An Analog Performance Estimator for Improving the Effectiveness of CMOS Analog Systems Circuit Synthesis

Adrian Nunez-Aldana and Ranga Vemuri
Electrical and Computer Engineering and Computer Science, University of Cincinnati
Cincinnati, OH 45221-0030, USA
{anunez,ranga}@ececs.uc.edu

Abstract

Critical to the automation of analog circuit systems is the estimation process of performance parameters which are used to guide the topology selection and circuit sizing processes. This paper presents a methodology to improve the effectiveness of the CMOS analog system circuit synthesis search process by developing an Analog Performance Estimator (APE) tool. APE is capable of accepting the design parameters of an analog circuit and determine its performance parameters along with anticipated sizes of all the circuit elements. The APE is structured as a hierarchical estimation engine containing performance models of analog circuits at various levels of abstraction.

1. Introduction

The analog CMOS integrated circuit synthesis process consists of three steps: topology selection, circuit sizing and design verification [3, 2]. Topology selection is the process of selecting analog circuit elements arranged in predetermined topologies to realize the desired function. Circuit sizing phase determines the physical dimensions, bias points, and element values to meet user constraints. The design verification phase is typically performed by a circuit simulator such as SPICE.

Due to the vast design-space available, analog circuit sizing is generally considered to be a time-consuming process. Typically, the circuit sizing process starts with an initial design point, in the form of initial sizes for various circuit elements, and employs an algorithm to search through the design-space until converges at a design that is likely to meet the user constraints. The speed of convergence and the quality of the design produced depend on the initial design point as well as the design-space exploration algorithm used.

This paper presents a methodology for improving the effectiveness of CMOS analog systems circuit synthesis by estimating the performance and generating a sized circuit which can be used as the initial design point in a more rigorous analog circuit sizing tool. The methodology is embedded in a tool called the Analog Performance Estimator (APE). APE permits a circuit designer or a circuit synthesis tool to estimate several characteristics of analog circuits, including DC gain, Unity Gain Frequency (UGF), transistor area, static power dissipation, operating points, etc., at an early stage of the design process.

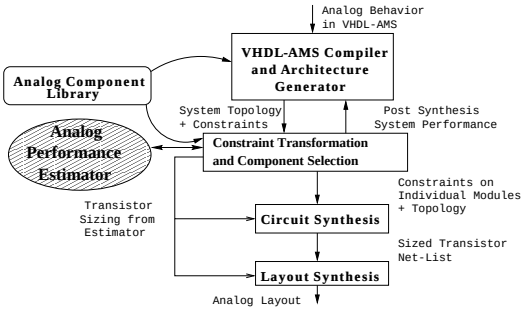

Figure 1. Analog Synthesis in VASE

The APE is structured as a hierarchical estimation engine and contains performance models of analog circuits at various levels of abstraction. The levels include the basic circuit elements (MOS transistors, resistors, capacitors etc.), simple analog circuits (current mirrors, V-I converters etc.), operational amplifiers (in various configurations), and analog library cells (integrators, filters, amplifiers etc.). The APE uses technology process parameters and SPICE models of analog circuit elements at the lowest level. It contains performance composition equations for determining circuit performance at the other levels of abstraction.

APE is part of VASE, a mixed-signal synthesis system being developed at the University of Cincinnati [20]. Figure 1 shows the synthesis flow of VASE. A mixed-signal system is described at behavioral level using a subset of VHDL-AMS. A VHDL-AMS compiler and architecture generator [6] transforms the specification into net-lists of components. A

constraint transformation process [5] allocates the system constraints onto analog modules. The architecture generator and the constraint transformation process are guided by the estimates produced by APE.

2 Motivation

In order to underline the importance of the initial design point during the search process, we present some experiments conducted using the ASTRX/OBLX analog circuit synthesis system developed at the Carnegie-Mellon University [14]. This system sizes a given analog circuit topology to meet specified user constraints. The sizing engine is based on a simulated annealing search algorithm. Table 1 (left side) shows the specifications of ten operational amplifiers we attempted to synthesize using ASTRX/OBLX. Each specification was submitted to ASTRX/OBLX without initial design points. Table 1 (right side) shows the results after simulating the sized circuits produced by ASTRX/OBLX upon convergence. Only in one case (OpAmp9) the design worked correctly and met the constraints within reasonable accuracy. In one case (OpAmp1) the circuit didn't simulate correctly. In all other cases, one of the specifications was unacceptably violated by the sized design produced by ASTRX/OBLX.

