An incremental evolutionary algorithm suitable for industrial platform for analog circuit design

Yuehan Gao
Department of Electronic Science and Technology
University of Science and Technology of China
Hefei, China

Jingsong He
Department of Electronic Science and Technology
University Science and Technology of China
Hefei, China

***Abstract*—The design of analog circuits by evolutionary algorithm is an approach to automatic design of analog circuits. Applying the method of automatic analog circuit evolutionary design in engineering application has practical values. However, it also results in a lot of time-consuming. The purpose of this paper is to deal with the problem of time-consuming. We propose an incremental evolutionary algorithm to reduce the time complexity during the evolution. The algorithm improves the precision of simulation incrementally. A low-pass filter from the practical problem is designed by the incremental evolutionary algorithm. And experimental results show that, the algorithm is remarkable effect in the evolutionary design of industrial analog circuits.**

Keywords-incremental environary algorithm; precision control; analog circuit design; industrial simultor

I. INTRODUCTION

Since the world is fundamentally analog in nature, analog circuits are of great important in electronic system design [1]. And analog circuits have various applications in communications and other fields. While analog circuits require experienced designer to design by hand. Moreover, manual design takes lots of time, manpower and material resources [10]. Consequently, designing the analog circuits automatically has become an important problem.

In the research of analog circuit evolutionary design, evolutionary algorithm and circuit simulation are used to evolve circuits. The majority of evolutionary algorithm they used is Genetic Programming (GP) [4] [8], Genetic Algorithms (GA) [5] [6], Differential Evolution (DE) [2], Evolutionary Strategies (ES) [9] [7]. Some circuit simulators including such simple models as winspice are adopted in the study of automatic analog circuit evolutionary design [6]. The simulator with simple models is mainly used to study the method of automatic analog circuit design, instead of applying the analog circuit design to industrial production directly. In this paper, we use Cadence as the evolutionary platform. Cadence is a professional tool of EDA (Electronic Design Automation) and its simulator with the name of spectre contains the most advanced models of the technology library. Therefor studying analog circuit evolutionary design in Cadence has practical values in industrial application.

When evolving the analog circuits in Cadence, a lot of time-consuming is obvious, and the time complexity is very significant. Through analyzing the time complexity of evolutionary algorithm, time complexity is mainly derived from the simulation in the process of evaluating the fitness. We simulate the same circuit with different simulators. It's easy to find out that using industrial simulator takes 22 times of time-consuming than the common simulator. For an automatic circuit evolutionary design of a population size of 30 and a generation of 100, using industrial simulator takes 66000 times of time-consuming than the common simulator. Due to the high time complexity of evolving analog circuits with industrial simulator, we propose an incremental evolutionary algorithm. The basic idea of this algorithm is improving the precision of the simulator gradually in the process of evolution. In other words, use low precision in the primary stage of simulation, reduce the scale of data to decrease the time-consuming and the excepted circuit will be simulated in high precision.

The remainder of this paper is organized as follows. The time complexity of analog circuit evolutionary design in industrial platform is significant. Then we explore the origin of time complexity of analog circuit evolutionary design in Cadence. And an incremental evolutionary algorithm is proposed to reduce the time complexity. This algorithm combines the alternant parameters of the simulator with the evolution of the algorithm. The time complexity is reduced by controlling the precision of the simulator to change the scale of data. Finally, the experiment of evolving the filter from practical problem is to verify that the incremental evolutionary algorithm is remarkable effect in the evolutionary design of industrial analog circuits.

