E24 series and E96 series, respectively, within the range of 10^3–$10^6\ \Omega$ for resistors and 10^{-9}–10^{-6} F for capacitors.

1) *Component Representation for GA:* The values of resistors and capacitors constitute the dimension of the chromosome. Each chromosome is comprised of eight genes as given in Fig. 4. Each gene is binary coded 5 bits, representing the resistor and capacitor values for E24 series and 7 bits representing the resistor and capacitor values for E96 series. As a result, for SVF design task, GA utilizes chromosomes with 40 bits and 56 bits for SVF design with components selected from E24 and E96 series, respectively.

TABLE I
EFFECTS OF GA'S OWN PARAMETERS ON BUTTERWORTH FILTER PERFORMANCE

CF Values (GA-E12)		MP								
		MP = 0.01			MP = 0.07			MP = 0.15		
		CP			CP			CP		
		CP = 0.5	CP = 0.63	CP = 0.8	CP = 0.5	CP = 0.63	CP = 0.8	CP = 0.5	CP = 0.63	CP = 0.8
No. of chrom-osomes (N)	N=15	0.0192	0.0373	0.044	0.0181	0.0402	0.0168	0.0531	0.0244	0.0462
	N=20	0.0371	**0.0166**	0.025	0.0173	0.018	0.029	0.0722	0.0214	0.0831
	N=30	0.0194	0.064	0.023	0.063	0.079	0.067	0.0564	0.0651	0.0632

R1	R2	R3	R4	R5	R6	C1	C2

Fig. 4. Component values in a chromosome for SVF.

2) *Component Representation for PSO and ABC:* In each decade, any of 24 and 96 preferred values can be taken according to standard E24 series and E96 series, respectively. Since the probable values vary from three decade range, a coding scheme is used as follows:

$$\begin{aligned} &R_1 = a \times 100 \times 10^{a1}(\Omega), && R_2 = b \times 100\, x 10^{b1}(\Omega)\\ &R_3 = c \times 100 \times 10^{c1}(\Omega), && R_4 = d \times 100 \times 10^{d1}(\Omega)\\ &R_5 = e \times 100 \times 10^{e1}(\text{pF}), && R_6 = f \times 100 \times 10^{f1}(\text{pF})\\ &C_1 = g \times 100 \times 10^{g1}(\text{pF}), && C_2 = h \times 100 \times 10^{h1}(\text{pF}). \end{aligned} \tag{22}$$

First, SVF is designed with components that are compatible with E24 series. Since each resistor should take an E24 serial value in the range of 10^3–$10^6\,\Omega$, the design constraint for resistors given in (23) must be satisfied. Similarly, each capacitor should take E24 serial value in the range of 10^{-9}–10^{-6} F. If capacitor values are defined in pF then design constraint for capacitors given in (24) would be valid as follows:

$$0.1 \le a, b, c, d, e, f \le 0.91 \quad 2 \le a1, b1, c1, d1, e1, f1 \le 4 \tag{23}$$

$$0.1 \le g, h \le 0.91 \quad 2 \le g1, h1 \le 4. \tag{24}$$

Second, PSO and ABC algorithms estimated the component values of SVF circuit that are compatible with E96 series. Design constraints are specified similarly as explained for E24 series. The difference is the upper limit constraints for resistors and capacitors given as follows:

$$0.1 \le a, b, c, d, e, f \le 0.976 \quad 2 \le a1, b1, c1, d1, e1, f1 \le 4 \tag{25}$$

$$0.1 \le g, h \le 0.976 \quad 2 \le g1, h1 \le 4. \tag{26}$$

VI. SIMULATION RESULTS

The performances of EA techniques on different types of filter design are evaluated by means of accuracy and computation time. Simulation results of GA, PSO, and ABC-based filter design tasks are investigated in the following. In order to evaluate the feasibility of component values obtained by EA-based design method, each filter topology is redesigned in SPICE simulator with corresponding component values and an appropriate op-amp macromodel.

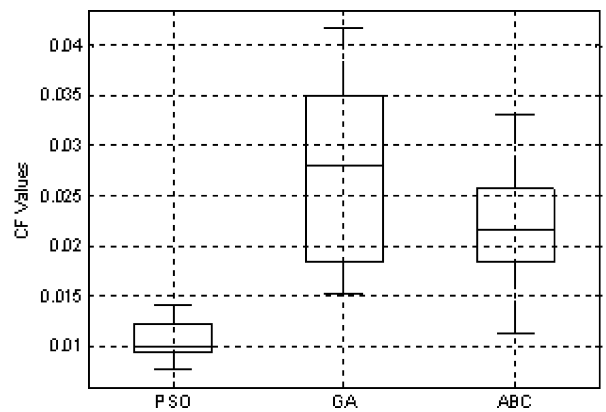

Fig. 5. Box and whisker plots for comparing EAs' performances for Butterworth filter design with E12 series over 20 runs.

A. Butterworth Filter Design Results

In Butterworth filter design with components selected from E12 series, the target CF result with EA techniques is aimed to be smaller than 0.018. Considering GA method, this requirement has been met at the 7768th iteration and the exact value of CF is obtained as 0.0166 with mutation probability (MP) of 0.01, crossover probability (CP) of 0.63, and chromosome number of 20. The effects of GA's own parameters over total error values (CF values) are given in Table I. Computation time for GA is 4.1 min due to the search of exact component values compatible with E12 series. Different from GA, PSO utilizes a search space within the constraints as given in (17) and (18). This method shortens the computation time required (3.7 s), and obtained component values are in the provided ranges; however, these values do not fit to the E12 values. Therefore, PSO-based results of discrete component values are rounded to the nearest preferred E12 serial value. Rounded values are evaluated whether they meet the target CF result. If not, PSO algorithm is rerun until satisfying CF results have been obtained when ideal values found by PSO is rounded to the nearest preferred E12 series. The duration of total process is 3.2 min for Butterworth filter design with a total error of 0.0076 utilizing acceleration factors of 1.7 and particle number of 10. The effects of PSO algorithm's own parameters over total error values (CF values) are given in Table II. Details of PSO-based Butterworth filter design are explained in [16].