The APE can be used as a front-end to a circuit synthesis tool such as ASTRX/OBLX to generate an initial design-point from which to conduct the search. Experimental results presented later in this paper show that this makes the search quite effective.

3 Related Work in Analog Synthesis

Approaches to synthesize analog CMOS circuits can be classified as knowledge-based or optimization-based [3]. The knowledge-based approaches encode the circuit behavior in memory, while the optimization-based systems obtain the behavior via simulation. Examples of knowledge-based analog synthesis tools are OASYS [9] and IDAC [4] and examples of optimization-based synthesis tools are OPASYN [11], STAIC [10] and DELIGHT.SPICE [13].

ARIADNE [17] uses a mathematical approach based on symbolic simulation (ISAAC [7]) and simulated annealing, to synthesize analog systems. OASYS [9] is a hierarchically structured synthesis tool making use of analog circuit design knowledge. Using a hierarchical representation, OASYS reduces the complexity of the synthesis process; however, it loses the ability to explore different configurations in a flat design. OASYS relies on generate-and-simulate approach and allows back-tracking across the levels in the hierarchy. STAIC is based on a successive solution refinement methodology. OPASYN assumes a synthesis by analysis approach based on analytical models. The synthesis problem is formulated as nonlinear constrained optimization based on analytical circuit models. DARWIN [12] is a tool that is able to synthesize CMOS opamps using a genetic algorithm. In DARWIN, opamp topologies are built from basic blocks. DONALD [16] uses constraint programming to relate the design equations to a user system application. FASY [19] is a fuzzy-logic based synthesis tool which selects a topology from a pre-defined library. In FASY, the components are pre-sized to satisfy the performance. Performance modeling in FASY is also based on fuzzy-logic [18].

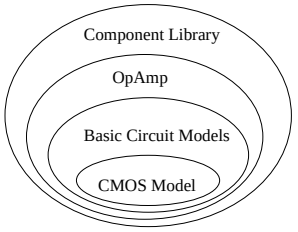

Figure 2. Hierarchical Structure of the APE

ASTRX/OBLX is an automated analog circuit synthesis tool developed at the Carnegie-Mellon University. In ASTRX/OBLX, the circuit topology is already selected. The transistor sizes and bias points are set as unknowns [1]. The user provides intervals to establish ranges of allowable values for the unknowns. If the intervals are smaller, the search will converge faster. The specifications, objectives and constraints embed the behavior of the circuit to be sized. ASTRX/OBLX generates a cost function from the objectives, specifications, constraints and Kirchoff Laws. The AWE (Asymptotic Waveform Evaluation) technique [15] is used to simulate the circuit. The optimization engine is based on a simulated annealing algorithm. The time of convergence and the accuracy of results are improved when the unknown intervals are smaller and the number of unknowns, specifications, objectives and constraints are as minimum as possible. Therefore, an initial knowledge of the transistor sizes, operating points, and small signal characteristics can speed up the process of convergence successfully.

4 APE: Analog Performance Estimation Methodology

APE has a layered hierarchical structure shown in Figure 2. Estimation and circuit sizing are carried out using a bottom-up approach, from the transistor level specification through the system level application. The user application is specified as a net-list of elements from the component library and a set of system requirements. Each library component is built using operational amplifiers, basic analog circuits (DC-bias voltage, current source, gain amplifiers,

