SEAS: A Simulated Evolution Approach for Analog Circuit Synthesis

Zhen-Qiu Ning, Ton Mouthaan, and Hans Wallinga

MESA Research Institute
Dept. of E.E., Twente University
The Netherlands

Abstract

This paper presents a simulated evolution approach for anolog circuit synthesis based on an analogy of the natural selection process in biological environments and on the iterative improvements in solving engineering problems. A prototype framework based on this idea, called SEAS, has been implemented. measurement results show that SEAS can handle non-fixed topology with global optimum solution, is easy to adapt as technology evolves, and is easy to use by unexperienced users.

1. Introduction

Progress in some classical analog applications is limited more by the lack of suitable CAD tools than by process technology per se. Design automation ideas from digital IC design have therefore begun to migrate into analog circuit design.

There are many synthesis tools now to perform design tasks (and/or to analize circuits) [1],[2],[3],[4],[5]. They can be roughly classified into three categories: standard cell method, module generator, and knowlegde-based approach. The first two methods, however, are effective only in niche markets since the specifications of analog function blocks are more diverse and complicated than their digital counterparts. Such techniques allow the designer to make only crude tradeoffs among performance specifications, and they become obsolete rapidly in the face of technological evolution [5].

A more promising method is the knowledge based method which uses expert system technology to create a design framework for analog circuits. Several attempts have been made in the past with this method to perform behavior to structure synthesis, translating specifications into sized circuit schematics [2],[3],[4],[5]. These attempts can also be classified into two groups: multiple-fixed-architecture approaches and hierarchically-structured frameworks. In the first group, one of a number of fixed circuit schematics stored in a data base that appears to be able to satisfy a given set of specifications is selected as a candidate, then transistor sizes are determined by a kind of optimization algorithm. Design variables are only transistor geometries, the circuit schematic can not be altered during the design process. In the second group, individual subcircuits are connected to each other to build up higher level circuits. They operate in a strongly hierarchical way, breaking down even simple analog blocks such as op amps into sub-blocks like current mirrors, differential pairs, and so forth. The hierarchy makes the synthesis task more tractable, however, since the synthesis is treated as a sequence of alternating topology selection and translation steps, the easy ability to implement design tricks to push circuits close to the limits of achievable performance is lost [5].

None of the above approaches can deal with hierarchy as a critical characteristic of analog design together with global optimum solution. Analog processing shines best in high performance applications where classical digital standard cell techniques are inefficient or too costly. A design environment should allow a designer to prove that a design is optimal or at least generate sufficient design alternatives and compare them objectively. For a computer aid to be effective, it must operate within a hierarchical design environment. In addition, intelligently applied optimization frequently provides better answers with less work than closed-form or approximation theory.

In this paper, we present a simulated evolution approach based on an anology of the natural selection process in biological environments and on the iterative improvements in solving engineering problems, which can operate within a hierarchical design environment and perform the global optimization. Problem definitions and basic idea are given in Section 2. System architecture is discussed in Section 3. In Section 4, we present measurement results. Finally, conclusions and discussions are made in Section 5.

2. Simulated Evolution

2.1 Problem Representations

Synthesis of an analog circuit can be viewed as a generation of sized transistor-level schematic diagrams from performance specifications and process specifications. Assume an analog circuit, e.g. an op amp, be partitioned into several sub-blocks: output stage, current source, differential pair, current mirror, feedback block and bias current generator etc. (see Fig.1). Let m be the number of the sub-blocks, n_i, i=0,1,....,m-1, be the number of possible design styles for the ith sub-block. The status of a possible combination of the sub-blocks can be represented by a state vector *state*[N_{ij}], where N_{ij}, i=0,1,...,m-1, j $\in$ {0,1,...,n_i-1}, denotes a possible design style of the ith sub-block. It is clear that the state vector represents a definite circuit schematic. *state*[1,1,0,0,0], for instance, represents the schematic of the two stage op amp shown in Fig.3, of which the output stage is in class-AB style, the input stage current source in cascode style, etc. Let *size*[] be a sizing vector whose elements are the

IEEE 1991 CUSTOM INTEGRATED CIRCUITS CONFERENCE

independent variables which determine the dc operating point of a circuit: bias currents and bias voltages. All transistor dimensions can be calculated from this dc operation point by using a general transistor model. So the sizing vector represents a definite sized transistor-level schematic diagram. *size*$[(V_{GS}\text{-}V_T)_{1,5,6,5A}, I_{0,7,10}]$, for instance, represents the sized transistor-level schematic diagram of the op amp shown in Fig.3. The analog circuit synthesis problem is then reduced to thefollowing optimization problem:

Optimize { *state*[], *size*[] }
Such that { all specifications satisfied }. (1)

2.2 The Idea of Evolution

The problem shown in (1) is not easily solved in a closed form. Since the computation of an exact solution to this problem is normally not feasible, some heuristics should be applied to reduce the search space and generate an approximate solution.