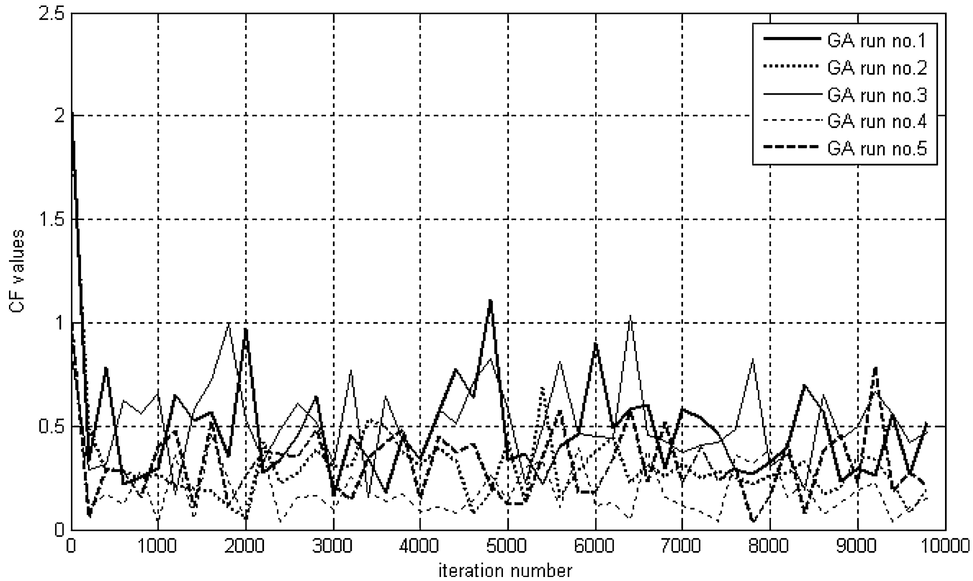

Fig. 6. CF values versus iteration number for GA method (E12 series).

TABLE II
EFFECTS OF PSO ALGORITHM'S OWN PARAMETERS ON BUTTERWORTH FILTER PERFORMANCE

CF Values (PSO-E12)		Acceleration Factors (c_1, c_2)		
		$c_1 = c_2 = 1.5$	$c_1 = c_2 = 1.7$	$c_1 = c_2 = 2$
No. of Particles (N)	N=5	0.014	0.0097	0.0796
	N=10	0.011	**0.0076**	0.0895
	N=15	0.014	0.0091	0.0139

TABLE III
EFFECTS OF ABC ALGORITHM'S OWN PARAMETERS ON BUTTERWORTH FILTER PERFORMANCE

CF Values (ABC-E12)		SL		
		SL = 10	SL = 100	SL = 1000
No. of Bees (N)	NP = 10	0.2611	0.0467	0.1569
	NP = 100	0.0951	0.0193	0.0268
	NP = 1000	0.0225	**0.0113**	0.0251

ABC optimization method obtained a design error of 0.0113 at 525th iteration in 0.7 s with a SL of 100 and total bee population of 1000. The effects of ABC algorithm's own parameters over total error values (CF values) are given in Table III.

Following, we explored the performance of EAs over 20 runs with optimal own parameters as given in Tables I–III. The resulting CF values obtained in each run were used to produce box and whisker plots to show the median performance as well as outliers as given in Fig. 5.

Upper and lower ends of boxes represent 75th and 25th percentiles. Median is depicted by the red line. The whiskers are lines extending from each end of the boxes to show the extent of the rest of the data. Outliers are data with values beyond the ends of the whiskers.

CF values versus iteration number for five independent runs are plotted in Figs. 6–8 for GA, ABC, and PSO algorithms, respectively. In those figures, it can be seen that number of iterations required to achieve the quality requirements are slightly different in each run.

Exact values of discrete components, deviations, and total error of GA, PSO, and ABC-based design and previously used methods are tabulated in Table IV. Compared to the previous methods and other techniques, PSO algorithm achieved a smaller design error in a shorter computation time than GA even if the ideal results are rounded to the nearest preferred values. However, in terms of execution time, ABC outperforms both GA and PSO.

The frequency responses of the filter achieved by the EA techniques are shown in Fig. 9. Butterworth filter is realized with E12 compatible results of GA, ABC, and PSO methods and LM741 op-amp macromodel with SPICE simulator. SPICE simulation proves that all proposed methods provide a maximally flat response in the passband. In this figure, x-axis represents the amplitude response in decibels and y-axis is the frequency. Since ω_c is the frequency point which the output of the circuit is -3 dB of the nominal passband value, those points are also marked on Fig. 9. Yet, it should be noted that cutoff frequency obtained at the output should be lower than each of the designed cascade stages. Due to the non-idealities of the LM-741 op-amp macromodel, this decrement was not observed in Fig. 9.