Specifications/Topologies										ASTRX/OBLX Stand Alone Results					
ckt	Gain	UGF	Area	Ibias	CurrSrc	Diffgain	Buff	Z	CL	Gain	UGF	Gate Area	power	CPU	Comments
	abs	Mhz	μ^2	μA				$K\Omega$	pF	abs	Mhz	μ^2	mW	Sec.	
oa0	200	1.3	5000	1.0	Wilson	CMOS	Yes	1	10	236.7	0.776	1684.9	4.65	598.85	UGF < spec
oa1	70	3.0	3000	2.0	Wilson	CMOS	Yes	1	10	-	-	532788.9	3.77	738.25	doesn't work.
oa2	100	2.5	2000	1.5	Wilson	CMOS	Yes	2	10	0.253	-	50882.6	12.2	1557.51	Gain << Spec
oa3	250	8.0	1000	1.0	Mirror	CMOS	No		10	2099.7	12.47	23745.8	3.74	355.09	Area >> Spec
oa4	150	3.0	1000	100.0	Mirror	CMOS	No		10	1.94	12.47	932.8	26.4	367.71	Gain << Spec
oa5	200	8.0	5000	10.0	Mirror	CMOS	No		10	0.45	-	93602.6	39.2	266.55	Gain << Spec
oa6	50	10.0	200	10.0	Mirror	CMOS	No		10	0.098	-	2786.8	41.8	245.64	Gain << Spec
oa7	200	3.0	6000	1.0	Mirror	CMOS	Yes	1	10	226.78	59.2	234397.4	10.8	764.36	Area >> Spec
oa8	100	2.0	1000	1.0	Mirror	CMOS	Yes	10	10	0.0053	-	488.5	22.8	568.80	Gain << Spec
oa9	200	5.0	5000	10.0	Mirror	CMOS	Yes	10	10	315.72	10.59	4635	3.71	479.93	Meets spec

Table 1. Operational Amplifiers: Design Specifications and Synthesis Results

etc.), transistors, resistors and capacitors. Several topologies of operational amplifiers can be generated by using different arrangements of basic analog circuits. The basic circuits contain CMOS transistors. Each level in the hierarchy gets the requirements from its parent and decomposes them into its subcomponents. Eventually, the transistors are sized using the requirements and their performance estimates are generated. The estimates are propagated through all the levels using symbolic equations that relate them. At the end, each performance parameter has been estimated and each circuit element is sized. The following paragraphs describe the models used in the APE.

1. CMOS Transistor Models: At lowest level of the hierarchy are the CMOS transistor models. A transistor is sized according to its DC operating point and the fabrication process parameters. Then the small signal characteristics are evaluated. The sized transistor is saved as an object which contains the size and performance parameters. Several objects can be generated with different operating points as they are needed to construct the other levels in the circuit hierarchy.

For example a CMOS transistor in the saturation region can be modeled by the following symbolic equations:

$$I_{ds} = (\mu\epsilon/t_{ox})(W/L)(V_{gs} - V_{th})^2/2 \quad (1)$$

$$gm = \sqrt{4(KP)(W/L)|I_{ds}|} \quad (2)$$

$$gmb = \frac{\gamma/2}{\sqrt{2\phi_f + |Vsb|}} gm \quad (3)$$

$$gd = \frac{\lambda I_{ds}}{1 + \lambda|V_{ds}|} \quad (4)$$

The transistor sizing process consists in solving these symbolic equations such that the constraints are met. For example, if a transistor is specified by a given transconductance g_m (Gain) and a drain current, APE estimates the transistor size, the output drain conductance and the parasite capacitances. It should be noted that, the sizing process is tied to the fabrication process parameters and the sizing accuracy is directly dependent on the transistor model used. The current version of APE can use Level 1, 2, 3 or BSIM SPICE device models.

2. Basic Analog Components: A library of basic components is the next level in the APE. Some of these components are DC-bias voltages, current sources, gain amplifiers, output buffers, differential amplifiers and differential-to-single-ended converters. This library contains several topologies for each component, e.g. a current source can be implemented as a Cascode or a Wilson topology. APE contains a set of symbolic equations which relate the performance of the components to the circuit topology.

To estimate the performance of a basic analog component, APE evaluates the performance of each transistor according to the topology and specifications. With this information, the performance parameters of the component are calculated. For example, a differential CMOS amplifier is modeled by the following equations:

$$A_{dm} \approx \frac{g_{mi}}{g_{dl} + g_{di}} \quad (5)$$

$$A_{cm} \approx \frac{-g_0 g_{di}}{2g_{ml}(g_{dl} + g_{di})} \quad (6)$$

$$CMRR \approx 2\frac{g_{mi} g_{ml}}{g_0 g_{di}} \quad (7)$$

If a differential CMOS amplifier is specified by its differential gain mode A_{dm}, APE solves the equations for the transistor transconductances. Then, with the transistor transconductances, all parameters of each transistor are estimated using the CMOS transistor Models. Finally, the performance parameters (A_{cm}, $CMRR$, $GateArea$, UGF, Z_{out}, $DCPower$ and $SlewRate$) are evaluated. A new object is created with the estimates and sizes attached as attributes.

