ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
Indian Journal of Science and Technology, *Vol 9(39), DOI: 10.17485/ijst/2016/v9i39/97980, October 2016*

Towards Analog Design Automation using Evolutionary Algorithm: A Review

A. Sasikumar and R. Muthaiah

School of Computing, Information Communication Technology, SASTRA University, Thirumalaisamudram, Thanjavur - 613401, Tamil Nadu, India;
vlsisasi@gmail.com, muthaiah66@gmail.com

Abstract

Analog circuits are the most important parts in many Integrated Circuit (IC) design. This paper reviews the basic concepts in analog design automation using evolutionary algorithm. Analog design problem is a multi objective problem; this can be solved by Evolutionary computation methods. Computation methods provide the set of feasible solutions for the optimal circuit design of analog integrated circuits. It is necessary to integrate both analog and digital in a single chip for real world communication. Due to system level integration we need analog design automation tool for IC design. This paper summarized recent start of art in analog optimization and also lists the survey of main people working in this field. Finally, we listed several open research problem to improve the analog design automation for analog IC using evolutionary computation.

Keywords: Analog Design Automation, Analog Integrated Circuits, Evolutionary Computation, Multi Objective

1. Introduction

Automated design of analog ICs is becoming a viable solution for complex analog components. Over the past decade, more number of research focusing on analog design automation. Automated techniques for layout, device sizing and basic design centering have been successfully applied. A complete survey of the area described in[1]. However, the transistor level does not scale successful to system level ICs. Normally the analog circuit has only 100 devices, in case of small system like a analog to digital converter, or a phase-locked loop, or an entire RF circuit might requires 1000 to 10000 devices or more.

Typical analog circuits needed to optimize more number of continuous values for system performance. These continuous values are nonlinear signal. It is hard to cover all specification range across the circuits to systems. This system design successful depends on the circuit's designer ability. Such designs are being integrated in large system-on-chip (soc) environments is challenging process. It is difficult to verifying, optimizing, and synthesizing such complex system when they are considered flat. This difficult design must be solved by hierarchical tools that deal with the system design parameters[2].

Traditional optimizations for single objective function do not allow multiple objectives; and also this optimization does not give freedom to choose best among different solutions, and equally best solutions. These drawbacks can be overcome by multi objective optimization. This optimization technique allow multiple objective to be treated as single and iteratively during the optimization process[3].

The multi objective optimization algorithm used to provide an optimal solution or pareto front (non dominated individuals). The generation of pareto front can be more expensive and it does not provide the feasible solution. This drawback can be overcome by stochastic method (Simulated Annealing, Evolutionary Computation, and

*Author for correspondence

Ant Colony Optimization). Evolutionary computation methods are the best optimization method to provide good result on complex problems without explicit knowledge[4].

2. Automatic IC Design Automation

Analog design automation using optimization method divided into two groups. The two group of methods are knowledge based method, optimization based method[5]. Figure 1 illustrates the fundamental techniques of analog design automation.

2.1 Knowledge Based Method for Analog Circuit Optimization

The starting analog design automation system tool only depends on analog designer. The analog designer design a complete plan for particular system and this system did not use any optimization techniques[6–10]vol. 14, no. 2, p. 218-38 (1995. In Knowledge based method; a complete plan consists of system design equations and the parameter sizes that meet the performance requirements. This knowledge based method only used for simple circuit system and did not suitable for complex system applied with moderate success. The execution time of small system is shot but deriving the design plan is hard for small system and time-consuming. In this method the design equation needed to update with technology parameter. One more drawback is the results are not feasible, it suitable for small system.

The next method of analog IC design tool is optimization based tool. Optimization method applied to analog IC design to overcome drawbacks of knowledge-based methods. Optimization based method divided into two types: equation based and simulation based method. These methods used for analog IC design optimization.

2.2 Equation Based Method for Analog Circuit Optimization

Here the analytical equations used to express the circuit parameters and design variables. These equations can be solved by either deterministic or stochastic optimization techniques. Normally equation based circuit sizing is solved by classical optimization methods. In OPASYN[11], the steepest descent method is used for optimization of analog circuit sizing; similarly, in STAIC the equations are solved by refinements technique[12].

The analog circuit sizing problem converted into constrained nonlinear optimization problem. It can be solved