B. State Variable Filter Design Results

In state variable filter design with components selected from E24 and E96 series, the target CF result with EA techniques is aimed to be smaller than 1×10^{-3}. Considering GA method for design with E24 series, this requirement has been met at the 1541th iteration and the exact value of CF is obtained as 2.1735×10^{-4} with MP of 0.01, CP of 0.63, and chromosome number of 20. The effects of GA's own parameters over total error values (CF values) are given in Table V. Computation time for GA is 5.2 min due to the search of exact component values compatible with E24 series. GA-based design with E96 series possessed a CF value of 1.0449×10^{-4} at 4441th

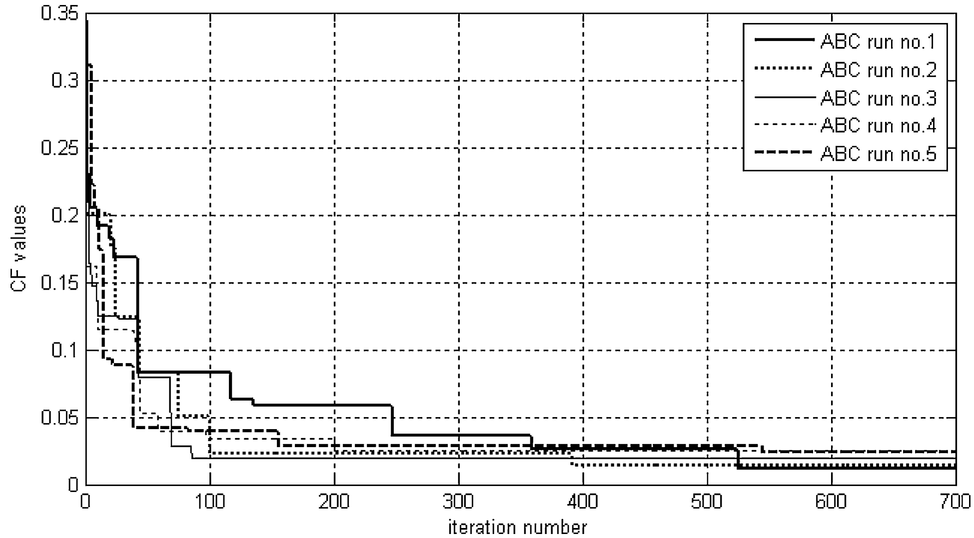

Fig. 7. CF values versus iteration number for ABC method (E12 series).

TABLE IV
COMPONENT VALUES AND PERFORMANCE OF PREVIOUS METHODS AND GA, ABC, PSO TECHNIQUES FOR BUTTERWORTH FILTER DESIGN

	Conventional [15]		TS [15]	GA [15]	CSA [15]	GA	ABC	PSO [16]
	Ideal Values	Nearest Preferred						
R1	1 kΩ	1 kΩ	27 kΩ	4.7 kΩ	4.7 kΩ	6.8 kΩ	4.7 kΩ	4.58 kΩ
R2	1 kΩ	1 kΩ	270 Ω	1.8 kΩ	4.7 kΩ	6.8 kΩ	4.7 kΩ	4.7 kΩ
C1	38.27 nF	39 nF	2.7 nF	12 nF	8.2 nF	5.6 nF	8.2 nF	8.2 nF
C2	26.13 nF	0.27 μF	470 nF	0.1 μF	56 nF	39 nF	56 nF	56 nF
R3	1 kΩ	1 kΩ	220 kΩ	100 kΩ	270 Ω	39 kΩ	1 kΩ	1.1 kΩ
R4	1 kΩ	1 kΩ	820 Ω	4.7 kΩ	27 kΩ	1 kΩ	39 kΩ	1 kΩ
C3	92.39 nF	0.1 μF	82 nF	1.8 nF	6.8 nF	4.7 nF	4.7 nF	87.6 nF
C4	0.2613 μF	0.1 μF	0.68 nF	12 nF	0.2 μF	56 nF	56 nF	102.2 nF
$\Delta\omega$	0	0.02549	0.01291	0.01503	0.01141	0.0179	0.0201	0.0135
ΔQ	0	0.05026	0.04264	0.02130	0.00436	0.0153	0.0024	0.0018
Total error	0	0.03788	0.02778	0.01817	0.00789	0.0166	0.0113	**0.0076**

iteration with MP of 0.01, CP of 0.63, and chromosome number of 20. The effects of GA's own parameters over total error values (CF values) are given in Table VI. The search of exact component values compatible with E96 series requires 7.4 min. PSO and ABC design method was explained in *Butterworth Filter Design Results* section and utilized for E24 and E96 series. The duration for PSO-based design with E24 series is 4.5 min with a total design error of 3.6603×10^{-4} utilizing acceleration factors of 1.7 and particle number of 10.

The effects of PSO algorithm's own parameters over total error values (CF values) are given in Table VII. Considering E96 series, PSO possessed a CF result of 3.1084×10^{-4} in 5.6 min with acceleration factors of 1.7 and particle number of 10. The effects of PSO algorithm's own parameters over total error values (CF values) are given in Table VIII. ABC optimization of SVF design task results in 3.4 s after 110 iterations with a total error of 3.81×10^{-4} utilizing SL of 100 and bee population of 1000 when components are selected from E24 series. The effects of ABC algorithm's own parameters over total error values (CF values) are tabulated in Table IX. However, when considering E96 series, ABC is the most successful algorithm among the others with the shortest computation time and minimum design error. ABC-based SVF design is concluded in 2.6 s after 175 iterations with a total error of 0.171×10^{-4} utilizing SL of 100 and bee population of 1000. The effects of ABC algorithm's own parameters over CF values are given as follows; we explored the performance of EAs over 20 runs with optimal own parameters as given in Tables V–X. The resulting CF values obtained in each run were used to produce box and whisker plots for E24 series and E96 series in Figs. 10 and 11, respectively. Detailed explanation of this type of plots was

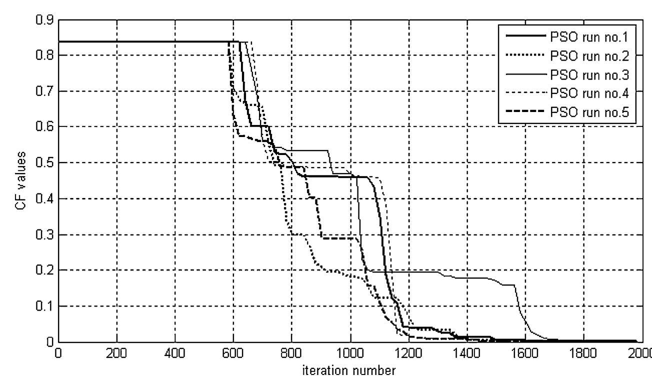

Fig. 8. CF values versus iteration number for PSO method (E12 series).