Table 2 compares the APE estimations and SPICE simulation results of some basic components after being sized according to user-specified requirements. This table shows that the models used in the APE are reasonably accurate.

3. Operational Amplifiers: The third level in the APE hierarchy consists of a set of operational amplifier topologies. A general structure of an opamp can be represented by three stages [8]: (1) Differential input amplifier; (2) Level shift, differential to single-ended converter, and gain stage;

Circuit	DC Power mW		Adm		UGF MHz		Ibias μA		Zout $Kohm$		Gate Area μ^2		CMRR dB		Slew Rate $V/\mu S$	
	est	sim	est	sim	est	sim	est	sim	est	sim	est	sim	est	sim	est	sim
OpAmp1	.29	.28	206	223	1.3	2.1	1	.9	1	.9	4885.7	4884.4	129.7		0.1	0.106
OpAmp2	.17	.19	374	380	8.0	13.7	2	1.9	1	.9	2379.6	2376.2	141.8		0.2	0.22
OpAmp3	.15	.16	167	170	12.4	9.8	1.5	1.4	2	1.8	1010.8	1010.8	133.8		0.15	0.164
OpAmp4	.24	.29	514	489	2.6	4.0	1	1.1			696.9	696.9	98.6		4.53	4.32

Note 1: OpAmp1, OpAmp2, OpAmp3 topology: Wilson, DiffCMOS, Output Buffer. OpAmp4 topology: Mirror, DiffCMOS

Table 3. Estimation vs SPICE Simulation of OpAmp's

Topology	Gate Area μ^2		UGF MHz		DC Power mW		Gain		Current μA	
	est	sim	est	sim	est	sim	est	sim	est	sim
DCVolt	3.1	3.0			.5	.53	2.5	2.49	100	105
CurrMirr	165.7	165.6			.5	.56			100	97.9
Wilson	383.3	383.1			.5	.58			100	106
GainNMOS	101	101	15.7	15.2	.6	.82	-8.5	-8.0		
GainCMOS	101	101	26.7	30.3	.62	.8	-19.0	-16.9		
GainCMOSH	101	101	8.3	8.7	.23	.22	-5.1	-5.3		
Follower	123.4	123.4			.5	.64	.8	.81	100	128
DiffNMOS	47.2	45.8	6.0	6.5	.5	.49	-10	-10.2	1	1.02
DiffCMOS	15.8	15.8	4.4	4.6	.5	.49	1000	1055	1	1.02

Table 2. Estimation vs SPICE Simulation for Basic Analog Circuits

(3) Output buffer. The output buffer is only needed if a the amplifier is heavily loaded. Each of these stages can be implemented with elements from the library of basic components. The operational amplifier structure contains the supported topologies and the equations which relate the performance parameters of the opamp and the parameters of the basic component library. When an opamp is estimated, a topology is specified (bias current, type of current source, type of differential amplifier, gain stage, output buffer, output impedance, internal compensated, and load capacitance). The performance parameters of the opamp are estimated using the attributes of the basic components after being sized and the opamp performance equations.

Once the amplifier is sized and estimated, a new opamp object is created with its performance attributes and topology. Table 3 compares the performance estimation and SPICE simulation results of several operational amplifiers after being sized. Again these results show that the opamp models used in APE are reasonably accurate.

4. Analog Modules: A library of analog modules forms the fourth level of the APE methodology structure. Each component in the library is constructed using opamp's, elements from the basic component library, transistors, resistors and capacitors. The library consists of circuits such as inverting amplifiers, integrators, comparators, analog-to-digital converters, digital-to-analog converters, filters, sample-and-hold circuits, adders, etc. The performance parameters of these components are estimated using the operational amplifier estimation attributes and the equations in the component library which relate the ideal behavior of the component with the non-ideal characteristics of the opamp. Because of the hierarchical structure, the characterization of this library components carriers the topology information and the fabrication process parameters.

Table 5 shows experimental results concerning the performance of the APE at this level. Columns 1-3 show the name of the library module and the user-specifications for various design parameters. Columns 5-6 show the APE estimates and simulation results. These results demonstrate the accuracy of the APE estimates.

5 Experimental Results for Circuit Synthesis using the APE

In this section, we present experimental results involving circuit synthesis using APE to generate an initial circuit sizing, followed by the ASTRX/OBLX synthesis tool to search the design space in the vicinity of the initial sizes to more closely meet the constraints.

