The Sizing Rules Method for CMOS and Bipolar Analog Integrated Circuit Synthesis

Tobias Massier, Helmut Graeb, *Senior Member, IEEE*, and Ulf Schlichtmann, *Member, IEEE*

***Abstract*—This paper presents the sizing rules method for basic building blocks in analog CMOS and bipolar circuit design. It consists of the development of a hierarchical library of transistor-pair groups as basic building blocks for analog CMOS and bipolar circuits, the derivation of a hierarchical generic list of constraints that must be satisfied to guarantee the function and robustness of each block, and the development of a reliable automatic recognition procedure of building blocks in a circuit schematic. Sizing rules efficiently capture design knowledge on the technology-specific level of transistor-pair groups. This reduces the effort and improves the resulting quality for analog circuit synthesis. Results of applications like circuit sizing, design centering, response surface modeling, or analog placement show the benefits of the sizing rules method.**

***Index Terms*—Analog design, analog synthesis, bipolar, circuit sizing, CMOS, sizing rules.**

I. INTRODUCTION

ANALOG components are an important part of integrated systems: either in terms of elements and area in mixed-signal systems or as vital parts in digital systems, for instance, power-on reset, pad driving, or clock generation. Despite their importance, the design automation for analog circuits still lags behind that of digital circuits. As a consequence, analog components are often the bottleneck in the design flow.

Circuit synthesis is complicated because it does not only consist of topology and layout synthesis but also of component sizing. Additionally, it has to incorporate physical effects like process variations, variations of operating conditions, matching constraints, or noise. Since the 70s, analog topology synthesis [1]–[4], nominal design optimization [5]–[7], and sizing with respect to tolerances (design centering, yield optimization) [8]–[13] were in the focus of research interest. The approaches include equation-based methods like GPCAD [14] or AMGIE [15], where design equations are derived with the help of symbolic analysis [16] and simulation-based methods like ASTRX/OBLX [12], [17]–[20]. Sizing tasks have a key potential for providing automation support to the designer [21], particularly when consistently considering process and operating tolerances and mismatch [22]–[27].

Manuscript received December 18, 2007; revised May 10, 2008. Current version published November 19, 2008. This paper was recommended by Associate Editor J. R. Phillips.

T. Massier is with the Department of Electrical Engineering and Information Technology, Institute for Electronic Design Automation, Technische Universität München, 80333 Munich, Germany (e-mail: massier@tum.de).

H. Graeb and U. Schlichtmann are with the Institute for Electronic Design Automation, Technische Universität München, 80333 Munich, Germany (e-mail: graeb@tum.de; ulf.schlichtmann@tum.de).

Digital Object Identifier 10.1109/TCAD.2008.2006143

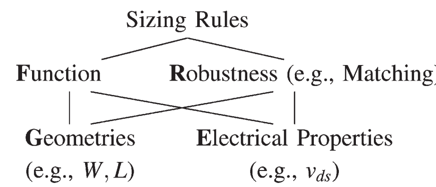

Fig. 1. Types of sizing rules.

A major obstacle of automatic sizing, in practice, is the often incomplete circuit specification. Specifying circuit performance bounds, e.g., for dc gain, slew rate, or phase margin of an operational amplifier is not sufficient to prevent mathematical optimizers from driving the circuit into technically meaningless regions. Often, the resulting circuit performs regularly in the nominal case, but exhibits increased sensitivity to process and operating variations and to noise [24] (see Fig. 11). Hence, additional constraints—so-called sizing rules—for transistor geometry or transistor voltages have to be considered, e.g., to prevent CMOS transistors from leaving the saturation region (Fig. 1). Sizing rules are formulated as equality or inequality constraints for transistor geometry parameters (transistor width, length, area) and for electrical transistor quantities (e.g., transistor drain-source voltage) (1). They can be checked during simulation-based analog synthesis without simulation overhead. Both types of sizing rules represent an optimal compromise between design performance (e.g., gain) and process yield. In particular, sizing rules guarantee the proper function of a building block and its robustness, e.g., toward mismatch or channel length modulation.