TABLE V

EFFECTS OF GA'S OWN PARAMETERS ON SVF PERFORMANCE: E24 SERIES

CF Values (GA-E24)		MP								
		MP = 0.01			MP = 0.07			MP = 0.15		
		CP			CP			CP		
		CP = 0.5	CP = 0.63	CP = 0.8	CP = 0.5	CP = 0.63	CP = 0.8	CP = 0.5	CP = 0.63	CP = 0.8
No. of chrom-osomes (N)	N = 15	26×10^{-4}	14×10^{-4}	12×10^{-4}	18×10^{-4}	11×10^{-4}	14×10^{-4}	9.5×10^{-4}	4.94×10^{-4}	6.57×10^{-4}
	N = 20	3.4×10^{-4}	$\mathbf{2.17 \times 10^{-4}}$	8.26×10^{-4}	4.1×10^{-4}	6.98×10^{-4}	3.8×10^{-4}	2.6×10^{-4}	9.95×10^{-4}	4.93×10^{-4}
	N = 30	8.7×10^{-4}	3.26×10^{-4}	5.52×10^{-4}	5.8×10^{-4}	3.66×10^{-4}	7.1×10^{-4}	11×10^{-4}	7.09×10^{-4}	9.91×10^{-4}

TABLE VI

EFFECTS OF GA'S OWN PARAMETERS ON SVF PERFORMANCE: E96 SERIES

CF Values (GA-E24)		MP								
		MP = 0.01			MP = 0.07			MP = 0.15		
		CP			CP			CP		
		CP = 0.5	CP = 0.63	CP = 0.8	CP = 0.5	CP = 0.63	CP = 0.8	CP = 0.5	CP = 0.63	CP = 0.8
No. of chrom-osomes (N)	N = 15	3.25×10^{-4}	3.34×10^{-4}	5.4×10^{-4}	5.1×10^{-4}	1.94×10^{-4}	2.71×10^{-4}	2×10^{-4}	8.1×10^{-4}	6.94×10^{-4}
	N = 20	8.07×10^{-4}	$\mathbf{1.045 \times 10^{-4}}$	4.6×10^{-4}	2.1×10^{-4}	1.43×10^{-4}	7.2×10^{-4}	8×10^{-4}	2.5×10^{-4}	3.89×10^{-4}
	N = 30	3.17×10^{-4}	3.41×10^{-4}	4.3×10^{-4}	3.8×10^{-4}	2.5×10^{-4}	6.56×10^{-4}	2×10^{-4}	3.16×10^{-4}	5.32×10^{-4}

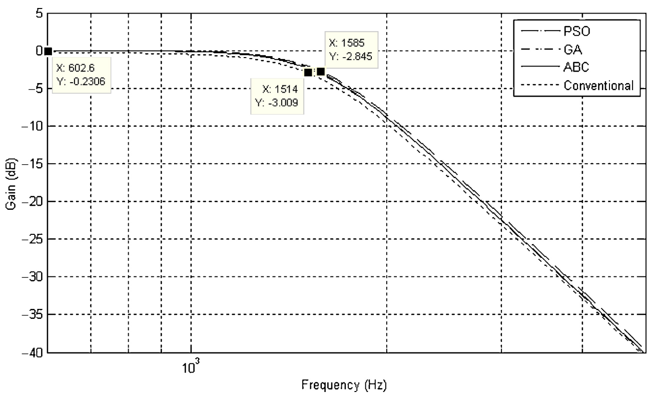

Fig. 9. Frequency responses of fourth order Butterworth VCVS low-pass filter (compatible with E12 series).

TABLE VII

EFFECTS OF PSO ALGORITHM'S OWN PARAMETERS ON SVF PERFORMANCE: E24 SERIES

CF Values (PSO-E24)		Acceleration Factors (c_1, c_2)		
		$c_1 = c_2 = 1.5$	$c_1 = c_2 = 1.7$	$c_1 = c_2 = 2$
No. of particles (N)	$N = 5$	6.99×10^{-4}	4.76×10^{-4}	2.05×10^{-3}
	$N = 10$	6.54×10^{-4}	$\mathbf{3.66 \times 10^{-4}}$	3.53×10^{-3}
	$N = 15$	5.26×10^{-4}	3.72×10^{-4}	1.05×10^{-2}

given for Fig. 5. Different from Fig. 5, outliers are marked with red plus.

Considering SVF design, CF values versus iteration number for five independent runs are plotted in Figs. 12–14 for GA, ABC, and PSO algorithms utilized for E24 series, and in Figs. 15–17 for the same EAs utilized for E96 series, respectively. In those figures, it can be seen that number of iterations required to achieve the quality requirements are slightly different in each run.

Exact values of discrete components for both E24 and E96 series, deviations, and total error of GA, PSO, and ABC-based

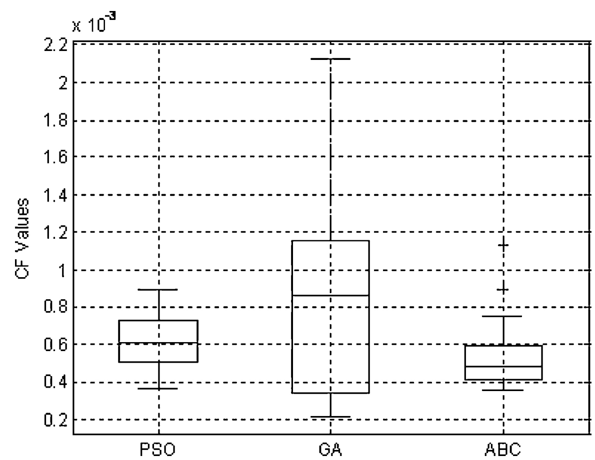

Fig. 10. Box and whisker plots for comparing EAs' performances for SVF design with E24 series over 20 runs.