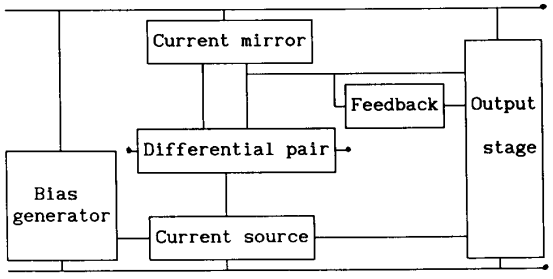

Fig.1, A partition of basic two stage op amp.

A solution for problem (1) is feasible if a given set of specifications is satisfied. We generate sufficient feasible solutions and compare them objectively. The "best" one is chosen and defined as the optimum solution. The simulated evolution approach can arrive at this purpose, which performs the task not only by means of abstract algorithmic modelling but also by including some of pragmatic knowledge about the task. The basic idea is (1) to initialize a circuit schematic (a seed); (2) to generate a sized-circuit (generation); (3) to analyze: if feasible, compare with previous solution and/or store data; (4) to score each sub-block (creature) in the current circuit; (5) to determine the survival value of each sub-block over the next circuit; (6) to replace those that failed to survive with new ones that have better (at least different) characteristics, and (7) to output a new schematic (a new seed). The iteration continues until all the specifications are satisfied and the number of feasible solutions is sufficient.

3.SEAS's System Architecture

A prototype framework based on the simulated evolution, called SEAS (Simulated Evolution for Analog Synthesis), has been implemented. SEAS's system architecture is shown in Fig.2, which is modular and extendible. The system consists of seven nodes that specialize in one or more of the major evolution aspects. They are (1) system expert, (2) initializer, (3) the simulated annealing (SA) optimization, (4) analyzer, (5) score maker, (6) schematic reconstruction, and (7) data base.

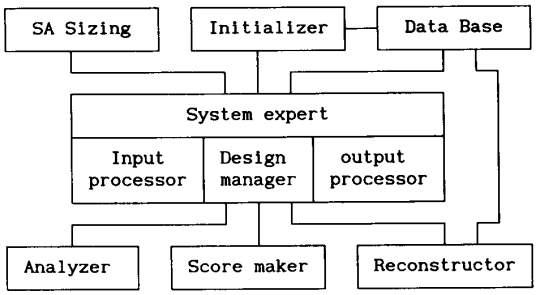

Fig.2, SEAS's system architecture.

The system expert consists of three parts: input processor, design manager and output processor. The input processor which resides on the top of the design manager makes a routine consistency check on user's input and generates deductions about the other questions it needs to ask. The user does not have to provide a complete set of specifications which the system may need, since the absent data can be reasonably deduced from the received ones by the input processor. The design manager handles the design plan, keeps track of various activities, and does the design book-keeping. The output processor reports the results of the design process to the user: sized circuit and achieved performances.

Initialization of the state vector and the sizing vector is the first step in the design plan. The initialization of the state vector consists of two selections: the selection of design style [5] and the selection of the corresponding sub-blocks. A circuit schematic is then constructed by the interconnection of the sub-blocks. It is important to correctly initialize a seed schematic fitting the given specifications since the promising one will speed up the mutation toward a good generation. In SEAS, this is implemented by a knowledge-based subroutine. Some key specifications, such as open-loop gain, power supply rejection ratio (PSRR) etc., are used to define a decision tree [3]. Searching for the seed schematic starts at the root of the decision tree, and terminates in a promising node of the tree based on the comparison with the given specifications. The node selected is then forwarded to the next step.

For a known schematic, the transistor sizing is implemented by an optimization-based algorithm. A central problem of the optimization is to determine what parameters should be used as a measure of the quality of the current design, i.e. how to determine a cost function. Since the global objective is to produce a sized schematic to satisfy all specifications, SEAS incorporates the circuit performances in the cost function: a weighted sum of design performances. The small-signal characteristics as well as the large signal characteristics are both considered. Let C denote the cost function, x_k the kth element of the sizing vector, G = $\{g_j\}$ a set of constraint functions, $P^{(s)}$ a given set of specifications, F = $\{f_i\}$ a set of objective

functions to be optimized, and w_i the weight related to the ith objective. g_j can be, for instance, slew rate, output range ect., and f_i can be chip area, power dissipation, etc. The transistor sizing problem then becomes

$$\textbf{Minimize}\quad C = \sum w_i f_i(size[]),$$

$$\textbf{Such that}\ g_j(size[]) > p_j^{(s)},\ \text{and}\ x_k^{low} < x_k < x_k^{up}. \quad (2)$$

This problem is solved in terms of simulated annealing (SA) [6], a global optimization algorithm, based on an analogy with thermodynamics, specifically with the way that metals cool and anneal.