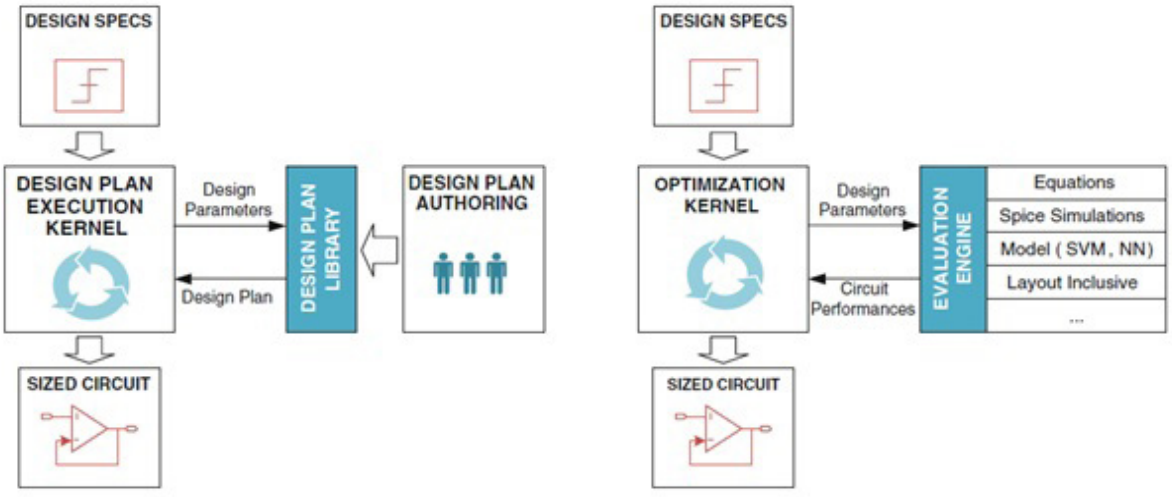

Figure 1. Analog circuit design optimization types; (a) knowledge based method; (b) Optimization based method.

by sequential quadratic programming[13,14]. The convex optimization used for design of delta-sigma and pipeline analog to digital converters. The convex equations relate the input parameters and circuit performance[15].

GPCAD[16], the Geometrical Programming (GP) used to optimize a posynomial circuit model. The execution time of this model is in the order of few seconds, but it is very difficult to design a posynomial model and also it takes more time to create model for new circuits. Posynomial modeling were revisited, accuracy issue is solved by additional generation steps[17]. These additional steps are added to Simulated Annealing (SA) for solving local optimization problem. The same method is applied to solve the analytical expressions, these expression used to generate an initial solution[18]. Simulation based methods provide the feasible solution for analog circuit design.

The problem formulation of equation based method is not limited one. Problem formulation used for selection of optimization techniques and also it relying on heuristic optimization. In OPTIMAM [19]the SA applied to analytical models for solving analog circuit sizing problem. ASTRX/OBLX[20], an SA is also applied to analytical models for solving cost functions (i.e., dc operating point of analog circuits). This model can be by Asymptotic Waveform Evaluation (AWE) based simulation tools. In DARWIN instead of SA the Genetic Algorithms (GA) used to solve the analytical models. Doboli[21] the sub-block specifications, transistor sizing and sub-block topology selection are simultaneously derive by genetic programming techniques.

The evaluation time of equation based method is less, like knowledge based method and also it is extremely suited for small circuit design. The analytical equation did not capture all design characteristics of analog circuit. It is difficult to make generalization of analytical equation for different analog circuits. One more drawback is that the equation approximation introduced low accuracy design for complex analog circuits. The approximation equation also needed additional evaluation time to ensure that the circuit specification.

2.3 Simulation Based Method for Analog Circuit Optimization

Simulation based approach is the most common method found in analog circuit sizing. In this method the circuit simulator used to evaluate the circuit performance (for example AIDA-C is a circuit simulator).

The previous methods of simulation based automatic sizing uses local optimization method for solving circuit optimization problem. SA[22] is the most common method for solving local optimization problem. In DELIGTH. SPICE[23], the starting point of local design is obtained from the optimization algorithm. FASY[18] and Kuo-Hsuan[17] an approximate solution is derived from equation-based technique, then the optimization kernel used to simulate the circuit operation. The Simulated Annealing optimization algorithm to finding bias value of the transistor instead of transistor sizes[24]. FRIDGE[25] uses annealing like optimization technique for starting point of design. A deterministic and SA method used for fine-tuning to perform the optimization[26].

The next important type of optimization methods is the Genetic Algorithm (GA). Optimization of circuit parameters are supported by GA[5,27]. The population evaluations were selected based on support vector machine and circuit simulator. The combination of evolutionary and annealing optimization used for circuit performance figures and simulations[28].

In[29–31] introduce the parallel implementation of evolutionary algorithm to reduce the simulation time. The parallel mechanism used to share the computation load among multiple devices. Especially MAELSTROM follows the hybrid method. Variation pattern search algorithm used in ANACONDA simulation approach instead of PRSA. The pattern search named stochastic pattern search.

Circuit sizing optimizations use a different method to employs evolutionary computation for devices parameter and circuit design variables (i.e., circuit topology generation). Koza, Sripramong[32,33] Hongying[34] create new topologies using proposed design methodology that explore the search space starting from low abstraction level. A new topology is formed by basic elementary blocks. These blocks are connected in bottom-up approach. The proposed design method consists of various fundamental entities such as single transistor sizing, elementary topology block, and circuit connection. This method needed a huge simulation to generate circuit structure because they often include more number of variables for analog circuit.

Analog circuit sizing optimization can also solved by swarm intelligence algorithms. Many simple agents used as fundament of swarm intelligence algorithms. The ant colony optimization (ACO) used for circuit sizing[35,36] and particle swarm optimization (PSO) applied for circuit optimization[37,38–40]. These two algorithms are the successful common method of swarm intelligence algorithm.

Normally the analog circuit optimization problem is consists multi objective and multi constrained. The tradeoff among the performance measures can be solved by multi objective optimization method. For example minimizes the power with maximum bandwidth or minimizes area with maximum gain of an amplifier. The multi-objective optimization techniques are common optimization method for amplifier design. The output of multi objective optimization consist set of optimal tradeoff solutions. This optimal set is also called as Pareto Front (PF). Both evolutionary and swarm intelligence algorithms follows the multiple element approach. In GENOM[9] and MOJITO[41], the evolutionary algorithm with multi objective is used for circuit parameter and design variable. Particle Swarm Optimization (PSO) implemented with single objective as well as multi objective circuit parameter[40]. Pradhan and Vemuri[42], the analog circuit optimization can be solved by multi-objective simulated annealing (MOSA).