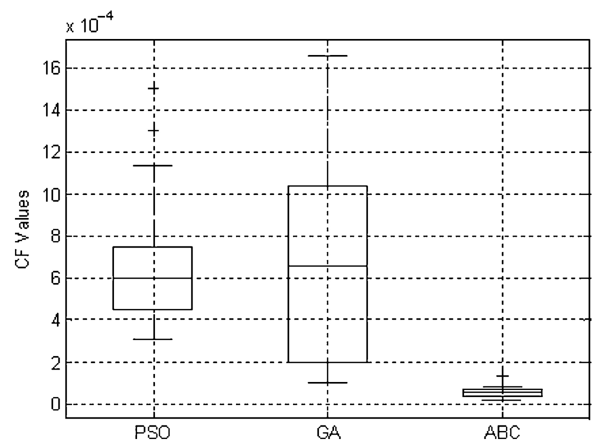

Fig. 11. Box and whisker plots for comparing EAs' performances for SVF design with E96 series over 20 runs.

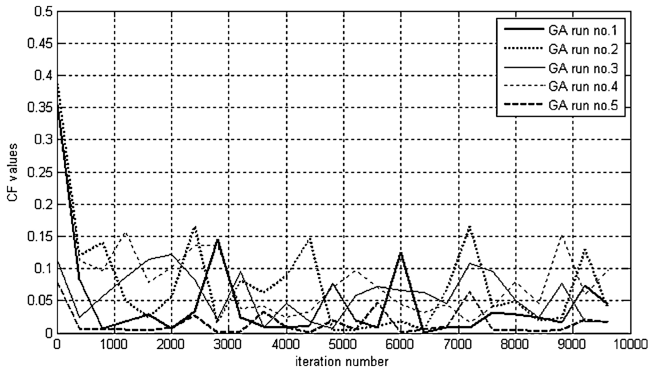

Fig. 12. CF values versus iteration number for GA method (E24 series).

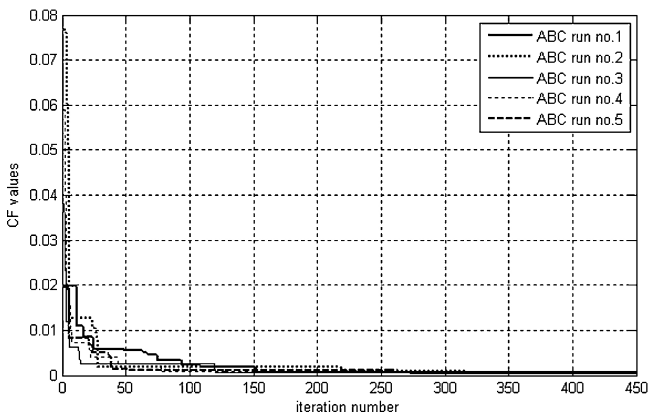

Fig. 13. CF values versus iteration number for ABC method (E24 series).

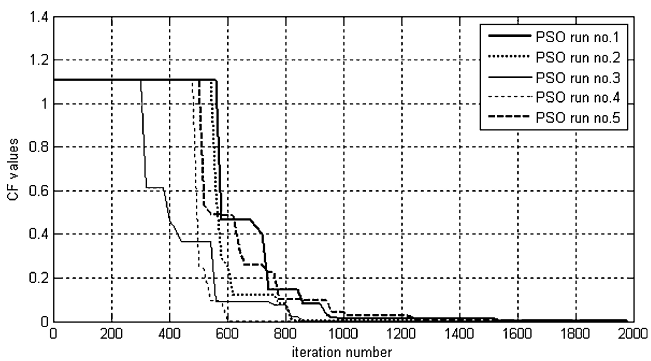

Fig. 14. CF values versus iteration number for PSO method (E24 series).

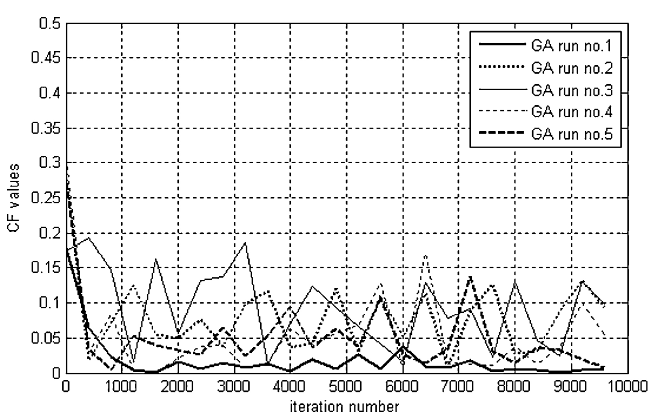

Fig. 15. CF values versus iteration number for GA method (E96 series).

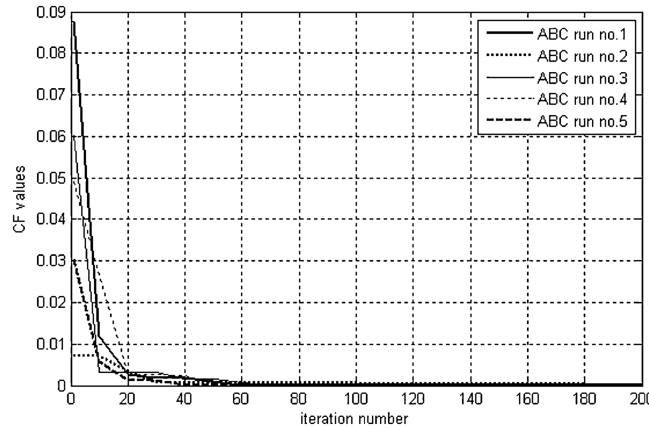

Fig. 16. CF values versus iteration number for ABC method (E96 series).

