Deterministic Approaches to Analog Performance Space Exploration (PSE)

Daniel Mueller Guido Stehr Helmut Graeb Ulf Schlichtmann

Institute for Electronic Design Automation, Techn. Univ. Muenchen, Arcisstr. 21, 80333 Munich, Germany

ABSTRACT

Performance space exploration (PSE) determines the range of feasible performance values of a circuit block for a given topology and technology. In this paper, we present two deterministic approaches for PSE. One approximates the feasible performance space based on linearized circuit models and is suitable for investigating a large number of performances. The other one computes discretizations of the Pareto front of competing performances. In addition, a motivation and application of PSE using a hierarchical design example is presented.

Categories and Subject Descriptors

B.7.2 [**Hardware**]: Integrated Circuits—*Design Aids*

General Terms: Algorithms, Design

Keywords

Performance Space Exploration, Analog Integrated Circuits, Pareto Optimization, Fourier Motzkin Elimination

1. INTRODUCTION, BASIC CONCEPTS

Analog components play an important role in integrated circuits and systems. Virtually most of the chips designed today contain analog parts, for instance for A/D-D/A signal conversion or clock generation. With the dramatically increasing requirements on design time, quality and complexity, analog parts turn out to be a bottleneck in the design flow. Due to its physical nature, analog design is very difficult to automate, and in addition to structure and layout synthesis, it includes circuit sizing.

1.1 Circuit sizing

Circuit sizing aims at sizing *circuit parameters* $\mathbf{p}$[1] like transistor widths or resistor values such that *circuit performances* $\mathbf{f}$ like gain, bandwidth, slew rate are optimized. Minimum performance requirements are given by permissible values called *specifications* $\mathbf{f}(\mathbf{p}) < \mathbf{f}_B$[2]. Circuit sizing is often referred to as parametric synthesis, nominal design or circuit optimization. If statistical variations of parameters, e.g. due to process fluctuations, are involved and the percentage of circuits satisfying the specs (parametric yield) is to be maximized, circuit sizing is referred to as design centering, yield optimization or tolerance design [29].

1.2 Circuit simulation

Circuit sizing is based on a method to evaluate a set of performances for a set of parameter values:

$$\mathbf{p} \mapsto \mathbf{f}(\mathbf{p}) \tag{1}$$

This mapping is usually done by *circuit simulation* with spice-like analog mixed-signal simulators. Simulations are called very often within a loop during automatic circuit sizing. As circuit simulation is extremely expensive in terms of computational cost, tools for automatic circuit sizing have to be designed to use a minimum number of simulations.

Many methods use performance models instead of circuit simulation for the sizing process [5,6,7,8,9,20,21,22,23,24]. Please note that this includes a preparation phase which again involves a large number of simulations.

1.3 Multi-objective optimization

As a number n of usually competing objectives $f_1 \ldots f_n$ have to be optimized, circuit sizing is a multi-objective optimization[3] problem [11] over the circuit parameters $\mathbf{p}$ such that certain sizing constraints[4] $\mathbf{c}(\mathbf{p}) \geq \mathbf{0}$ are satisfied:

$$\max_{\mathbf{p}} \mathbf{f}(\mathbf{p}) = \begin{bmatrix} f_1(\mathbf{p}) \\ \vdots \\ f_n(\mathbf{p}) \end{bmatrix} \text{ s.t. } \mathbf{c}(\mathbf{p}) \geq \mathbf{0} \quad \rightarrow \quad \mathbf{p}^*, \mathbf{f}^* = \mathbf{f}(\mathbf{p}^*) \tag{2}$$

Usually it is not possible to find a set of parameters for which all performances are maximal. Most of the time, a trade-off situation between performances occurs, where it is only possible to improve one performance at the cost of another.

This introduces the concept of Pareto optimality. A set of performances $\mathbf{a}$ is considered more optimal than a set $\mathbf{b}$ if it dominates $\mathbf{b}$:

$$\mathbf{a} \succ \mathbf{b} \Leftrightarrow \underset{i \in \{1,\ldots,n\}}{\forall} \left(a_i \geq b_i\right) \wedge \underset{i \in \{1,\ldots,n\}}{\exists} \left(a_i > b_i\right) \tag{3}$$

[1] Vectors are denoted in bold letters, matrices in bold capitals, vector inequalities are interpreted element-wise.