The non-dominated solutions are generated instead of existing approaches to the design variables. The old approaches follows the multi objective optimization without non-dominated variables[43,44]. The survey of analog circuit sizing and optimization approaches, it is clear that there is no single specific algorithm for automatic circuit sizing.

3. Problem Formulation in Analog Circuit Optimization

The crucial part of Mixed-Signal IC design is analog because design of analog part is very difficult. Analog signals are nonlinear and continuous; due to this nonlinear behavior of circuit performance it is necessary to measure the input parameters. Sensitivity of design parameters calculated from circuit performance measures. Mathematical representation of analog design automation problem using general nonlinear programming (NP) is given by

$$\text{Minimize } F(x)$$
$$\text{Constraints: } \Omega = \{x \pounds R^n ! \; G(x) \leq 0\}$$

Here, x denotes multidimensional decision parameters, R^n denote decision parameters with lower and upper limits are given by $x^i_{min} \leq x^i \leq x^i_{max}$. The function F(x) denotes the m number of objective ($F_1(x)$... Fm (x)). The function G(x) represents the of p-number of constraints. The same mathematical representation also used for single objective problem. For single objective problem the value of m equal to one. The G (x) constraints are represented in terms of inequality expressions. These inequality expression used to represent correct value of problem specifications, i.e., $G_i(x) \leq \text{Specifications}_i$ or $G_j(x) \geq \text{Specifications}_j$ or $G_k(x) = \text{Specifications}_k$ with respect toi+j+k =p. the inequality constraints are transformed into equality constraints using following expression $|G_k(x)| - \pounds \leq \text{Specifications}_k$, where the small tolerance denoted by £. The tolerance values are used for every time of inequality constraints. The Domain space (Ω) normally represented by S (Nonempty set).The S set in R^n, the objective functions are expressed by $f_i : R^n \rightarrow R$.

For each design estimation consists of single or multiple objective functions with global, high dimensional optimization search problems are represented by multiple constrains. For example the circuit optimization variables are Area, Power, Circuit Gain, Phase Margin etc. The design parameters are Transistor Width, Length, Resister values, Capacitor values etc. A particular value of the design parameters ($x_1, x_2, \ldots, x_n$) belonging to Ω calculated by the undertaken design problem. This domain space used to select the set of optimal points for the objective function and also this function satisfied the required specification. The specifications are shown in Figure 2.

Optimization method used to find the global optimum solution for the high-dimensional vector problem. This method further classified into knowledge based and optimization based approach. The knowledge based approach form a design equation for circuit performance. The optimization based problem solved by numeri-

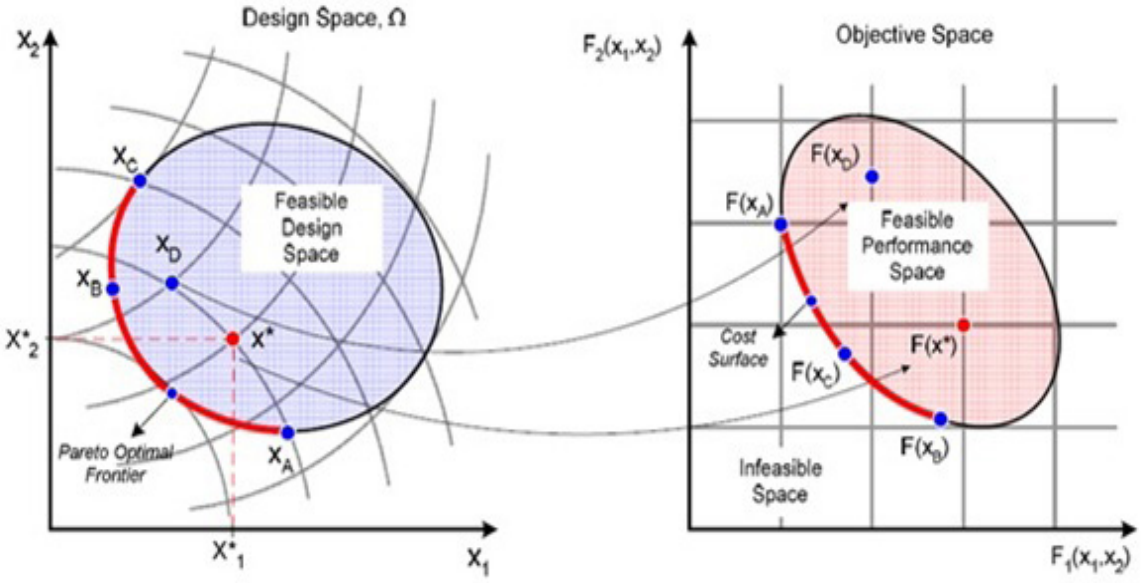

Figure 2. Multi objective concepts used in optimization problem.

cal programming techniques. The recent research work focusing on optimization-based approach. Next subsection reviews the common algorithm technique used in optimization approach.