II. THE INCREMENTAL EVOLUTIONARY ALGORITHM

A. Algorithm Description

We aim to guarantee and control the precision and reduce the time complexity at the same time. Then we adjust the precision of simulation by controlling the sampling points of simulation. The precision is higher with more sampling points and controlled incrementally. Namely, low precision is enough when in the early stage the general direction of the evolution is needed to be determined. When the precision is gradually higher, the amplitude-frequency curve is more detailed to approach the expected target circuit during the evolution. Differential Evolution (DE) is adopted to evolve the analog circuits, and the specific steps of the algorithm are as follows:

a) Initialization: Initial the population P(t) (The size of the population is NP)

b) Evaulate the individuals in the first generation: Simulate every individual of the population P(t), the number of the sampling points is P, the values of (frequency(Hz),voltage(V)) are the results of the sampling points , namely: (frequency[i],Vout[i]); subtract the target voltage S from the voltage Vout[i] of each sampling point and get E(i), multiply the E(i) by a coefficient of the penalty w and the equation of the fitness is as below:

$$\text{fitness}=\sum_{i=0}^{p} E(i)*w \tag{1}$$

If the low-pass filter will be evolved, the transition zone is (f1,f2), the differential values of the expected voltage and the target voltage between both sides of the transition zones are $\delta 1$ and $\delta 1$:

$$E(i)=|Vout[i]-S| \tag{2}$$

The target voltage S is different due to passband and stopband.

$$\text{frequency}[i]\le f1, w=\begin{cases} 1, & E(i)\le\delta 1 \\ 10, & E(i)>\delta 1 \end{cases} \tag{3}$$

$$\text{frequency}[i]\ge f2, w=\begin{cases} 1, & E(i)>\delta 2 \\ 10, & E(i)\le\delta 2 \end{cases} \tag{4}$$

c) Crossover and mutation: New generation is P'(t), it's produced by mutating and crossing each circuit of P(t) (the specific means of mutation and crossover are mutating and crossing the Individual v(i) (which iϵ(1,NP)), that is, variation individual is synthesized by two individuals selected from the P(t) randomly and the variation individual which is ready to be mutated; the new individual is new(i) and it's generated by crossing the variation individual and the individual is v(i)).

$$m(i)=x3+F*(x1-x2) \tag{5}$$

$$new(i)=(new_1(i),new_2(i)\ldots\ldots new_D(i)) \tag{6}$$

$$new(j)=\begin{cases} m_j(i), & \text{if rand}(j)\le CR \\ v_j(i), & \text{if rand}(j)>CR \end{cases} \tag{7}$$

Among them, j=(*1*,*2*……*D*),x*1* 、 x*2* 、 x*3* are selected randomly from v(i) and x*1*≠x*2*≠x*3*, m(i) is the variation individual, the scaling factor is Fϵ(*0*,*1*), the crossover rate is CRϵ(*0*,*1*)

d) Evaluate the generation: simulate the circuits in P'(t) and the precision of the simulator's parameter (dec) is increased with the evolution.

$$dec=30*\left|\frac{gen}{20}\right|+10,\text{"}|.|\text{means rounding"} \tag{8}$$

The number of sampling points is P:

$$p=(lgN-lgM)*dec+1 \tag{9}$$

where the range of the sampling frequency is (M,N)Hz, and (9) is calculated by experimental analysis. With taking the simulation results of controlled precision into the fitness equation (1), the circuit is estimated.

e) Section: The individuals in P(t) and P'(t) are arranged in ascending order according to the values of fitness and choose the first NP individuals as a new generation, P(t+1).

f) Stop the evolution if the target generation is reached. Otherwise turn to *c)* and update the generation P(t) with P(t+1).

Equation (8) is the fundamental part of the incremental algorithm. The parameter of precision is not an infinite value. The two cases are as follows. 1) When the generation of evolution is appropriate, the precision is increased as the equation (8). 2) When the generation of evolution is large, then a maximum value is set for the precision. The value of the precision won't increase with the larger generation until it reaches the maximum value. And the constant parts of (8) originate from experience.