[2] Lower permissible values are included in this formulation by multiplication with –1.

[3] Optimization is formulated as maximization. Minimization can be included by maximization of the negative objective value.

[4] Sizing constraints describe the valid parameter space and are crucial for successful optimization of analog circuits. Please see Sec. 7.

DAC 2005, June 13–17, 2005, Anaheim, California, USA.

A set of performances **f*** is Pareto optimal if it is not dominated by any other set **f**. The whole of all Pareto optimal sets of performances make up the *Pareto optimal front*. This front describes the capabilities of a given circuit and the trade-offs between competing performances. In Fig. 1, the curve between $\mathbf{f}^{*1}$ and $\mathbf{f}^{*2}$ illustrates such a Pareto front for two performances f_1 and f_2. The shaded area is the feasible performance space that can be achieved considering the sizing constraints $\mathbf{c}(\mathbf{p}) \geq \mathbf{0}$.

1.3.1 Finding one optimal solution

In the past, many tools for circuit sizing were developed that focused on finding a single point on the Pareto front, e.g. [7,10,12,14,15,29]. Towards this, the multi-objective optimization problem is transformed into a scalar optimization problem through a weighted sum of the performances:

$$\max o(\mathbf{p}) = \sum_{i=1}^{n} w_i f_i(\mathbf{p}) \quad \text{s.t.} \quad \mathbf{c}(\mathbf{p}) \geq 0 \tag{4}$$

A set of weights w_i, $i=1,\ldots n$, leads to a respective solution on the Pareto front.

The solution methods can be classified according to whether they use circuit simulation [12,14,15,29] or performance models [7,10] for performance evaluation, whether they implement a deterministic [7,12,14,29] or stochastic [10,15] optimization process, or whether they include tolerance design [29]. Methods for finding a single Pareto optimal point are mature and available from CAD vendors today, e.g. [30].

1.3.2 Finding many optimal solutions

In recent years, the research focus changed from finding *one* optimal solution to exploring the *whole range* of Pareto optimal solutions for competing performance objectives [1,2,6,8,9,17,18,19,23,25,28].

This is referred to as *design*/*performance space exploration* (PSE). As PSE includes many individual sizing processes, it becomes much more critical concerning the computational cost. Nevertheless, PSE is required in nowadays complex analog-mixed-signal designs of filters, converters, phase-locked loops etc. Due to the increased complexity of such systems, neither a one-step design of the complete system nor the simulation of its transistor netlist is possible. As a consequence, the system is partitioned and hierarchically modeled on behavioral level (e.g. with Verilog-AMS or VHDL-AMS) and circuit level (e.g. with transistor netlists). This mixed behavioral-transistor level modeling of analog systems leads to a *hierarchical* design process, where system specifications are propagated top-down from the behavioral level to the transistor level [2,3,4,13,18,20,22,23,25,26,27,28]. Many of the current works aim at the automation of this well-known hierarchical design process.

PSE provides a complete bottom-up characterization of the performance potential of available topologies for system blocks (e.g. opamps in a filter, oscillator in a phase-locked loop). It is a key to a true hierarchical sizing and design process for two reasons:

First, PSE contributes to an automatic *topology selection*, as it can be applied to compare different variants and select the best one for the given requirements. Second, it provides the achievable space of system parameters and prevents system sizing from producing requirements that cannot be fulfilled by circuit sizing on transistor level. Hence it contributes to an automatic *hierarchical sizing* process [2].

Research these days works on developing tools for PSE which are applicable for a broad range of design problems. However, very often expensive stochastic optimization techniques [9,17,18] and intermediate performance modeling [6,8,19,23,25] are used.

Instead in the following, two deterministic, simulation-based methods for PSE will be given; one aims at accurate results (Sec. 2), the other one at fast results (Sec. 3). Sec. 4 will apply these methods to PSE for topology selection. Sec. 5 will then apply one of these methods to a hierarchical sizing process of a filter. Sec. 6 concludes.