Different from combinational problems, such as the traveling salesman problem, the optimization problem (2) involves minimization with respect to continuous parameters. The principal complication introduced in going from the discrete to the continuous application of the simulated annealing is that the choice of the random moves becomes more subtle. If moves are too small, trial configurations are too close to the starting points, thus being very inefficient in exploring the search space. On the contrary, moves which are too large will always be rejected, thus wasting computation effort. To cope with this, SEAS's SA subroutine uses a self- regulatory mechanism for the move distribution which automatically insures that an efficient choice of move size is maintained throughout the annealing [7]. This faeture keeps the CPU-time consumption acceptable even for large circuits.

In SEAS's SA subroutine, random number is generated by a pseudorandom number generator. A new cadidate point is accepted or rejected according to the Metropolie criterion [7]. The SA algorithm does not depend on any rules nor on special properties of the cost function and its derivatives. The search for an optimal solution starts only with a roughly initialized samplings. A good initial guss is not compulsory.

The analyzer calculates the performance parameters of the current circuit based on the resulting sizing vector, checks if the resulting solution is feasible, and stores data if so. Depending on the feasible conditions and the number of feasible solutions the users define, the evolution process is repeated by restructuring solutions. Finally a number of feasible solutions is obtained and compared objectively. All performances have to be included and evaluated in order to allow a meaningful comparison.

The score maker scores each sub-block (creature) in the current circuit (generation) based on an analogy of the natural selection in the biological environments. If a resulting parameter does not satisfy its specification, positive mutation values will be assigned to the sub-blocks related to this parameter. On the contrary, if satisfied, negative mutation values will be given. The mutation values characterize the mutation necessities of the sub-blocks with respect to the corresponding parameter.

For the current circuit, let s_i denote the score of the ith sub-block, α_{ij} the positive mutation value of the ith sub-block related to the jth performance parameter, β_{ik} the negative mutation value of the ith sub-block related to the kth performance parameter, and w the weights. SEAS scores the ith sub-block as follows

$$s_i = -\sum_j w_{ij}\,\alpha_{ij} - \sum_k w_{ik}\,\beta_{ik} - w_{ia}\,\frac{\text{actual-number-of-transistors}}{\text{lower-bound}}. \quad (3)$$

The first term directs the evolution toward a feasible solution, the second term deals with the balance of conflicting requirements, and the third term guides the process toward a simple design style. If the current ciruit is feasible, the first term will become zero, and the second term will dominate the score which shows a more sensitive part to be ripped-up and reconstructed (sign is changed from negative to positive for this case).

The mutation values are determined by sensitivity analysis and/or design knowledge. As an example for the sensitivity analysis, consider a basic two stage op amp. The mutation value of the input stage current mirror related to the low frequence gain A_0 can be expressed as the sensitivity of A_0 with respect to the output resistance of the current mirror. As an example for the knowledge-based approach, consider the mutation value of the output stage related to the power dissipation. Assume two possible design styles for the output stage are available: simple style and class-AB style. For a basic two stage op amp, it is known, the standby current of output stage can often be reduced by using a class-AB stage [8]. If the output stage of the current op amp is in the simple style, it seems reasonable that the simple style is ripped-up and replaced by the class-AB one; but if already in the class-AB style, it seems not reasonable to mutate it back to the simple one. SEAS simply defines a mutation value, say 1, to denote "reasonable to mutate", and a mutation value, say 0, to denote "not reasonable to mutate".

With the score of each sub-block in hand, SEAS is ready to decide which sub-block is failed to survive. It is a greedy heuristics that the sub- block with the lowest score is always ripped-up and replaced. It may converge to a local optimum. To cope with this weakness, SEAS reorders the scores by multiplying them with a set of random numbers generated in the range [0, 1] by a pseudorandom number generator, and then chooses the "worst" one, i.e. the one with the lowest new score (survival value).

Selection of alternatives is based on the knowledge-based approach. If the input stage current source is in simple style and failed to survive, for instance, one solution is to replace it with a cascode one. This is due to the fact that the common-mode rejection ratio (CMRR) of a two stage op amp is directly proportional to the output resistance of the input stage current source. If there is no knowledge available, SEAS just moves randomly from the current style to a new one that has different characteristics. It should be noted in this step that there is another way to improve the circuit performance. If the weights used in the cost function are modified, it is also possible to favor the more important performance in a comprimise.