B. Implement of precision control

In the study of automatic analog circuit evolution, simulator is used with evolutionary algorithm in the design of evolving analog circuits. In the study, simulator only privates results of the simulation. However, in the incremental evolutionary algorithm the parameter of simulator is added into the evolution. The precision is adjusted incrementally in the process of the evolution, at the same time the number of sampling points is increased. In the first stage of evolution, scale of sampling data is reduced by low precision of simulation to guide the direction of circuit evolution. In the latter stage, performance of circuits is described in detail with high precision and the evolutionary circuit is closer to the target circuit. This incremental evolutionary algorithm with precision control is mainly applied to the problem of circuit evolution using industrial simulator and the industrial simulator needs consuming a larger amount of time.

The most important part in incremental evolutionary algorithm is adjusting the precision of simulation, namely the equation (8). We turn manually adjust to automatically adjust to realize the precision control in algorithm. Manual processes of simulating circuits in Cadence are shown in Figure1: firstly, create the schematic and verify its correctness; then set the environmental parameters in the environment of spectre, such as: the type of simulation, the type and range of sweep variable, the parameter of precision, the output and so on; the results of simulation appear after clicking the button of start. The automatic process is achieved by SKILL. SKILL is developed by Cadence and is based on LISP-like language, most of the characteristic and applications in cadence design environment are written in SKILL language. The process of evaluating the fitness incrementally is as follows: turn the individual circuit to the netlist file which can be identified by spectre; then import this netlist file in spectre and set the precision in the simulated environment (The relationship between precision and generation is equation (8).); finally, calculate the fitness of the individual circuit with the output values from the fitness equation, where output values are (frequency, voltage). In the entire process of evaluating fitness, the scale of sampling data,

the time-consuming and the time complexity are reduced with the incremental precision.

III. THE EXPERIMENT

The emphasis of the incremental evolutionary algorithm is adjusting the precision of simulation gradually in the period of evolution. The scale of sampling data is reduced in low precision stage. Time-consuming of the entire evolution is less while the circuit is simulated with high precision. We compare the experiments of incremental evolutionary algorithm with the ones of no precision control (the precision is a constant during the evolution videlicet). We make the comparative experiments to reflect the effectiveness of the incremental evolutionary algorithm. The experiments of no precision control are experiment of high and low precision.

In addition to the different precision of simulation, the other parameters of the comparative experiments are the same. We use DE (differential evolution) to evolve the filter from practical application. The source voltage of the passive low-pass filter is 1V, the passband and stopband are 0.8912GHz (3db) and 1GHz (60db). Then the equation of the fitness is as shown:

$$E(i)=\begin{cases}|Vout[i]-0.5|, & \text{for passband } S=0.5\\ |Vout[i]-0|, & \text{for stopband } S=0\end{cases} \quad (10)$$

$$\text{frequency}[i]\le 0.8912\text{GHz},\ w=\begin{cases}1, & E(i)\le 0.1465\\ 10, & E(i)>0.1465\end{cases} \quad (11)$$

$$\text{frequency}[i]\ge 1\text{GHz},\ w=\begin{cases}1, & E(i)>0.495\\ 10, & E(i)\le 0.495\end{cases} \quad (12)$$

(The filter of closely practical and commercial application is a kind filter of wider broadband. Its frequency range is wide with quantities of sampling points. And the large scale of simulating data results in a lot of time in simulation.) In the