2. PSE BY DISCRETIZED PARETO FRONTS (DISC)

A common way to compute the Pareto front is to calculate discrete points of it by solving the optimization problem (4) with a suitable sweep over the weights w_i. The challenge lies in selecting sets of weights such that the Pareto front is covered with as less discrete points as possible. This is a difficult task, as the curvature of the Pareto front may differ largely from design to design and as it may be non-convex. In [1], a different approach has been described that is illustrated in Fig. 1 for two performances f_1 and f_2.

In a first step, the maximum achievable values of all individual performances are determined. Although this is a scalar optimization problem, it imposes the challenge to find the global optima, in order to cover the full Pareto curve. In Fig. 1, the individual optima are marked by lines intersecting the f_1 and f_2 axes. The corresponding vectors of all performance values at each individual optimum are denoted by $\mathbf{f}^{*1}$ and $\mathbf{f}^{*2}$. The convex hull between the individual optima, geometrically this is the hyperplane spanned between the individual optima, is described by:

$$\mathbf{F} \cdot \mathbf{w} = \begin{bmatrix} \mathbf{f}^{*1} & \mathbf{f}^{*2} \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \quad ; \quad w_{1,2} \geq 0, w_1 + w_2 = 1 \tag{5}$$

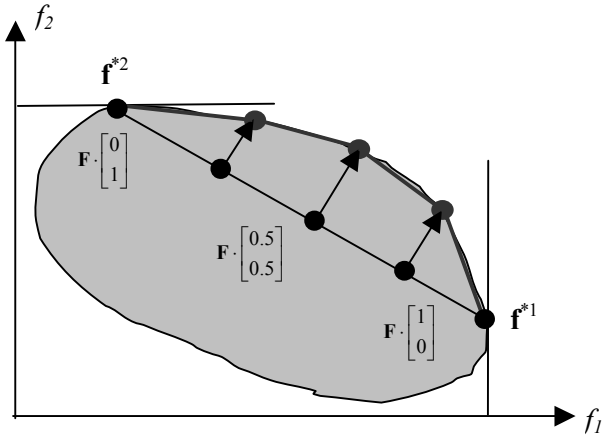

Figure 1. Computing the Pareto front by searching along normal rays on the intersected convex hull between the individual optima of performances.

This convex hull is equally intersected as illustrated in Fig. 1. Along the normal ray of each of the obtained discrete points of the convex hull, described in parametric form by the parameter t and a normal vector $\mathbf{n}$, the following optimization problem is solved to obtain the DISCrete points of the Pareto front of performances.

$$\max_{\mathbf{p},\lambda} \ \lambda \ \text{s.t.} \ \mathbf{F}\cdot\mathbf{w}+\lambda\cdot\mathbf{n}=\mathbf{f}(\mathbf{p}) \wedge \mathbf{c}(\mathbf{p})\geq\mathbf{0} \qquad (6)$$

The solution of (6) can be obtained by suitable optimization algorithms based on circuit simulation for performance evaluation [1]. This method can also handle non-convex Pareto fronts and is suitable to compute an accurate discretization of the Pareto fronts for a relatively small number of performances.

3. PSE BY POLYTOPAL APPROXIMATION (POLY)

In this section, we will outline a different approach for PSE, which is more efficient at the price of lower accuracy. It is based on a linearized circuit model and a POLYtopal approximation of the *whole* feasible performance space. A detailed description of this method can be found in [2].

The basic task of PSE can be interpreted as transferring the description of the valid parameter space (sizing constraints) $\mathbf{c}(\mathbf{p}) \geq \mathbf{0}$ using performance evaluation (circuit simulation) $\mathbf{f} = \mathbf{f}(\mathbf{p})$ into a description of the feasible performance space $\mathbf{k}(\mathbf{f}) \geq \mathbf{0}$ that does not depend on parameters (i.e. is implementation-independent):

$$\mathbf{c}(\mathbf{p})\geq\mathbf{0} \wedge \mathbf{f}=\mathbf{f}(\mathbf{p}) \rightarrow \mathbf{k}(\mathbf{f})\geq\mathbf{0} \qquad (7)$$

This can be done by introducing a linearized approximation for the nonlinear constraint and performance functions $\mathbf{c}$ and $\mathbf{f}$ and calculating a linear model for the feasible performance space:

$$\mathbf{C}\cdot\Delta\mathbf{p}\geq\mathbf{c}_0 \wedge \mathbf{F}\cdot\Delta\mathbf{p}=\Delta\mathbf{f} \rightarrow \mathbf{K}\cdot\Delta\mathbf{f}\geq\mathbf{k}_0 \qquad (8)$$