TABLE VIII
EFFECTS OF PSO ALGORITHM'S OWN PARAMETERS ON SVF PERFORMANCE: E96 SERIES

CF Values (PSO-E96)		Acceleration Factors (c_1, c_2)		
		$c_1 = c_2 = 1.5$	$c_1 = c_2 = 1.7$	$c_1 = c_2 = 2$
No. of particles (N)	$N = 5$	6.01×10^{-4}	5.71×10^{-4}	2.36×10^{-2}
	$N = 10$	8.01×10^{-4}	$\mathbf{3.11 \times 10^{-4}}$	4.36×10^{-2}
	$N = 15$	9.89×10^{-4}	4.52×10^{-4}	7.34×10^{-2}

TABLE IX
EFFECTS OF ABC ALGORITHM'S OWN PARAMETERS ON SVF PERFORMANCE: E24 SERIES

CF Values (ABC-E24)		SL		
		SL = 10	SL = 100	SL = 1000
No. of bees (NP)	NP = 10	0.00944	0.00621	0.01572
	NP = 100	0.00070	0.00068	0.00363
	NP = 1000	0.00048	**0.00038**	0.00056

TABLE X
EFFECTS OF ABC ALGORITHM'S OWN PARAMETERS ON SVF PERFORMANCE: E96 SERIES

CF Values (ABC-E96)		SL		
		SL = 10	SL = 100	SL = 1000
No. of bees (NP)	NP = 10	0.08794	0.001124	0.00219
	NP = 100	0.00057	0.000176	0.00049
	NP = 1000	0.0006	**0.000017**	0.00088

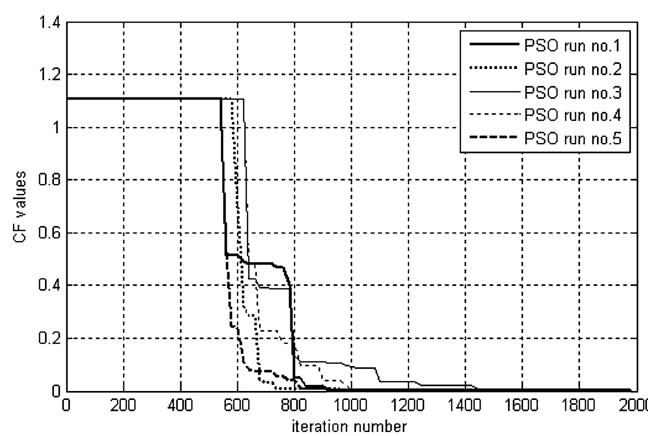

Fig. 17. CF values versus iteration number for PSO method (E96 series).

TABLE XI
COMPONENT VALUES AND PERFORMANCE OF CONVENTIONAL METHOD AND GA, ABC, PSO TECHNIQUES FOR STATE VARIABLE FILTER DESIGN

	Conventional [20]		GA (E24)	ABC (E24)	PSO [20] (E24)	GA (E96)	ABC (E96)	PSO [20] (E96)
	Ideal Values	Nearest Preferred						
R1 (Ω)	4000	4120	43 000	62 000	10 000	69 000	59 000	10 200
R2 (Ω)	1656	1690	5600	1000	1650	2550	88 700	8660
R3 (Ω)	4000	4120	24 000	91 000	30 110	65 300	54 900	14 700
R4 (Ω)	4000	4120	280 000	4300	212 000	237 000	90 900	187 000
R5 (Ω)	4000	4120	4400	27 000	1039	2870	10 000	1130
R6 (Ω)	4000	4120	9200	1800	3900	1430	51 100	2940
C1 (nF)	25	25.5	180	2.7	470	110	7.5	464
C2 (nF)	25	25.5	16	3.6	37	80.4	4.32	82.5
$\Delta\omega$	0	0.049	3.613×10^{-4}	1.434×10^{-4}	4.082×10^{-4}	3.627×10^{-5}	0.295×10^{-4}	1.457×10^{-4}
ΔQ	0	0.003	0.734×10^{-4}	6.167×10^{-4}	3.239×10^{-4}	1.727×10^{-4}	0.047×10^{-4}	4.759×10^{-4}
Total error	0	0.026	$\mathbf{2.174 \times 10^{-4}}$	3.801×10^{-4}	3.661×10^{-4}	1.045×10^{-4}	$\mathbf{0.171 \times 10^{-4}}$	3.108×10^{-4}

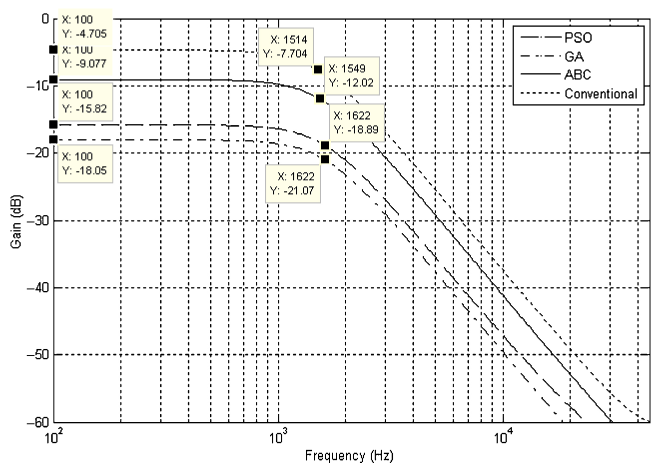

Fig. 18. Frequency responses of second order state variable low-pass filter (compatible with E24 series).