3.2 Numeric Programming Techniques for Optimization

Optimization algorithms play a vital role for the past few years in the field of both research community and the industry. Optimization algorithms used to achieve approximate solutions for NP-hard [45]problem with high efficiency. Optimization algorithms become an important method to provide solution for nonlinear differential equations without applying any analytic methods. The main objective of scientific and industrial design is to minimize or maximize the functions through an optimization algorithm. The vast range of optimization method available in this field of numerical optimization is shown in Figure 3.

The optimization method is suitable for circuit design parameter selection. Therefore, the optimization algorithm selection is depends on the knowledge of problem nature. In case of, linear programming techniques (LP) are more suitable for finding a solution of linear equations. George Dantzig[46] developed a simplex algorithm for analog circuit optimization. Sometime these kinds of problem have equality constraints. Using addition of slack variables the inequalities can be mathematically converted into equalities[45].

The challenging task of nonlinear programming problem is to determine the global optimum solution. There is no systematic way for specific method to solve global optimum[45]. The most common methods used in optimization techniques are random search. The random search method used for nonlinear optimization problem. Some other search methods for nonlinear optimization are gradient-based methods, constraints programming, and stochastic methods. These optimization methods are suitable for solving single and multi objective problems.

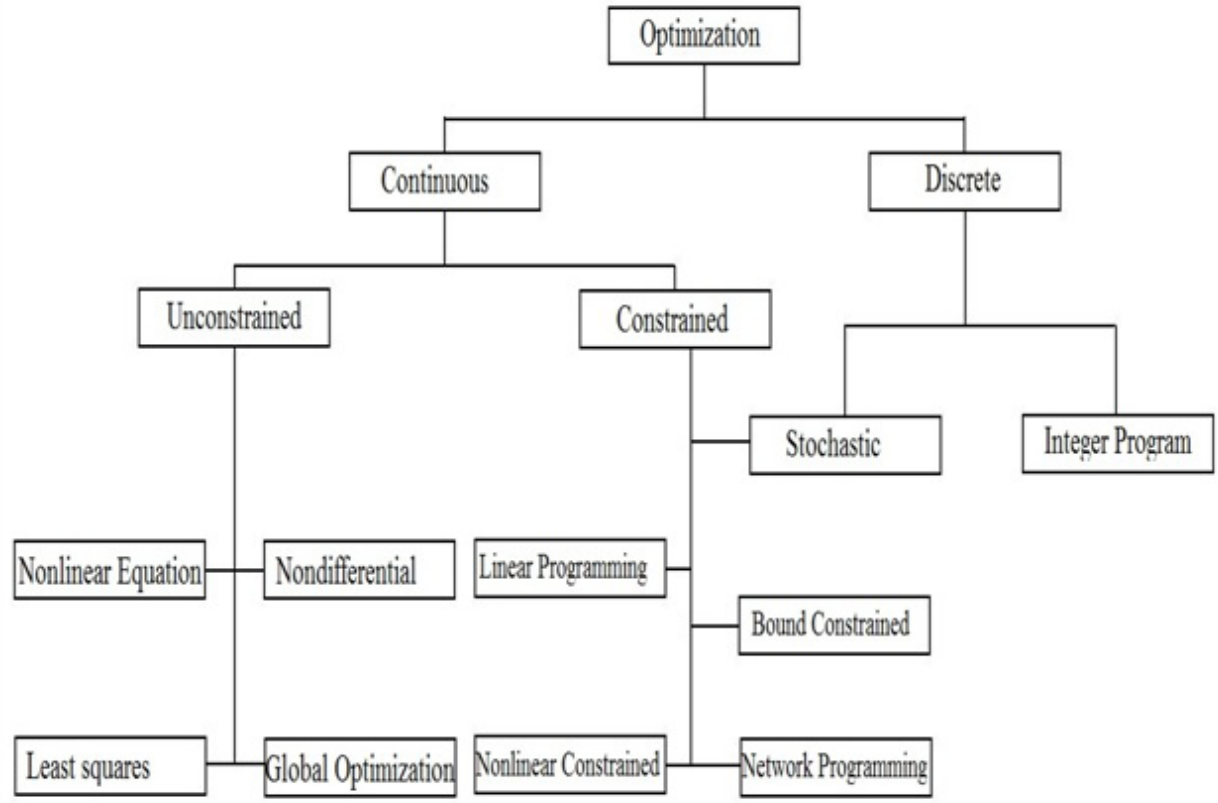

Figure 3. Basic types of optimization algorithm used in analog circuit optimization.

Table 1. Optimization types and their properties

Optimization method type	Type of problems and description	Advantages	Limitations
Random search	Global unconstrained. Randomly select potential solution and evaluating them.	It follows heuristic search method. The random search used as reference tool (Monte Carlo Method tool).	It is general specific method to guide the search engine. Random methods are blind, slow.
Gradient based[63].	It is local unconstrained nonlinear optimization method with successive search operation. Here the gradient or derivative function used as a search engine.	The improved version of gradient based method works fast. Mostly the Gradient based method used for local search (Newton Search Method).	It takes only local information. It required derivative function and unimodel space.
Constraint programming [64–66]	It is Continuous or discrete constraint. Mostly used for complex optimization problems with constraint.	Easily models the complex problem. The model is created by constraints of the given function.	Poor efficiency when dealing with cyclic dependencies.