Usually, the number of parameters n_p is larger than the number of performances n_f, hence we introduce a suitable partitioning of parameters and rewrite (8):

$$\mathbf{C}_{1<n_c\times n_f>}\cdot\Delta\mathbf{p}_{1<n_f>}+\mathbf{C}_{2<n_c\times n_p-n_f>}\cdot\Delta\mathbf{p}_{2<n_p-n_f>}\geq\mathbf{c}_0 \qquad (9a)$$

$$\mathbf{F}_{1<n_f\times n_f>}\cdot\Delta\mathbf{p}_{1<n_f>}+\mathbf{F}_{2<n_f\times n_p-n_f>}\cdot\Delta\mathbf{p}_{2<n_p-n_f>}=\Delta\mathbf{f} \qquad (9b)$$

Inserting $\Delta\mathbf{p}_1$ from (9b) into (9a), assuming that $\mathbf{F}_1$ is not rank-deficient, leads to:

$$\mathbf{C}_1\mathbf{F}_1^{-1}\cdot\Delta\mathbf{f}+\left(\mathbf{C}_2-\mathbf{C}_1\mathbf{F}_1^{-1}\mathbf{F}_2\right)\cdot\Delta\mathbf{p}_2\geq\mathbf{c}_0 \qquad (10)$$

Based on Fourier-Motzkin elimination and a sophisticated redundancy detection [2], (10) can be transformed into the required linear description $\mathbf{k}(\mathbf{f}) \geq \mathbf{0}$ of the feasible performance space. This method is suitable to efficiently approximate the feasible performance space for a large number of performances. In the following section, the deterministic PSE methods DISC and POLY presented in this section and Sec. 2 will be applied to two opamp topologies and compared with respect to accuracy and efficiency.

4. TOPOLOGY SELECTION

In general, the designer can choose between several architectures or topologies to build an analog circuit with the desired functionality. At that stage of the design process, usually no circuit sizing is available and the designer has little information about the different performance values achievable by each topology. PSE helps him make this choice. In the following, this is illustrated for two opamp topologies, which are compared in terms of speed and power consumption. The opamps shall meet the following AC specifications: phase margin greater than 60 degrees to guarantee stability, DC gain greater than 75dB and transit frequency greater than 10 MHz. These application-specific performance specifications can be handled equally to sizing constraints during the exploration process. Fig. 2 shows the results for the two approaches introduced in the previous sections.

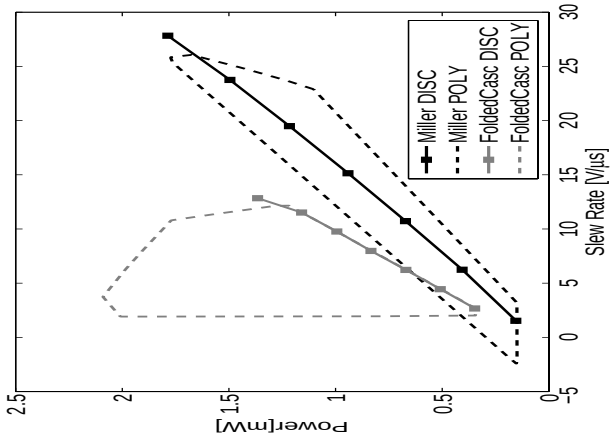

Figure 2. PSE results for two opamp topologies

We can see that the polytopal approximation is able to capture the qualitative from of the Pareto front quite well. In this case, the Miller opamp is superior in terms of speed and power consumption. This may not be true for a different set of AC specifications. Especially if a high phase margin is demanded, the Folded-Cascode topology becomes superior. The corresponding graph is not shown here.

Table 1 shows that PSE based on polytopal approximation by POLY is significantly faster compared to the discretization by DISC. Experience showed that the accuracy of this method depends heavily on a suitable choice of the linearization point and that the approximation of the border of the feasible performance space is more reliable if it is close to this point.