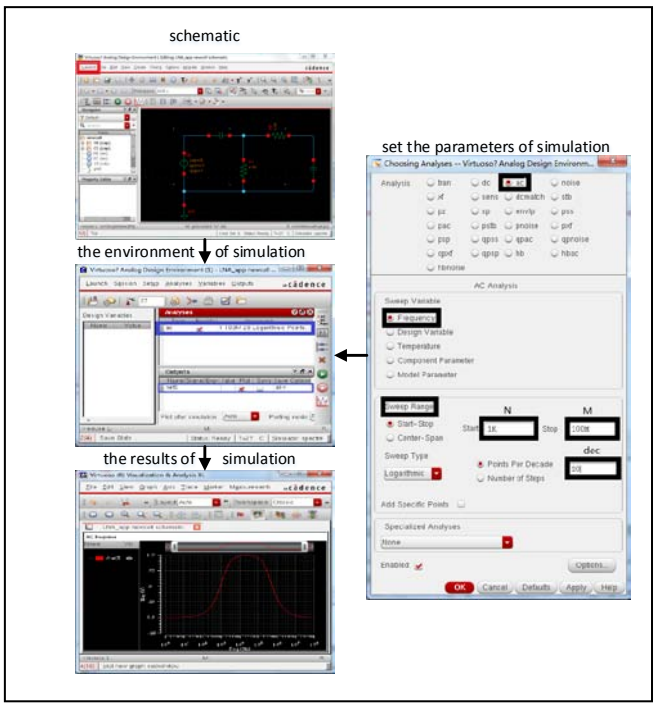

Figure 1. The process of simulating circuits with spectre

TABLE I. THE SAME PARAMETRES OF EVOLUTION

Population size	*Gene-ration*	*Frequency range*	*Cross rate*	*Mutation rate*	*Process*
40	*80*	*1K-100GHz*	*0.8*	*(0.5,1)*	*0.18um*

TABLE II. THE PARAMETERS IN 0.18UM

Componment	***Capacitor value range***	0.01-20nH
	Inductor value range	0.01-30nH

TABLE III. THE TIME-CONSUMING COMPARISON OF DIFFERENT EVOLUTIONARY ALGORITHM

Type of precision control	*No precision control (low precision: dec=10)*	*No precision control (high precision: dec=100)*	*Incremental evolutionary algorithm($dec=30*\frac{|gen|}{20}+10$)*
Time(s)	*74858*	*11912*	*79023*

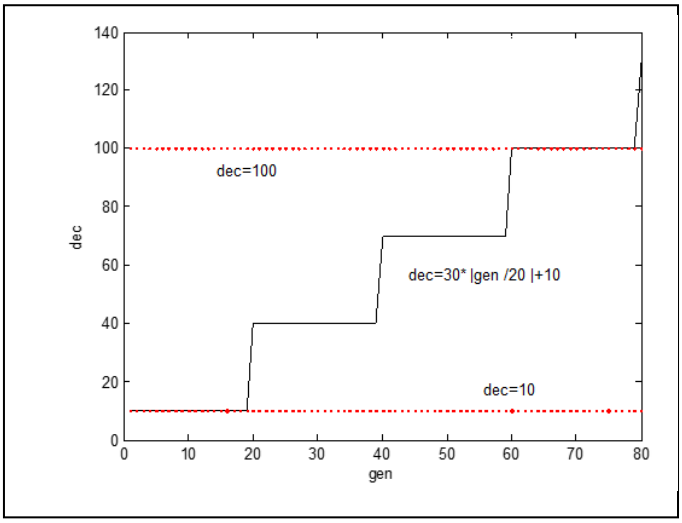

Figure 2. The ralationship with precision and generation

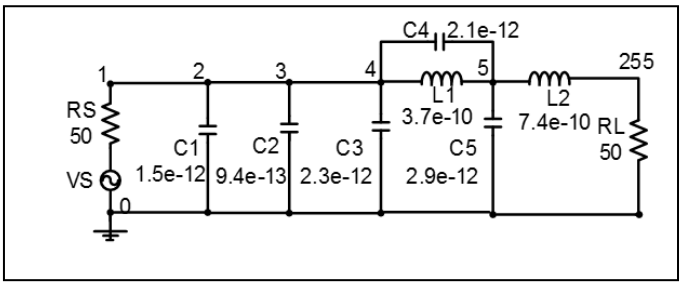

Figure 3. The diagram of evolved final circuit

experiment of incremental evolutionary algorithm with precision control, the equation of precision and generation is shown in Figure2. We set the step 20, and the step is the represent of increment. The value of the step will change with different evolutionary problems.