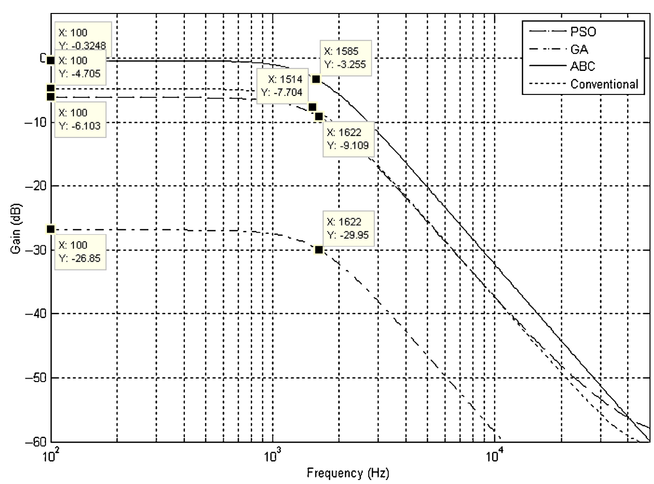

Fig. 19. Frequency responses of second order state variable low-pass filter (compatible with E96 series).

design are tabulated in Table XI. Considering SVF design task with components selected from E24 series, GA results in minimum total error value. However, execution time of GA-based design is the longest among others. ABC outperforms other EA techniques by means of accuracy and computation time when components are selected from E96 series.

In order to demonstrate the accuracy of EA-based SVF design methods, SVF is simulated with LM741 op-amp macro-model and E24/E96 compatible results of EA methods using SPICE. The frequency responses of SVF are given in Table X, and in Figs. 18 and 19 for E24 and E96 compatible results, respectively. Here, gain values are different since gain was unconstrained at the beginning of the design procedure. From SPICE simulation results, we clearly know that evolutionary approaches provide a maximally flat response and less cutoff frequency (-3 dB) deviation than conventional method.

VII. DISCUSSION AND CONCLUSION

The performances of EAs on analog active filter design have been investigated comprehensively. GA, ABC, and PSO algorithms were utilized for both fourth order Butterworth low-pass analog filter and second order SVF design and was investigated for the selection of passive components from different manufactured series by means of accuracy and execution time. Selection of the optimal own parameters is very crucial on minimizing total design error value thus effecting filter performance. Among the evolutionary approaches utilized in this paper, GA required more fine-tuning of the own parameters and increasing number of chromosomes decreased total error values at the cost of execution time. Considering ABC, increasing SL value facilitated obtaining better CF values when bee population was small. However, when bee population was crowded, the selection of bigger SLs decelerated the algorithm and worsened the performance. Increasing particle number in PSO improved the performance unless the acceleration factors were selected as 2.

Following this, we performed 20 runs with optimal own parameters obtained previously. The resulting CF values were used to produce box and whisker plots which shows that CF values obtained with GA method varies the most among the other methods. The iteration number required to achieve the quality restrictions are slightly different in each run for each method which can be seen in the plots of CF values versus iteration number for five different runs.

Considering Butterworth filter design with components selected from E12 series for a true comparison with [15], PSO achieved the smallest design error with respect to previous methods and other EA methods. Moreover, less design error is obtained with GA than the previously one used in [15].

Components of SVF are selected from two different manufactured series in order to investigate whether performance of EA methods will increase or not when same topology is designed with different series. GA algorithm achieved the smallest design error but the longest execution time when selecting components from E24 series. This is mainly due to the fact that PSO and ABC has fewer primitive mathematical operators than in GA (e.g., reproduction, mutation, and crossover). Those mathematical operations require more fine-tuning of own parameters. However, when tolerances of components became tighter in E96 series, ABC algorithm obtained the most successful results by means of both accuracy and execution time. Choosing optimal SL value avoids local minimum trapping, and increasing number of bee population improves the probability of converging to global minimum. Therefore, selecting optimal own parameters for ABC improves the accuracy as well as the convergence rate. Moreover, the performance of PSO was not affected significantly due to the usage of different manufactured series for same filter topology as other EA methods were. Therefore, it has been proved that the selection of components from different series had an influence on the performance of EA methods. ABC outperforms other methods by means of execution time for all design cases, however, when considering design accuracy, each method has achieved the smallest design error depending on the design case. From SPICE simulation results, we clearly know that the conventional method does not provide a maximally flat response in the passband and results in bigger cutoff frequency deviation. Moreover, SVF design with conventional method screens stop band ripples during frequency analysis unlike with EA techniques.

Automation of optimized analog circuits is a very challenging and time-consuming task. Design of analog filters with high accuracy and short execution time is successfully realized using evolutionary methods. Currently, we are working on utilizing these methods for other analog circuit topologies with particular design constraints. The most important target of this research is to develop an optimized synthesis methodology for analog circuit design based on EAs which can handle with all specified constraints in a reasonable execution time and high accuracy. An extension of this research might be the estimation of transistor sizes of analog circuits for newer semiconductor technologies using the present design knowledge obtained with older technology parameters.

References

[1] W. Wang, Y. Lu, J. Fu, and Y. Xiong, "Particle swarm optimization and finite-element based approach for microwave filter design," *IEEE Trans. Magnetics*, vol. 41, no. 5, pp. 1800–1803, May 2005.

[2] N. E. Mastorakis, I. F. Gonos, and M. N. S. Swamy, "Design of 2-D recursive filters using genetic algorithms," *IEEE Trans. Circuits Syst I*, vol. 50, no. 5, pp. 634–639, May 2003.

[3] N. Karaboga, "A new design method based on artificial bee colony algorithm for digital IIR filters," *J. Franklin Inst.*, vol. 346, no. 4, pp. 328–348, 2009.

[4] B. Luitel and G. K. Venayagamoorthy, "Differential evolution particle swarm optimization for digital filter design," in *Proc. IEEE Conf. Evol. Comput.*, Jun. 2008, pp. 3954–3961.

[5] R. S. Zebulum, M. A. Pacheco, and M. Vellasco, "Comparison of different evolutionary methodologies applied to electronic filter design," in *Proc. IEEE Conf. Evol. Comput.*, May 1998, pp. 434–439.

[6] A. Das and R. Vemuri, "An automated passive analog circuit synthesis framework using genetic algorithms," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, Mar. 2007, pp. 145–152.