Table 1. Computation time on a cluster of 15 Pentium IV

Topology	DISC	POLY
Miller	18 min 25 sec	3 sec
Folded Cascode	55min 38 sec	1min 45 sec

5. HIERARCHICAL SIZING

With the example of a bandpass filter we want to illustrate a hierarchical sizing algorithm based on behavioral models and PSE. Fig. 3 shows the system level model of an OTA-C biquad filter. In this case, the system level model is a structure consisting of eight operational transconductance amplifiers (OTAs) and two capacitors C_1 and C_2 as system blocks.

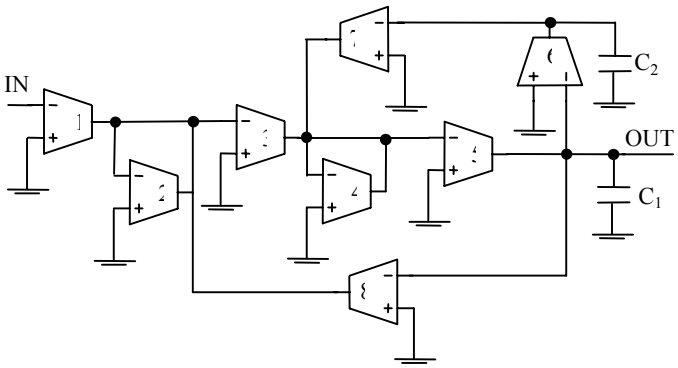

Figure 3. System: OTA-C biquad filter, system level (structural) model

The *system performances* and *system specifications* (AC behavior) of the filter are given in Table 2. The system block OTA is modeled with VHDL-AMS based on the model given in [5]. This behavioral model[5] is given in Table 3. It features 5 *system design parameters*, given in Table 4.

Table 2. Filter, system performances and specifications

center frequency f_c	f_c = 2 MHz
center frequency gain G_c	G_c > 20 dB
quality factor Q	Q > 15

Table 3. OTA, system level (behavioral) model

```
ENTITY ota is
GENERIC (gdm:real;Cin:real;CT:real;Cout:real;Rout:real;);
PORT ( terminal Tplus, Tminus, Tout, Tvdd, Tvss , Tibias:
electrical);
END ENTITY ota;
-- Simple Ota architecture
architecture behavioural_simple of ota is
---***external quantities***
...
---***internal quantities***
...
begin  -- behavioural_simple
---***input stage***
  icd == cin*vd'dot;
  vd == ird*rin;
---***transfer stage***
  iT == vd / rT;
  iT == icT + irT;
  vT == irT*rT;
  icT == cT*vT'dot;
---***output stage***
  vout == irout*rout;
  icout == cout*vout'dot;
  iout0 == vT *gdm;
  iout == iout0+icout+irout;
end behavioural_simple;
```

[5] Analogously, a system level structural macromodel of an OTA could be given by a three-stage equivalent circuit.

Table 4. Filter, system design parameters

OTA transconductances g_{dm}	OTA output resistances R_{out}
OTA input capacitances C_{in}	OTA capacitances C_{out}
OTA capacitances C_T	C_1, C_2

The transistor implementation of an OTA has an offset in the phase $\varphi(f)$ of its transfer function, which has a dominant influence on the system behavior. C_T models this offset with the following fitting function:

$$C_T = \frac{1}{-2\pi f_c R_T \tan(\varphi(f_c))} \tag{7}$$

The system model parameters g_{dm}, C_{in}, C_T, R_{out} and C_{out} can be obtained from simulation of the OTA on circuit level. That means that they also constitute the *circuit performances* of a system block OTA. The circuit schematic of an OTA is given in Fig. 4.

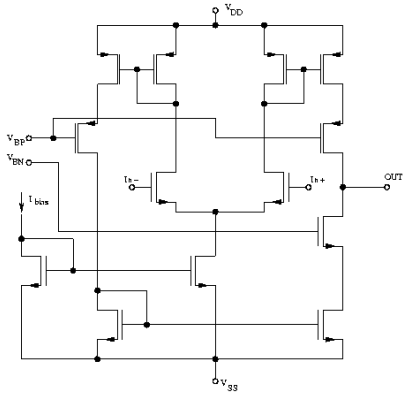

Figure 4. OTA, circuit level (transistor netlist) model

For each OTA, eight independent transistor widths and lengths form the *circuit design parameters*. As two OTA pairs are sized identically due to symmetries in the design, we have 48 circuit parameters. Together with the two system design parameters C_1 and C_2 we have 50 design parameters in total to be sized.