Table 1 Continued

Stochastic search [22,67].	Discrete or continuous local and global search. A convex or differentiable cost function doesn't require continuous value.	Multi-Objective Model, Multimodal, Nonlinear Objective, Multi Constraints. Used for high spectrum applications.	Low circuit performance. The probabilistic nature required more iteration to cover entire design space.
Multi Objective Optimization [68–72].	Discrete or continuous Global. Optimization problem has more than one conflicting objective functions.	Multi Objective Model, Multi Constraints, Nonlinear Objectives, Trade-offs.	Performance efficiency.

Main advantages and limitation of these optimization techniques briefly resumes in Table 1.

3.2 Review of Evolutionary Computation Technique

For last few years, the efficiency of evolutionary computation (EC)[47–50] increase with respect to complex problem. It is difficult to apply classical method for solving complex problem[51] this can be overcome by EC. Evolutionary algorithm is different from other optimization techniques because it has own several characteristics. The main characteristics of evolutionary algorithm are summarized below.

- Flexible: EA applied to the problem with little knowledge or easily adapt to different types of problems. Evolutionary algorithms avoid some mathematical terms.
- Simple: With short period they allowed for model setup and problem parameter range changes.
- Robust: Evolutionary algorithm can be effective in noisy environments.
- Adaptive: They allowing dynamic changes of process parameters and also the variables are deal with self-adaptation.
- Decentralized: They are easily parallelizable[52] power distributed and highly parallel computing environments. They able to lead with solution of populations.

These are the important characteristics of Evolutionary Algorithm. Evolutionary computation is an iterative and stochastic optimization technique. These basic ideas are come from the Darwinian natural theorem of evolution. The programming technique of evolutionary algorithm are genetic algorithm[33,53–55] evolution strategies[56], evolutionary programming[54,57], and genetic programming[33]. At the same time Genetic algorithms (GA), evolution strategies (ES) and evolutionary programming (EP) are developed independently. The basic structure cycle is used by these techniques to focus on the same aims. The slight differences among these techniques are the representation of candidate selection, and the implementation of problem, recombination and mutation operators. The described techniques resumes in Table 2. At present these techniques doesn't have any big differences. Many of algorithms only differ in the constant interchange. Based on the problem specific, the researchers came to the decision that the representation of problem and type operators they achieved best solution.

Evolutionary Computation field of applications have been expanded last few years. More number of other approaches also adopts to the evolutionary computation techniques. The major advantages and disadvantages of Evolutionary Computation are given in Table 3. An evolutionary computation technique not only follows its own method and also it adapt to some other mechanism.

Table 2. Overview of Evolutionary Techniques

Evolutionary Techniques type	Main contributions	Activity Period
ES method by Ingo Rechenberg	Continuous parameter optimizations introduced in this evolution and mutation operator expand to continuous stochastic search variable. Control parameters of the search can be adjusted by self-adaptation. Don't use crossover operator.	1960s and 1970s.
Genetic Algorithms by Holland's.	Use discrete representation for encoding, traditionally 0s and 1s strings of binary. Simple optimization algorithm mostly used to for solving optimization problem and also able to evolve toward better solutions.	Became popular in 1970s.
Evolutionary Programming by Fogel.	Candidate solution represented by FSM concept, the selection strategies adapt to this environment. A population of solution evaluate with mutation and selection criteria.	End 1990s.
Genetic Programming by Koza.	Computer codes (program) are mutated by change or swap of sub trees. These sub trees represented by different kinds of problems.	1990s.

Table 3. Overview of evolutionary computation

Optimization Algorithms	Main Contributions	Drawbacks
Ant Colony Optimization (ACO)[73] introduced in 1990s.	For information holding, this optimization algorithm uses the memory structures with and deals with parallel search. It has more advantages than other methods of stochastic search algorithm.	It is hard to solve combinational and discrete optimization problems.
Particle Swarm Optimization (PSO)[74] introduced in the middle of 1990s.	PSO has small number of input parameters, so it is easy to adjust and orient for parallelization. Conceptually simple optimization so it did not require user-defined parameters. Mostly this optimization algorithm designed for local minimization and it allow the incorporation of algorithm for global optimization.	Particle Swarm is parameters dependency. The vicinity of the global optima is slow process.

Table 3 Continued

Estimation of Distribution Algorithm (EDA)[56,75].	Learning between variables is automated by incorporates methods. Probabilistic models used for discrete or continuous variables. Here crossover and mutation operations are used for probability distribution. The outperform GA application replaced by EDA.	This algorithm is not applicable or efficient to the real-time optimization, continuous optimization and multi-objective optimization.
Differential Evolution (DE)[76] introduced in the middle of 1990s.	Easy method to use with EAs. DE used to create offspring; a step size adaptation was executed automatically for the search process to converge the solution.	Require more number of variations in the algorithm. It is unclear algorithm for static conditions.

In hybrid systems some other method also involved. The simple EA does not provide the complete solution of hybrid system. On the other hand, there is more number of evolutionary algorithms available with accurate and efficient solution. This range of algorithms can be used for development of many hybrid approaches. In hybrid method the evolutionary algorithm combined with local search method, expert encoding method. The memetic algorithm follow this hybrid method[58–60]. The memetic algorithm adapt with several metaheuristics search method. This process effectively used to improve search space of global optimum problem. The search space also used for local optimum problem to find the candidate solutions[61]Evolutionary Algorithms (EAs. The other approaches with EAs optimization techniques are Dynamic Model, integer LP Method and branch and bound Model[62]. Using the specific knowledge of problem, the standard evolutionary methods and other standard techniques these hybrid approaches are extended.