Most of the approaches to analog synthesis mention, e.g., "design-space constraints" [28], "dimension constraints" [14], "manufacturability and operationality constraints" [23], "component constraints" [25], [29], or "soft constraints" [30].

A first approach that presented a detailed automatic construction of sizing rules for CMOS transistor circuits was presented in [31]. While analog design literature, e.g., [33]–[37], focuses on the constructive part of design knowledge, which aims at creating equations to propagate circuit specifications "top–down" to parameter values, the constraint part characterizes "bottom–up" conditions that have to be satisfied for a manufacturable design. This constraint part will be gathered in the form of sizing rules, which are established for basic building blocks of analog circuits on *transistor-pair level* (Table I). Since every analog circuit is based on transistor-pair building blocks like, for example, current mirrors, this allows us to capture a major portion of design constraint knowledge in a *design-independent technology-specific* manner. In [32], sizing rules for bipolar transistor circuits were presented.

TABLE I
ANALOG DESIGN LEVELS

Level	Example	
System	DAC, ADC, PLL	design-specific
Circuit	OpAmp	
Transistor Pair	**Current Mirror, Differential Stage**	**technology-specific**
Device	Transistor	

This paper is based on the material presented in [31] and [32] with two novel contributions. First, a mathematical formulation of the circuit recognition process is given. Second, compared to [31], this paper presents an extended sizing rules method. An important problem of the building block recognition has not been mentioned in [31] and [32]: the many ambiguities in assigning transistors to pairs and higher order groups of transistors. This paper presents a novel heuristic methodology for the arbitration of assignment ambiguities.

The remainder of this paper is organized as follows. Section II develops a hierarchical library of basic building blocks for CMOS and bipolar technology, based on transistor pairs. A procedure for the automatic hierarchical recognition of these building blocks is presented in Section III. Section IV introduces a heuristic methodology for the arbitration of ambiguous module assignments. In Sections V and VI, sizing rules for CMOS and bipolar transistor building blocks according to functional and robustness constraints are presented. In Section VII, results and applications are shown. Section VIII concludes this paper. Section IX mentions some future work.

II. HIERARCHICAL BUILDING BLOCK LIBRARY

Figs. 2 and 3 show hierarchical libraries L_{CMOS} and L_{Bipolar} of basic building blocks for CMOS and bipolar transistor technologies. The whole library which is denoted by L is the union of both libraries. A hierarchical library is a strictly ordered set. It can be divided into subsets that represent a hierarchical level each. The elements on each hierarchy level from 1 upward consist of elements of lower hierarchy levels. For example, a cascode current mirror (CCM) on level 2 consists of two building blocks from hierarchy level 1, i.e., a level shifter and a simple current mirror, denoted by "ls" and "cm." A Wilson current mirror (WCM) on level 2 consists of building blocks from different hierarchy levels, i.e., one building block from hierarchy level 1 (a simple current mirror) and a single transistor from level 0. The components of transistor pairs (on level 1) are numbered. Number (1) refers to the left or upper transistor, number (2) on the right or lower one. These numbers will be referred to in Sections V and VI where the sizing rules will be presented in detail. There are $B_6 = 203$ possible transistor pair structures, where B_k is the Bell number [38]. Most of them are technically senseless. Figs. 2 and 3 contain only those transistor pairs on hierarchy level 1, which either produce sizing rules on their own or are contained in building blocks on higher hierarchy levels. In this sense, the list of transistor pairs on level 1 is complete.

Beginning with the CMOS library, the lowest level 0 contains a single transistor which operates either in saturation or triode