A simulation of a transistor implementation of the filter is a two-stage process. For a given set of transistor widths, all OTAs are first simulated on circuit (transistor) level with the netlist in Fig. 4. The resulting circuit performances are part of the system design parameters according to Table 4 and input to the OTA system level (behavioral) model in Table 3 using (7). The filter is simulated in a second step with the system model in Fig. 3 and Table 3 in order to obtain the filter performances.[6]

In the following, we illustrate the hierarchical sizing process.

As demonstrated, the OTA blocks are modeled by behavioral models on system level. These models depend on system design parameters (Table 4) that are independent of transistor geometries and therefore implementation-independent.

[6] Although this example could as well be simulated flat on transistor level, the two-step procedure over behavioral and transistor level is usually mandatory because the system complexity induces prohibitive flat simulation cost.

Table 5. OTA-C filter during hierarchical sizing with PSE

System performance after	f_c [MHz]	G_c [dB]	Q
system sizing	2.001	20.64	16.23
circuit sizing	2.001✓	20.23✓	15.46✓

Table 6. OTA5 during hierarchical sizing with PSE

	g_m [µS]	C_{in} [fF]	$\varphi(f_c)$ [deg]	R_{out} [MΩ]	C_{out} [fF]
Spec after system sizing	205.1	32.94	174.0	184.4	21.6
Achieved after circuit sizing	204.9	30.3	174.0	184.6	21.8

Table 7. OTA-C filter during hierarchical sizing without PSE

System performance after	f_c [MHz]	G_c [dB]	Q
system sizing	1.998	20.01	21.14
circuit sizing	1.972✓	14.98↘	12.34↘

Table 8. OTA5 during hierarchical sizing without PSE

	g_m [µS]	C_{in} [fF]	$\varphi(f_c)$ [deg]	R_{out} [MΩ]	C_{out} [fF]
Spec after system sizing	299.7	32.57	175.3	229.4	22.29
Achieved after circuit sizing	297.1	28.57	174.2	122.4	28.57

By sizing on system level, a set of optimal system design parameters is obtained in a first step. This set represents the behavior of the underlying system block which guarantees optimal performance of the complete filter. This set can therefore be seen as a list of specifications which are passed down to the blocks in a hierarchical top-down sizing approach [2,3]. On circuit level, the optimal system design parameters constitute circuit level specifications. In a second step, circuit level sizing is conducted on each OTA separately and simultaneously, using its transistor level netlist. This leads to a set of optimal circuit design parameters.

Sizing of the filter on behavioral level usually tends to lead to unrealistic system model parameters, producing overly ambitious specifications, which can not be met by the underlying OTA circuit. As this turns out only after the circuit sizing step, unnecessary "resizing" loops are the consequence. Therefore the information of the capabilities of the OTA must be passed up to the behavioral level to avoid unrealistic specifications. In [4], this is done by optimizing on behavioral level for highest design flexibility, where design flexibility is a heuristic measure describing the capability of the underlying circuit blocks to meet their specification.

The task of avoiding unrealistic specifications can be accomplished in a systematic way by applying PSE techniques. The obtained feasible performance space of an OTA can be interpreted as sizing constraints on system level similar to the sizing constraints on circuit level. This leads to a closed and complete automatic hierarchical sizing methodology.

The proceeding is illustrated with the OTA-C filter. We use the PSE approach based on polytopal approximation in Sec. 3 to calculate the sizing constraints on system level.

After performing the system level sizing, we obtain the system level performance as given in the first row of Table 5. E.g. for the OTA5 we obtain the optimal system parameters given in the first row of Table 6, which are passed to the circuit level sizing process as specifications.

Using the specifications obtained from system level sizing (e.g. those in the first row of Table 6 for OTA5), circuit level sizing for all OTAs is performed. E.g. for the OTA5, we achieve the circuit performances given in the second row of Table 6. Using the achieved circuit performances for all OTAs in system level simulation, we obtain the system performance in the second row of Table 5.