EA-based techniques for problem specific hybrid systems is little complex than normal systems. The performance of hybrid systems can be improved by EA-based techniques. EAs described as very flexible tool for solving hybrid system problem. In conclusion, it is essential to hybrid EA-based techniques with other optimization techniques. This combination recognized as a contribution of great worth.

4. Conclusion

This paper reviewed the history and present state of analog design automation. Analog design automation consists of a large volume of research, so we have restricted ourselves to evolutionary algorithm applied to analog integrated circuits. The nature of the inter-disciplinary field requires close collaborations among computer science engineer and electrical scientists in developing usable methodologies for analog design automation. It is our hope that most future research in evolutionary algorithm will have the aim of topologies selection and improve the performance of analog integrated circuit design. Finally, the topologies selection and search space reduction (i.e., selecting best pareto performances) can be considered as general open problems in solving multi objective optimization problems by applying EAs.

5. References

1. Gielen GGE, Rutenbar RA. Computer-aided design of analog and mixed-signal integrated circuits. Proceedingsof the IEEE [Internet]. 2000; 88(12):1825–54.
2. Prabhakaran G, Kannan V. Design and analysis of high gain, low power and low voltage a-Si TFT based Operational Amplifier. Indian Journal of Science and Technology. 2015; 8(16):1-10.

3. Nicosia G, Rinaudo S, Sciacca E. An evolutionary algorithm-based approach to robust analog circuit design using constrained multi-objective optimization. Knowledge-Based Systems. 2008; 21(3):175–83.
4. Greenwood GW, Tyrrell AM. New Jersey: John-Wiley & Sons Inc: Introduction to Evolvable Hardware. 2007.
5. Barros M, Guilherme J, Horta N. Analog circuits optimization based on evolutionary computation techniques. Integration the VLSI Journal. 2010; 43(1):136–55.
6. Makris CA, Toumazou C. Analog IC design automation: part II - automated circuit correction by qualitative reasoning. IEEE Transactionson Comput Design Integrated Circuits Systems. 1995; 14(2):239–54.
7. Toumazou C, Makris CA. Analog IC design automation. I. Automated circuit generation: new concepts and methods. IEEE Transactionson Computer-Aided Design Integrated Circuits Systems. 1995; 14(2):218–38.
8. Degrauwe MGR, Nys O, Dijkstra E, Rijmenants J. IDAC: An Interactive Design Tool for Analog CMOS Circuits. IEEE Journal of Solid-State Circuits. 1987; 22(6):1106–16.
9. Lourenco N, Horta N. GENOM-POF: Multi-Objective Evolutionary Synthesis of Analog ICs with Corners Validation. Proceedings of the Fourteenth International Conference on Genetic Evolutionary Computation [Internet]. 2012; p. 1119–26.
10. Horta N. Analogue and mixed-signal systems topologies exploration using symbolic methods. Analog Integrated Circuits and Signal Processing. 2002; 31(2):161–76.
11. Koh HY, Sequin CH, Gray PR. OPASYN: A compiler for CMOS operational amplifiers. IEEE Transactionson Computer Design Integrated Circuits and Systems. 1990; 9(2):113–25.
12. Harvey JP, Elmasry MI, Leung B. STAIC: An Interactive Framework for Synthesizing CMOS and BiCMOS Analog Circuits. IEEE Transactionson Computer-Aided Design Integrated Circuits Systems. 1992; 11(1):1402–17.
13. Maulik PC, Carley LR, Rutenbar RA. Integer programming based topology selection of cell-level analog circuits. IEEE Transactions on Computer-Aided Design Integrated Circuits Systems. 1995; 14(4):401–12.
14. Maulik PC, Carley LR, Allstot DJ. Sizing of cell-level analog circuits using constrained optimization techniques. IEEE Journal of Solid-State Circuits. 1993; 28(3):233–41.
15. Matsukawa K, Morie T, Tokunaga Y, Sakiyama S, Mitani Y, Takayama M, et al. Design methods for pipeline & delta-sigma A-to-D converters with convex optimization. Yokohama: Proc Asia South Pacific Design Automation Conference, ASP-DAC'09. 2009; p. 690–95.
16. Hershenson MDM, Boyd SP, Lee TH. GPCAD: a tool for CMOS op-amp synthesis. 1998 IEEE/ACM International Conference on Computer Design. 1998; p. 296-303.
17. Meng KH, Pan PC, Chen HM. Integrated hierarchical synthesis of analog/RF circuits with accurate performance mapping. Santa Clara, CA: Proceedings of the 12th International Symposium on Quality Electronic Design, ISQED' 2011. 2011; p.1-8.
18. Torralba A, Chavez J, Franquelo LG. FASY: A fuzzy-logic based tool for analog synthesis. IEEE Transactionson Computer-Aided Design Integrated Circuits Systems. 1996; 15(7):705–15.
19. Gielen GGE, Walscharts HCC, Sansen WMC. Analog circuit design optimization based on symbolic simulation and simulated annealing. IEEE Journal of Solid-State Circuits. 1990; 25(3):707-13.
20. Ochotta ES, Rutenbar RA, Carley LR. Synthesis of high-performance analog circuits in ASTRX/OBLX. IEEE Transactionson Computer Design Integrated Circuits Systems. 1996; 15(3):273–94.
21. Doboli A, Dhanwada N, Nunez-Aldana A, Vemuri R. A two-layer library-based approach to synthesis of analog systems from VHDL-AMS specifications. ACM Transactions on Design Automation Electronic Systems. 2004; 9(2):238–71.
22. Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by Simulated Annealing. Science. 1983; 220(4598):671–80.
23. Nye W, Riley DC, Sangiovanni-Vincentelli A, Tits AL. DELIGHT.SPICE: an optimization-based system for the design of integrated circuits. IEEE Transactions on Computer Design Integrated Circuits Systems. 1988; 7(4):501-19.
24. Lin C-W, Sue P-D, et al. A Bias-Driven Approach for Automated Design of Operational Amplifiers. Hsinchu: 2009 International Conference on VLSI Design, Automation Test, VLSI-DAT'09. 2009; p. 118-21.
25. Medeiro F, Fernandez FV, Dominguez-Catro R, et al. A statistical optimization-based approach for automated sizing of analog cells. IEEE/ACM International Conference on Computer-Aided Design. 1994; p. 594-97.
26. Castro-Lopez R, Guerra O, Roca E, Fernandez FV. An integrated layout-synthesis approach for analog ICs. IEEE Transactions on Computer Design Integrated Circuits Systems. 2008; 27(7):1179–89.
27. Barros M, Guilherme J, Horta N. GA-SVM optimization kernel applied to analog IC design automation. Nice: Proc IEEE International Conference on Electronics Circuits, and Systems. 2006; p. 486–89.