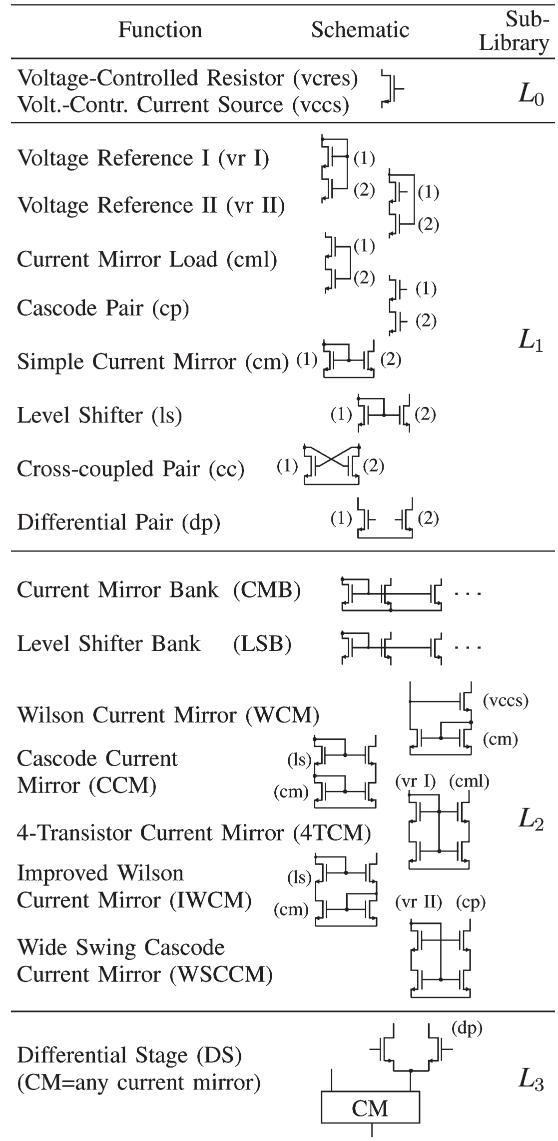

Fig. 2. Hierarchical library of basic CMOS building blocks (NMOS).

region. On level 1, we have identified eight transistor pairs. To four of the building blocks on level 1, namely, the simple current mirror, the ls, the cross-coupled pair (cc), and the differential pair (dp), sizing rules assuring correct functionality apply immediately. On level 2, several current mirrors consisting of three or four transistors and two types of banks are defined. On level 3, the differential stage (DS) is located. It consists of a dp and an arbitrary current mirror (CM). Banks of current mirrors from level 2 also appear on level 3 but are not shown.

The bipolar building block library in Fig. 3 is organized the same way. On level 0, the single bipolar transistor is listed in forward active region only since all of the building blocks on higher levels require the contained transistors to operate in this region. On level 1, six transistor pairs can be found. Four of them have the same connectivity as those in the CMOS library. The Darlington configuration appears once with common collector, once without. Like in the CMOS library, hierarchy level 2 contains mainly current mirrors consisting of three or

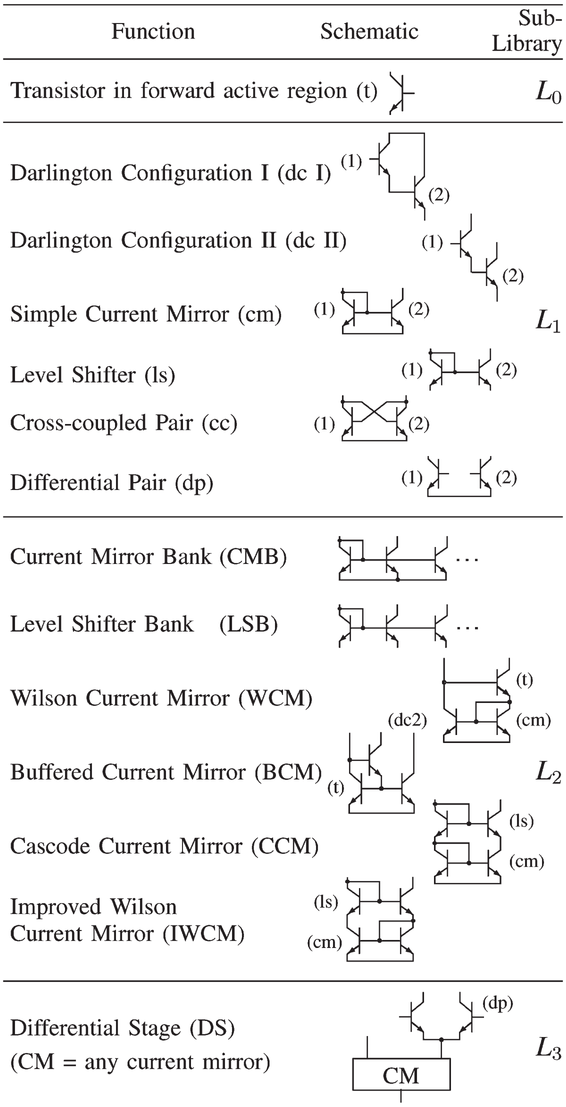

Fig. 3. Hierarchical library of basic bipolar building blocks (n-p-n).