We can see, that the result is fine. Often however, the system performance after circuit level sizing is not satisfying. This is the result of the inaccuracy induced by the POLY approximation of the system level sizing constraints. As a consequence, the system level sizing may drive circuit parameters into regions which are actually not achievable on circuit level. In that case, the top-down sizing process is iteratively repeated as described in [2].

The significance of the system level sizing constraints obtained by PSE can be demonstrated by running hierarchical sizing without them. The values of the system level parameters are now only limited by realistic box constraints. Table 7 shows the results. The system level sizing without PSE finds a good solution, but the system performances are poor after circuit level sizing. This is due to overly ambitious specifications passed down to the OTAs, which cannot be achieved by circuit sizing. For example we obtain the specifications shown in the first row of Table 8 for OTA 5. A relative high g_m is demanded for this OTA. This specified g_m can be obtained by circuit sizing as shown in the second row but on the expense of a lower $\varphi(f_c)$ and less R_{out} as specified. There is a trade-off between g_m, $\varphi(f_c)$ and R_{out} of the OTAs. Obviously if this information is not made available to the system sizing algorithm by PSE system level constraints, resizing loops become necessary.

Conclusion

We have described two deterministic, simulation-based methods for performance space exploration (PSE) and have applied them to hierarchical sizing. These two PSE methods feature the ability to use circuit simulation (in order to capture the exact physical circuit behavior) at reasonable computational cost. With the example of hierarchical sizing of a filter, it has been demonstrated that PSE provides the means for a first-time successful top-down sizing.

6. Appendix: Sizing constraints

Sizing constraints $\mathbf{c}(\mathbf{p}) \geq \mathbf{0}$ are crucial for the success of analog circuit optimization [16] and form the basis to calculate the feasible performance space and for performance space exploration as in [1,2]. If not considered, optimization most often will fail to converge or to produce robust sizings. Sizing constraints describe the valid parameter space of transistor pairs and basic analog building blocks. These constraints refer to transistor widths and lengths and to DC properties of transistors and implement basic design knowledge e.g. concerning matching, transistor operation. Some approaches define box constraints for parameters [6,13], many methods for sizing just mention that sizing constraints exist [9,10]. In [16], a systematic library of sizing constraints and a tools for automatic structural analysis of analog circuits and creation of sizing constraints has been presented. Advanced sizing constraint

methods like [16] are technology-specific and automatically provide the actual sizing constraints for a given circuit topology.

REFERENCES

[1] G. Stehr, H. Graeb and K. Antreich. Performance Trade-off Analysis of Analog Circuits By Normal-Boundary Intersection. *ACM/IEEE DAC 2003.*

[2] G. Stehr, H. Graeb and K. Antreich. Analog Performance Space Exploration by Fourier-Motzkin Elimination with Application to Hierarchical Sizing. *IEEE ICCAD 2004.*

[3] H. Chang, E. Charbon, U. Choudhury, A. Demir, E. Felt,E. Liu, E. Malavasi, A. Sangiovanni-Vincentelli, and I. Vassiliou. A Top-Down, Constraint-Driven Design Methodology for Analog Integrated Circuits. *Kluwer Acad. Pub. 1997.*

[4] I. Vassiliou, H. Chang, A. Demir, E. Charbon, P. Miliozzi and A. Sangiovanni-Vincentelli. A Video Driver System Designed Using a Top-Down, Constraint-Driven Methodology. *IEEE ICCAD 1996.*

[5] G.J. Gomez, S.H.K. Embabi, E. Sanchez-Sinencio and M.C. Levebvre. A nonlinear macromodel for CMOS OTAs. *IEEE ISCAS 1995.*

[6] F. De Bernardinis, M. I. Jordan, and A. Sangiovanni-Vincentelli. Support vector machines for analog circuit performance representation. *ACM/IEEE DAC, 2003.*

[7] M. del Mar Hershenson, S. P. Boyd, and T. H. Lee. Optimal design of a CMOS Op-Amp via geometric programming. *IEEE Trans. CAD* 2001.

[8] R. Harjani and J. Shao. Feasibility and performance region modeling of analog and digital circuits. *Analog Integrated Circuits and Signal Processing* 1996.