28. Alpaydin G, Balkir S, Dundar G. An evolutionary approach to automatic synthesis of high-performance analog integrated circuits. IEEE Transactions on Evolutionary Computation. 2003; 7(3):240–52.
29. Santos-Tavares R, Paulino N, Higino J, Goes J, Oliveira JP. Optimization of multi-stage amplifiers in deep-submicron CMOS using a distributed/parallel genetic algorithm. Seattle, WA: Proceedings - IEEE International Symposium on Circuits and Systems. 2008; p. 724–27.
30. Krasnicki M, Phelps R, Rutenbar RA, Richard Carley L. MAELSTROM: efficient simulation-based synthesis for custom analogcells. New Orleans, Louisiana: Proc 1999 Des Autom Conference. 1999; p. 945–50.
31. Phelps R, Krasnicki M, Rutenbar RA, Richard Carley L, Heliums JR. Anaconda: simulation-based synthesis of analog circuits via stochastic pattern search. IEEE Transactionson Computer-Aided Design Integrated Circuits Systems. 2000; 19(6):703–17.
32. Sripramong T, Toumazou C. The invention of CMOS amplifiers using genetic programming and current-flow analysis. IEEE Transactionson Computer-Aided Design Integrated Circuits Systems. 2002; 21(11):1237–52.
33. Koza JR, Bennett FH, Andre D, Keane MA, Dunlap F. Automated synthesis of analog electrical circuits by means of genetic programming. IEEE Transactions on Evolutionary Computing. 1997; 1(2):109–28.
34. Yuan H, He J. Evolutionary design of operational amplifier using variable-length differential evolution algorithm. Taiyuan: 2010 International Conference on Computer Application Systems Modelling (ICCASM' 2010). 2010; p. 610 –14.
35. Gupta H, Ghosh B. Analog Circuits Design Using Ant Colony Optimization. International Journal of Electronics, Computer and Communications Technologies. 2012; 2(3):9–21.
36. Benhala B, Ahaitouf A, Fakhfakh M, Mechaqrane A. New Adaptation of the ACO Algorithm for the Analog Circuits Design Optimization. International Journal of Computer Sciences Issues. 2012; 9(3):360–67.
37. Manjula S, Selvathi D. Design and optimization of ultra low power low noise amplifier using particle swarm optimization. Indian Journal of Science and Technology. 2015; 8(36):1-8.
38. Kamisetty S, Garg J, Tripathi JN, Mukherjee J. Optimization of Analog RF Circuit parameters using randomness in particle swarm optimization. Mumbai: 2011 World Congress on Information Communication and Technology. 2011; p. 274–78.
39. Kumar P, Duraiswamy K. An Optimized Device Sizing of Analog Circuits using Particle Swarm Optimization. Journal of Computer Sciences. 2012; 8(6):930–35.
40. Fakhfakh M, Cooren Y, Sallem A, Loulou M, Siarry P. Analog circuit design optimization through the particle swarm optimization technique. Analog Integrated Circuits and Signal Processing. 2010; 63(1):71–82.
41. McConaghy T, Palmers P, Steyaert M, Gielen GGE. Trustworthy genetic programming-based synthesis of analog circuit topologies using hierarchical domain-specific building blocks. IEEE Transactions on Evolutionary Computation. 2011; 15(4):557–70.
42. Pradhan A, Vemuri R. Efficient synthesis of a uniformly spread layout aware pareto surface for analog circuits. New Delhi: Proc 22nd International Conferenceon VLSI Design. 2009; p. 131–36.
43. Deniz E, Dundar G. Hierarchical Performance Estimation of Analog Blocks using Pareto Fronts. Berlin: 2010 Research in Microelectronics and Electronics (PRIME). p. 1-4.
44. Gielen G, McConaghy T, Eeckelaert T. Performance space modeling for hierarchical synthesis of analog integrated circuits. Anaheim, CA: Proceedings - 42nd Design Automation Conference [Internet]. 2005; p. 881–86.
45. Michalewicz Z. Evolutionary computation techniques for nonlinear programming problems. International Transactionsin Operational Research. 1994; 1(2):223–40.
46. Dantzig G. USA: Princeton University Press: Linear Programming and Extensions. 1998.
47. Goldberg DE. USA, Addison Wesley: Genetic Algorithms in Search, Optimization, and Machine Learning. 1989. Available from: http://www.mendeley.com/research/genetic-algorithms-in-search-optimization-and-machine-learning/.
48. Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs. Berlin Heidelberg: Spinger-Verlag: Computational Statistics & Data Analysis. 1996; 24.
49. Cantu-Paz E, Goldberg DE. On the scalability of parallel genetic algorithms. Evolutionary Computation. 1999; 7(4):429–49.
50. Kicinger R, Arciszewski T, DeJong K. Evolutionary computation and structural design: A survey of the state-of-the-art. Computer Structures. 2005; 83(23-24):1943–78.
51. Zilouchian A, Jamshidi M. USA, CRC Press: Intelligent control systems using soft computing methodologies. 2001.
52. Haupt RL, Haupt SE. USA: Wiley-Blackwell: Practical Genetic Algorithms Second Edition. 2004.
53. Holland JH. USA: University of Michigan Press: Adaptation in Natural and Artificial Systems [Internet]. 1975.