1. Synthesis of Operational Amplifiers: We have synthesized the ten operational amplifiers specified in Table 1 using APE followed by ASTRX/OBLX. The results are shown in Table 4. For each unknown parameter specified to the ASTRX/OBLX system, we have specified its search region to be centered around the corresponding value estimated by the APE with the lower/upper bounds specified as being +/- 20% away from this value. Note that in contrast to Table 1, ASTRX/OBLX system is now able to find correctly working designs while meeting the performance constraints in each case. Also note the overall synthesis time improvement in all cases but one. (The CPU time required to execute the APE for all the ten opamps combined was 0.12 seconds on a Ultra Sparc 30 workstation and is essentially negligible compared to the ASTRX/OBLX synthesis times. CPU times shown in the table include the total time for APE+ASTRX/OBLX execution).

2. Synthesis of Analog Library Modules: To further illustrate the effectiveness of the methodology, we synthesized 5 design examples (audio amplifier, sample and hold, 4-bit flash ADC, low-pass filter and band-pass filter) at the analog library module level of the APE hierarchy. The

ckt	Gain	UGF	Area	power	CPU	speed-up	Comments
	abs	Mhz	μ^2	mW	Sec.		
oa0	206.20	2.64	4666.9	0.64	516.41	13.8%	Meets spec
oa1	73.04	3.63	2507.3	0.62	457.63	38.0%	Meets spec
oa2	108.57	2.97	1120.6	0.56	440.17	71.7%	Meets spec
oa3	287.20	8.07	765.4	3.01	475.57	-33.9%	Meets spec
oa4	152.28	3.59	322.4	6.64	235.99	35.8%	Meets spec
oa5	227.57	9.88	4976.3	2.75	157.19	41.0%	Meets spec
oa6	54.01	10.51	155.2	0.72	190.36	22.5%	Meets spec
oa7	212.99	3.22	5068.6	2.41	384.27	49.7%	Meets spec
oa8	112.53	2.34	783.8	1.20	412.80	27.4%	Meets spec
oa9	227.25	5.30	2931.4	6.27	363.15	24.3%	Meets spec

Table 4. OpAmp Results: ASTRX/OBLX with APE init.

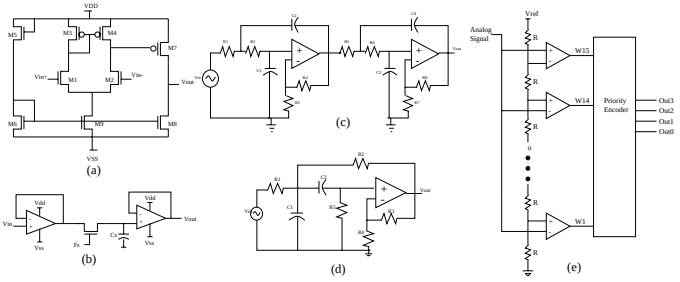

Figure 3. (a) amp, (b) S&H, (c) LPF, (d) BPF, (e) ADC

topology of the audio amplifier is a 2-stage operational amplifier in open-loop configuration with a gain of 100 and 20Khz bandwidth. The sample and hold was designed to have response time of $500\mu S$. The low-pass filter is a 4-order Sallen-Key, Butterworth, and 1Khz bandwidth. The band pass filter is a 2-order Sallen-Key, Butterworth, and 1Khz central frequency. Finally, The analog to digital converter is a 4-bit Flash ADC. These components are shown in Figure 3. Columns 1-3 in Table 5 show the specifications for these circuits.

We used ASTRX/OBLX tool to perform the circuit synthesis. First, we ran it without any pre-sizing information. The results are shown in Column 4, Table 5. ASTRX/OBLX was not able to give a functional design for the low-pass filter and band-pass filter circuits. The sample and hold and the audio amplifier circuits didn't meet the bandwidth constraint. The flash ADC produced has a very large circuit area.

Then we applied our methodology of approximate pre-sizing using the APE to each design. We ran APE to reduce the search space. The CPU time to run APE was within 0.14 seconds for all the examples. The ASTRX/OBLX input file was modified such that the the starting points were set as determined by the APE and the intervals were narrowed to be within +/- 20% of these points. Then, correctly functioning, constraint-satisfying circuits were produced by ASTRX/OBLX in each case. Table 5, Column 7 shows the final simulation results after the circuits were synthesized.