Once the current state vector is updated, the subsequent cost function C, constraint functions G, and dimensions of the sizing vector of the new circuit will be carried out

with the new alternative in place of the original one. This is implemented by an adaptive reconstruction technique, in which the functions corresponding to the replaced sub-block of the circuit are substituted by the functions of the replacing sub-block. To explain the concept of function substitution, consider the basic two stage op amp shown in Fig.3. Let g_{mi}, g_{ml} denote the transconductances of transistors Q_1 and Q_3, respectively. Let r_{01}, r_{05} be the output resistances of the differential pair and the current source, respectively. The CMRR of the circuit can be approximated as CMRR = $2g_{mi}g_{ml}r_{01}r_{05}$. If the current source is now modified from simple style to cascode one and the simple square-law MOSFET model is used, for instance, r_{05} will be replaced by $2/(\lambda^2 I_0(V_{GS}-V_T)_{5A})$ instead of by λI_0, i.e. the function corresponding to the simple style is substituted by the function of the cascode one. The resulting CMRR is then used for the next analysis.

4. Results

SEAS is written in C and is running under a UNIX system on an Apollo SR10.2. SEAS currently designs only CMOS op amps from performance specifications and process specifications.

Table 1, Design example: a two stage op amp

Parameters	Specif. values	Achieved values (SPICE simul.)
Low freq. gain (dB)	≥ 75.0	91
Unity gain freq. (MHz)	≥ 2.0	2.5
Slew rate (V/μs)	≥ 4.0	5.7
Common mode rejection ratio (dB)	≥ 90.0	104
Phase margin (deg.)	> 60	> 90
Power dissip. (mW)	≤ 2.0	1.9
Output range (V)	+ 4.4 - 4.4	+ 4.4 - 4.6

An example for basic two stage op amp is given in Table 1 and Fig.3. Table 1 shows the specifications and the achieved performance parameters which are measured by detailed circuit simulation. Fig.3 gives the resulting schematic diagram. The achieved performances are satisfied with the given specifications quite well.

5. Conclusions and Discussions

SEAS treats the synthesis task as an optimization problem of both circuit topology and its component values based on the simulated evolution. It can handle non-fixed topology with (global) optimum solution, can favor the more important performance goals in a compromise, is easy to adapt as technology evolves and is easy to use by unexperienced users.

Analytic circuit models, in which a set of the design

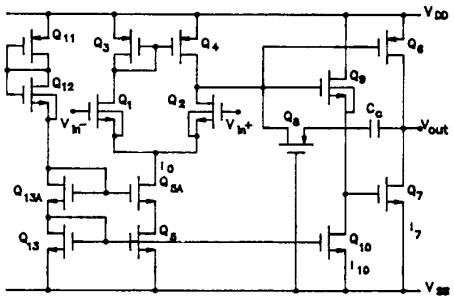

Fig.3, A two stage op amp with class-AB output stage and input stage cascode current source.

equations enable us to evaluate the performance of a circuit for different sets of specifications without actually going into the circuit details, can be developed not only for op amps but also for other type of analog functional blocks such as comparators, A/D convertors, etc., so SEAS can be extended to cover these analog circuit functions as well.

Analog circuit may interface to digital in VLSI design, some of the objectives will be discontinuous. This will result in a fatal problem for most design frameworks with optimization algorithms. But for SEAS, this problem can be handled, since the simulated annealing works well in discrete domain.

SEAS currently selects sub-blocks stored in the data base and builds up the design equations correspondingly. If a symbolic analysis tool is available to generate analytical design equations for new schematics, SEAS will become more general and attractive.

References:

[1] T. Pletersek, J. Trontelj, and L. Trontelj, "Analog LSI design with CMOS standard cells," Proc. IEEE CICC, pp. 479-483, 1985.

[2] M.G.R. DeGrauwe et al., "IDAC: An interactive design tool for analog CMOS circuits," IEEE J. Solid-State Circuits, vol. SC-22, Dec. 1987.

[3] H.Y. Koh, C.H. Sequin, and P.R. Gray, "OPASYN: A Compiler for CMOS Operation Amplifiers," IEEE Trans. CAD, vol.9, Feb. 1990.

[4] F.M. El- Turky and E.E. Perry, "BLADES: An Artificial Intelligence Approach to Analog Circuit Design," IEEE Trans. CAD, vol.8, June 1989.

[5] R. Harjani, R.A. Rutenbar, and L.R. Carley, "OASYS: A Framework for Analog Circuit Synthesis," IEEE Trans. CAD, vol.8, Dec. 1989.

[6] S. Kirkpatrick, C.D. Gelatt Jr., and M.P. Vecchi, "Optimization by simulated annealing," Science, vol. 220, no. 4598, pp. 671-680, May, 1983.

[7] D. Vanderbilt and S.G. Louie, "A Monte Carlo Simulated Annealing Approach to Optimization over Continuous Variables," J. of Computational Physics 56, 259-271, 1984.

[8] R. Gregorian and G.C. Temes, "Analog MOS Integrated Circuits for Signal Processing," JOHN WILEY & SONS, 1986.