54. Fogel DB. Piscataway, NJ, USA: Wiley: Evolutionary Computation: Toward a New Philosophy of Machine Intelligence. 1995.
55. Koza JR. Genetic programming as a means for programming computers by natural selection. Statisticsand Computing. 1994; 4(2):87–112.
56. Back T, Hoffmeister F, Schwefel H-P. A survey of evolution strategies. Proceedings of the Fourth International Conference on Genetic Algorithms. 1991; p. 1–8.
57. Fogel LJ. New York, NY, USA: John Wiley & Sons, Inc.: Intelligence Through Simulated Evolution: Forty Years of Evolutionary Programming. 1999.
58. Moscato P. On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. USA: Caltech Concurr Comput Program. 1989; p. 1-68.
59. Krasnogor N, Smith J. A tutorial for competent memetic algorithms: Model, taxonomy, and design issues. IEEE Transactions on Evolutionary Computation. 2005; 9(5):474–88.
60. Ong Y-S, Krasnogor N, Ishibuchi H. Special Issue on Memetic Algorithms. IEEE Transactions on System Man, Cybernetics Part B. 2007; 37(1):2–5.
61. Ong YS, Nair PB, Keane AJ, Wong KW. Surrogate-assisted evolutionary optimization frameworks for high-fidelity engineering design problems. Berlin Heidelberg, Springer-Verlag: Knowledge Incorporation in Evolutionary Computation. 2004; p. 307–31.
62. Bosman PAN, Thierens D. Exploiting Gradient Information in Continuous Iterated Density Estimation Evolutionary Algorithms. Proceedings of the BNAIC-2001 13th Belgium-Netherlands Conference on Artificial Intelligence. 2001; p. 69-76.
63. Bertsekas D. Nonlinear programming. Berkeley, CA: University of California Press: Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, J. Neyman (Ed.). 1993.
64. Deb K. An efficient constraint handling method for genetic algorithms. Computer Methods in Applied Mechanicsand Engineering. 2000; 186(2-4):311–38.
65. Coello Coello CA. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. Computer Methods Applied Mechanics and Engineering. 2002; 191(11-12):1245–87.
66. Mezura-Montes E, Velazquez-Reyes J, Coello Coello CA. Promising infeasibility and multiple offspring incorporated to differential evolution for constrained optimization. Proc 2005 Conference on Genetic Evolutionary Computation - GECCO '05. 2005; p. 225–32.
67. Zitzler E. Zurich: Swiss Fedral Institute of Technology: Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications. 1999.
68. Schaffer JD. Nashville, USA: Vanderbilt University: Some experiments in machine learning using vector evaluated genetic algorithms. 1984.
69. Fonseca CM, Fleming PJ. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. Proceedings of the International Conference on Genetic Alogrithms.1993; p. 416–23.
70. Fonseca CM, Fleming PJ. Multiobjective optimization and multiple constraint handling with evolutionary algorithms - Part I: A unified formulation. IEEE Transactions on Systems Man, Cybernetics Part A: Systems Humans. 1998; 28(1):26–37.
71. Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computations. 2002; 6(2):182–97.
72. Horn J, Nafpliotis N, Goldberg DE. A niched Pareto genetic algorithm for multiobjective optimization. Orlando, FL: Proc First IEEE Conferenceon Evolutionary Computation IEEE World Congr Comput Intelligence. 1994; 1:82–87.
73. Dorigo M, Maniezzo V, Colorni A. Ant system: Optimization by a colony of cooperating agents. IEEE Transactions on Systems Man, Cybernetics Part B Cybernetics. 1996; 26(1):29–41.
74. Kennedy J, Eberhart R. Particle swarm optimization. Perth, WA: 1995 Proceedings, IEEE Int Conference on Neural Networks. 1995; 4:1942–48.
75. Larranaga P, Lozano JA. Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation, A New Tool for Evolutionary Computation. New York: Springer-Sciences. 2002.
76. Storn R, Price K. Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization. 1997; 11(4):341–59.