1	2	3	4	5	6	7
ckt	param	spec	ASTRIX sim.	APE est.	APE sim.	APE+A/O sim.
s&h	gain	2.0	1.707	2.1	2.07	2.03
	BW	20khz	0.598khz	40.khz	49.2khz	51.172khz
	SR	.01	.002	.05	.041	0.081
	area	$500\mu^2$	$3668.3\mu^2$	$346\mu^2$	$346\mu^2$	$323.8\mu^2$
	CPU		**464.98sec.**	**0.14sec.**		**304.48sec.**
amp	gain	100	955	103.1	104.4	130.66
	BW	20khz	4.82khz	23.7khz	20.5khz	25.4khz
	area	$1000\mu^2$	$156537.6\mu^2$	$904\mu^2$	$902\mu^2$	$995.9\mu^2$
	CPU		**347.13sec.**	**0.14sec.**		**154.94sec.**
adc	bits	4	4	4	4	4
	delay	$5\mu S$	$1.37\mu S$	$4.53\mu S$	$5.2\mu S$	$2.53\mu S$
	area	$5000\mu^2$	$87441.3\mu^2$	$3998\mu^2$	$4005\mu^2$	$3604.5\mu^2$
	CPU		**812.13sec.**	**0.14sec.**		**608.96sec.**
lpf	style	flat	flat	flat	flat	flat
	type	SK	SK	SK	SK	SK
	order	4	4	4	4	4
	f_{-3dB}	1khz	Doesn't Work	1khz	914hz	940hz
	f_{-20dB}	1.7khz	Doesn't Work	1.75khz	1.82khz	1.742khz
	gain	4.469	Doesn't Work	4.5	4.45	4.2287
	area	$10000\mu^2$	$7737.5\mu^2$	$8219.6\mu^2$	$8212\mu^2$	$2287.2\mu^2$
	CPU		**3130.39sec.**	**0.14sec.**		**2770.28sec.**
bpf	style	flat	flat	flat	flat	flat
	type	SK	SK	SK	SK	SK
	order	2	2	2	2	2
	f_0	1khz	Doesn't Work	1.1khz	1054hz	993.1hz
	gain	2.728	Doesn't Work	3.0	3.29	2.874
	BW	1khz	Doesn't Work	950hz	823.5hz	911.8hz
	area	$5000\mu^2$	$9889.4\mu^2$	$4109.8\mu^2$	$4106\mu^2$	$1079.6\mu^2$
	CPU		**491.37sec.**	**0.14sec.**		**440.58sec.**

Table 5. Design examples

These results show that using the ASTRX/OBLX alone for analog circuit synthesis might be impractical. Typically, one may have to execute ASTRX/OBLX many times before a constraint-satisfying and functionally correct design can be generated. This is an extremely time consuming process, given the fact that even one run through the ASTRX/OBLX tool takes a long time, as shown by the results in this paper. The methodology presented in this paper helps alleviate the following mutually-related difficulties in using ASTRX/OBLX for circuit synthesis: (1) non-convergence of ASTRX/OBLX in a reasonable time, (2) the need to repeat ASTRX/OBLX synthesis runs many times before producing the required design, and, (3) the need to "intelligently guess" the ranges of allowable values for for the variables along with initial search points.

The CPU time required to run ASTRX/OBLX is directly dependent on the simulating annealing based optimization

engine. ASTRX/OBLX allows the user to override any of the parameters used during the annealing process. Tuning these parameters can help find a working design. However, it may increase the execution time. And, the parameter settings fine-tuned one design may not work for another design. In our experiments, we have used the same default simulated annealing parameter settings for all experiments. In our experiments, narrowing the search intervals by employing the APE didn't show as much impact as one might expect on the ASTRX/OBLX execution time. The impact was primarily on its ability to discover a working, constraint-satisfying design. We are currently exploring ways to utilize the APE-generated information during the annealing process such that a significant improvement on the execution time can be obtained.

6 Conclusion

The methodology for improving the effectiveness of CMOS analog systems circuit synthesis presented in this paper embodies a hierarchical approach to estimate the circuit sizing and performance parameters. We have demonstrated that the size estimates can be effectively used as the starting point during the process of analog circuit synthesis using an optimization-based circuit synthesis tool. The hierarchical structure allows to easily add new components to APE, making use of lower levels in the structure.