The consuming time of evolving 80 generations to get the target circuit is shown in Tab.III. And the time is the average

time of several experiments. Results indicate that the incremental evolutionary algorithm is effective. Time complexity is reduced by less scare of simulation data in the algorithm. The time is the same order of magnitude in the experiment of the incremental evolutionary algorithm and the algorithm with low precision. In this paper, we use an experimetal control to adjust the precision. So the experimental control isn't the best scheme. However, the incremental evolutionary algorithm has solved the time complexity in evolving analog circuits on industrial platform like Cadence. The algorithm is more beneficial to apply the study of analog circuit evolution to practical application.

IV. CONCLUSION

In this paper, we focus on the study of practical application about the method of evolving analog circuits automatically with evolutionary algorithm on industrial platform. The simulator of industrial platform includes advanced process and high-precision. Unfortunately, lots of time is spent on simulating practical circuits. However, little research has been conducted about time complexity of analog circuit evolution in recent study of analog circuit evolutionary method. We propose the incremental evolutionary algorithm due to the plenty of consuming time. The number of sampling points and the scale of simulation data are modified by incremental precision to reduce the consuming time. In this paper, the experiment of designing filter from practical problem verifies the effectiveness of the algorithm we propose. The incremental evolutionary algorithm has significant effect on analog circuit evolution to industrial application. The algorithm reduces the time complexity by precision control on industrial platform. And the key problem of applying the analog circuit evolutionary method to practical industrial process is solved.

ACKNOWLEDGMENT

Corresponding author: Jingsong He (E-mail: hjss@ustc.edu.cn). This work was supported by the National Natural Science Foundation of China through Grant No.61273315.

REFERENCES

[1] Lohn J D, Colombano S P. A circuit representation technique for automated circuit design[J]. Evolutionary Computation, IEEE Transactions on, 1999, 3(3): 205-219.

[2] S. L. Sabat, K. S. Kumar, and S. K. Udgata, "Differential evolution and swarm intelligence techniques for analog circuit synthesis," Nature \& Biologically inspired Computing, 2009, pp. 469-474.

[3] J. Yuan, N. Farhat, J.V. der Spiegel, "GBOPCAD: a synthesis tool for high performance gain-boosted opamp design," IEEE Transactions on Circuits and Systems i, Fundamental Theory and Applications, vol. S2, pp. 1535-1544,2005.

[4] Koza J R, Bennett III F H, Andre D, et al. Automated synthesis of analog electrical circuits by means of genetic programming[J]. Evolutionary Computation, IEEE Transactions on, 1997, 1(2): 109-128.

[5] Goh C, Li Y. GA automated design and synthesis of analog circuits with practical constraints[C]//Evolutionary Computation, 2001. Proceedings of the 2001 Congress on. IEEE, 2001, 1: 170-177.

[6] Grimbleby J B. Automatic analogue circuit synthesis using genetic algorithms[C]//Circuits, Devices and Systems, IEE Proceedings-. IET, 2000, 147(6): 319-323.

[7] Sapargaliyev Y, Kalganova T. Constrained and unconstrained evolution of "LCR" low-pass filters with oscillating length representation[J]. 2006.

[8] Koza J R, Bennett III F H, Andre D, et al. Automated design of both the topology and sizing of analog electrical circuits using genetic programming[M]//Artificial Intelligence in Design'96. Springer Netherlands, 1996: 151-170.

[9] K. J. Kim, A. Wong, and H. Lipson, "Automated synthesis of resilient and tamper-evident analog circuits without a single point of failure," Genet Program Evolvable Mac, vol. II, pp. 34-59, 2010.

[10] Xue M, He J. Towards real application: A development in Cadence design environment for evolutionary analog integrated circuit design[C]//Advanced Computational Intelligence (ICACI), 2013 Sixth International Conference on. IEEE, 2013: 159-164.