[7] J. R. Koza, F. H. Bennett, III, D. Andre, M. A. Keane, and F. Dunlap, "Automated synthesis of analog electrical circuit by means of genetic programming," *IEEE Trans. Evol. Comput.*, vol. 1, no. 2, pp. 109–128, Jul. 1997.

[8] S.-J. Chang, H.-S. Hou, and Y.-K. Su, "Automated passive filter synthesis using a novel tree representation and genetic programming," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 93–100, Feb. 2006.

[9] Y. Sapargaliyev and T. Kalganova, "Constrained and unconstrained evolution of LCR lowpass filters with oscillating length representation," in *Proc. IEEE Congr. Evol. Comput.*, Sep. 2006, pp. 1529–1536.

[10] J. Hu, X. Zhong, and E. Goodman, "Open-ended robust design of analog filters using genetic programming," in *Proc. Genet. Evol. Comput. Conf.*, 2005, pp. 1619–1626.

[11] C. Goh and Y. Li, "GA automated design and synthesis of analog circuits with practical constraints," in *Proc. IEEE Congr. Evol. Comput.*, vol. 1, no. 1. 2001, pp. 170–177.

[12] A. F. Sheta, "Analogue filter design using differential evolution," *Int. J. Bio-Inspired Comput.*, vol. 2, nos. 3–4, pp. 233–241, 2010.

[13] R. S. Zebulum, M. A. Pacheco, and M. Vellasco, "Artificial evolution of active filters: A case study," in *Proc. 1st NASA DoD Workshop Evol. Hardware*, 1999, pp. 66–75.

[14] H. Xu and Y. Ding, "Optimizing method for analog circuit design using adaptive immune genetic algorithm," in *Proc. Int. Conf. Frontier Comput. Sci. Technol.*, 2009, pp. 359–363.

[15] M. Jiang, Z. Yang, and Z. Gan, "Optimal components selection for analog active filters using clonal selection algorithm," in *Proc. ICIC-I*, LNCS 4681. 2007, pp. 950–959.

[16] R. A. Vural and T. Yildirim, "Component value selection for analog active filter using particle swarm optimization," in *Proc. 2nd ICCAE*, vol. 1. 2010, pp. 25–28.

[17] D. H. Horrocks and M. C. Spittle, "Component value selection for active filter using genetic algorithms," in *Proc. IEE/IEEE Workshop Nat. Algorithms Signal Process.*, vol. 1. Nov. 1993, pp. 1–6.

[18] A. Kalinli, "Optimal circuit design using immune algorithm," in *Proc. ICARIS*, LNCS 3239. 2004, pp. 42–52.

[19] A. Kalinli, "Component value selection for active filters using parallel tabu search algorithm," *AEU Int. J. Electron. Commun.*, vol. 60, no. 1, pp. 85–92, Jan. 2006.

[20] R. A. Vural and T. Yildirim, "State variable filter design using particle swarm optimization," in *Proc. 11th Int. Workshop SM2ACD*, Oct. 2010, pp. 1–4.

[21] R. Schaumann and M. V. Valkenburg, *Design of Analog Filters*. New York: Oxford Univ. Press, 2001.

[22] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*. Chichester, U.K.: Wiley, 2007.

[23] J. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. Michigan Press, 1975.

[24] D. E. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*. Reading, MA: Addison Wesley, 2005.

[25] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Global Optimiz.*, vol. 39, no. 3, pp. 459–471, 2007.

[26] D. Karaboga and B. Akay, "Artificial bee colony (ABC), harmony search and bees algorithms on numerical optimization," in *Proc. Innovative Product. Mach. Syst. Virtual Conf.*, 2009, pp. 417–422.

[27] R. C. Eberhart and J. Kennedy, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Nov.–Dec. 1995, pp. 1942–1948.

[28] M. Clerc, "The particle swarm: Explosion, stability and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, vol. 6, no. 1, pp. 58–73, Feb. 2002.

[29] T. Kiink, J. S. Vesterstroem, and J. Riget, "Particle swam optimization with spatial particle extension," in *Proc. lEEE Congr. Evol. Comput.*, May 2002, pp. 1474–1479.

[30] L. D. Paarman, *Design and Analysis of Analog Filters*. Norwell, MA: Kluwer, 2007.

[31] K. Lacanette, "A basic introduction to filters-active, passive and switched-capacitor," Nat. Semicond., Santa Clara, CA, Applicat. Note 779, Apr. 21, 2010.

Revna Acar Vural (S'09) received the B.S. and M.S. degrees in electronics and communication engineering from Yildiz Technical University, Istanbul, Turkey, in 2002 and 2004, respectively. She is currently pursuing the Ph.D. degree in artificial intelligence for electronic circuit design automation.

Her current research interests include design optimization, automated synthesis, analog integrated circuit design, and evolutionary algorithms.

Tulay Yildirim (M'08) received the B.S. and M.S. degrees in electronics and communication engineering from Yildiz Technical University (YTU), Istanbul, Turkey, in 1990 and 1992, respectively, and the Ph.D. degree in electrical and electronics engineering from the University of Liverpool, Merseyside, U.K., in 1997.

Currently, she is a Full Professor with YTU. Her current research interests include analog and digital integrated circuit design, hardware implementations of neural networks, medical electronics, and artificial intelligence.

Tevfik Kadioglu received the B.S. and M.S. degrees in electronics and communication engineering from Kocaeli University, Kocaeli, Turkey, in 2006 and 2010, respectively.

He is currently a Hardware Design Engineer with the National Electronic and Cryptography Research Institute, Istanbul, Turkey. His current research interests include discrete analog circuit design, noise in analog circuits, signal processing, and active filter design.

Aysen Basargan received the B.S. degree in electronics and communication engineering from Yildiz Technical University, Istanbul, Turkey, in 2006. She is currently pursuing the M.S. degree in electronics engineering from Istanbul Technical University, Istanbul.

Her current research interests include analog integrated circuit design and evolutionary algorithms.