Critical to the success of analog synthesis is the accuracy of analog performance estimation at various levels of abstraction. We are currently incorporating into the APE performance estimation procedures for user-level analog netlists. We are also currently exploring the applications of the APE for analog performance estimation at various stages in a top-down analog synthesis process, from behavior specifications to analog layouts.

7 Acknowledgement

This work was sponsored by the USAF, Wright Laboratories, Wright Patterson Air Force Base under contract number F33615-96-C-1911. The authors thank Nazanin Mansouri for her help with the manuscript. A. Nunez-Aldana would extend his gratitude to CONACYT-Fulbright for their support.

References

[1] ACACIA. Release 2.0, user guides. *ACACIA Group*, 1994.

[2] B. Antao and A. Brodersen. Techniques for synthesis of analog integrated circuits. *IEEE Design & Test of Computers*, pages 8–18, March 1992.

[3] L. R. Carley, G. Gielen, R. A. Rutenbar, and W. Sansen. Synthesis tools for mixed-signal ics: Progress on frontend and backend strategies. *DAC Proceedings*, pages 298–303, June 1996.

[4] M. Degrauwe, O. Nys, and E. Dijkstra. Idac: an interactive design tool for analog cmos circuits. *IEEE Sol. st. Circ.*, 22:1106–16, December 1987.

[5] N. Dhanwada, A. Nunez-Aldana, and R. Vemuri. Hierarchical constraint transformation using directed interval search for analog system synthesis. *DATE Proceedings*, March 1999.

[6] A. Doboli and R. Vemuri. A vhdl-ams compiler and architecture generator for behavioral synthesis of analog systems. *DATE Proceedings*, March 1999.

[7] G. E. Gielen, H. Walscharts, and W. Sansen. Isaac: A symbolic simulator for analog integrated circuits. *IEEE J. Solid-State Circuits*, 24(6):1587–1597, December 1989.

[8] R. Gregorian and G. C. Temes. *Analog MOS Integrated Circuits For Signal processing*. Wiley series on Filters, 1986.

[9] R. Harjani, R. Rutenbar, and L. Carley. Oasys: a framework for analog circuit synthesis. *IEEE Trans CAD*, 8(12):1247–1266, December 1989.

[10] J. Harvey, M. Elmasry, and B. Leung. Staic: An interactive framework for synthesizing cmos and bicmos analog circuits. *IEEE Trans. CAD*, 11(11):1402–1415, Nov. 1992.

[11] H. Koh, C. Sequin, and P. Gray. Opasyn: a compiler for cmos operational amplifiers. *IEEE Trans. CAD*, 9(2):113–125, Feb. 1990.

[12] W. Kruiskamp and D. Leenaerts. Darwin: Cmos opamp synthesis by means of a genetic algorithm. *32nd Design Automation Conference*, pages 433–438, Jun 1995.

[13] W. Nye, D. Riley, A. Sangiovanni-Vicentelli, and A. Tits. Delight.spice: An optimization-based system for the design of integrated circuits. *IEEE trans. CAD*, 7(4):501–519, Apr. 1988.

[14] E. S. Ochotta, R. A. Rutenbar, and L. R. Carley. Synthesis of high-performance analog circuits in astrx/oblx. *IEEE Trans. CAD*, 15:273–294, Mar. 1996.

[15] L. Pillage and R. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Trans. CAD*, 9(4):352–366, Apr. 1990.

[16] K. Swings and W. Sansen. Donald: a workbench for interactive design space exploration and sizing of analog circuits. *proc. EDAC*, pages 475–479, 1991.

[17] K. Swings and W. Sansen. Ariadne: A constraint-base approach to computer-aided synthesis and modeling of analog integrated circuits. *Analog Integrated Circuits and Signal Processing*, 3:197–215, 1993.

[18] A. Torralba, J. Chavez, and L. Franquelo. Circuit performance modeling by means of fuzzy logic. *IEEE Transactions on CAD*, 15:1391–8, Nov. 1996.

[19] A. Torralba, J. Chavez, and L. Franquelo. Fasy: a fuzzy logic based tool for analog synthesis. *IEEE Transactions on CAD*, 15:705–15, July 1996.

[20] R. Vemuri, N. Dhanwada, A. Nunez-Aldana, and P. Campisi. Vase: Vhdl-ams synthesis enviroment tools for mixed-signal system. *Proc. EETimes Conference on analog and mixed-signal applications*, pages 1C:77–1C:84, July 1997.