[9] B. D. Smedt and G. G. E. Gielen. WATSON: Design space boundary exploration and model generation for analog and RF IC design. *IEEE Trans. CAD* 2003.

[10] G. Van der Plas, G. Debyser, F. Leyn, K. Lampaert, J. Vandenbussche, G. Gielen, W. Sansen, P. Veselinovic, and D. Leenaerts. AMGIE.A synthesis environment for CMOS analog integrated circuits. *IEEE Trans. CAD 2001.*

[11] M. Lightner, T. Trick, and R. Zug. Circuit optimization and design. *Circuit Analysis, Simulation and Design, Part 2 (A. Ruehli). Advances in CAD for VLSI 3*, *1987.*

[12] G. Stehr, M. Pronath, F. Schenkel, H. Graeb, and K. Antreich. Initial sizing of analog integrated circuits by centering within topology-given implicit specifications. *IEEE ICCAD* 2003.

[13] A. Fuad Mas'ud, T. Ohtsuka, and H. Kunieda. Translation of specifications in hierarchical analog LSI design. In *IEEE ISCAS* 1996.

[14] W. Nye, D. Riley, A. Sangiovanni-Vincentelli, and A. Tits. DELIGHT.SPICE: An optimization-based system for the design of integrated circuits. *IEEE Trans. CAD 1988*.

[15] R. Phelps, M. Krasnicki, R. Rutenbar, L.R. Carley, and J. Hellums. Anaconda: Simulation-based synthesis of analog circuits via stochastic pattern search. *IEEE Trans. CAD 2000*.

[16] H. Graeb, S. Zizala, J. Eckmueller, and K. Antreich. The sizing rules method for analog integrated circuit design. *IEEE/ACM ICCAD 2001*.

[17] A Somani, P P Chakrabarti and A Patra. Mixing global and local competition in genetic optimization-based design space exploration of analog circuits, *DATE 2005*.

[18] T Eeckelaert, T McConaghy and G Gielen. Efficient multiobjective synthesis of analog circuits using hierarchical Pareto-optimal hypersurfaces, *DATE 2005.*

[19] Maria del Mar Hershenson. Efficient Description of the Design Space of Analog Circuits. *ACM/IEEE DAC 2003*.

[20] M. Vogels, G. Gielen. Architectural Selection of A/D Converters. *ACM/IEEE DAC 2003*.

[21] Hongzhou Liu, Amit Singhee, Rob A. Rutenbar, L. Richard Carley. Remembrance of Circuits Past: Macromodeling by Data Mining in Large Analog Design Spaces. *ACM/IEEE DAC 2002*.

[22] J. Vandenbussche, K. Uyttenhove, E. Lauwers, M. Steyaert and G. Gielen. Systematic Design of a 200 Ms/s 8-bit Interpolating/Averaging A/D Converter. *ACM/IEEE DAC 2002*.

[23] F De Bernardinis, S Gambini, F Vincis, F Svelto, R Castello, A Sangiovanni Vincentelli. Design Space Exploration for a UMTS Front-end Exploiting Analog Platforms. *IEEE ICCAD 2004.*

[24] Glenn A Wolfe, Ranga Vemuri. Adaptive Sampling and Modeling of Analog Circuit Performance Parameters with Pseudo-Cubic Splines. *IEEE ICCAD 2004.*

[25] O. Bajdechi, G. Gielen, J. Huijsing. Optimal Design of Delta-Sigma ADCs by Design Space Exploration. *ACM/IEEE DAC 2002*.

[26] S. Ganesan, R. Vemuri. Behavioral Partitioning in the Synthesis of Mixed Analog-Digital Systems, *ACM/IEEE DAC 2001*.

[27] J. Dawson, S. Boyd, M. del Mar Hershenson, T. Lee. Optimal Allocation of Local Feedback in Multistage Amplifiers via Geometric Programming. *IEEE Trans. CAS-I 2001*.

[28] S.K. Tiwary, S.Velu, R.A. Rutenbar and T. Mukherjee: Pareto-Optimal Modeling for Efficient PLL Optimization. *NSTI Nanotech 2004, Boston*

[29] K. Antreich, H. Graeb, C. Wieser: Circuit analysis and optimization driven by worst-case distances. *IEEE Trans. CAD 1994.*

[30] www.muneda.com