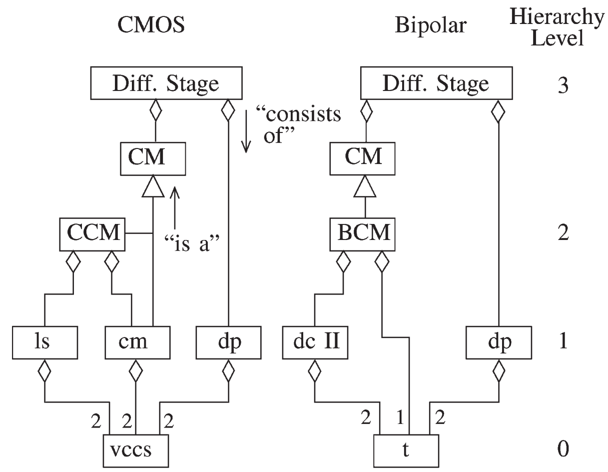

Fig. 4. Hierarchical library of building blocks in UML notation (extract).

four transistors. Level 3 includes the DS, as well as banks of current mirrors from level 2 which are not shown.

Resistors are often added to the emitter pins of bipolar transistors, producing new building blocks (e.g., a Widlar current mirror arises from a simple current mirror by adding a resistor on the right). These building blocks are not shown, since the basic structure is still the same. During the recognition process, a transistor with a resistor is treated as one circuit element.

Block schematics and sizing rules are given for the NMOS and n-p-n parts and hold analogously for the PMOS and p-n-p parts. A hierarchy of building blocks results from the structural property that basic functions are realized based on transistor pairs, groups of transistor pairs, or transistor pairs combined with single transistors.

Of course, these libraries are not complete for levels above 1, and a variety of other building blocks can be included. However, they represent a majority of used building blocks and can be considered standard building block libraries (which could be used as libraries for schematic entry during topology design). The described libraries shall be used to introduce sizing rules to the sizing part of analog synthesis. For that, we need a generic list of constraints for each building block that result from function and robustness requirements (i.e., a generic list of sizing rules). Sizing rules will be equalities and inequalities of transistor geometries and dc quantities and can be checked during simulation-based analog synthesis without simulation overhead. We additionally provide a procedure that searches a given circuit schematic to identify all building blocks contained in it. For each identified building block, the respective list of sizing rules is instantiated. Generally, even for small circuits, a large number of detected building blocks are found, which in turn, leads to a large number of sizing rules.

This modeling enables the automatic recognition within a circuit structure and, upon this, the assignment of sizing rules. In the following sections, a procedure for the automatic recognition of building blocks which includes the arbitration of assignment ambiguities is presented first, and generic sizing rules are introduced afterward.

III. Automatic Hierarchical Recognition of Building Blocks

Fig. 4 exemplarily shows an extract of the hierarchical library $L = L_{\mathrm{CMOS}} \cup L_{\mathrm{Bipolar}}$ from Figs. 2 and 3 in unified modeling language (UML) notation [39]. For instance, for the CMOS part, it reads: A DS "consists of" a current mirror (CM) and a dp; a CCM "is a" current mirror (CM).

It can be seen that the libraries L_{CMOS} and L_{Bipolar} are organized bottom–up—from a single transistor to multitransistor building blocks. The main benefit from the hierarchically organized pairwise block building in L is that new building blocks can be formed by combining existing ones. Hence, the assignment of sizing rules is rather simple since most building blocks inherit the rules from the building blocks they consist of. That means, if a new building block is added to the library, only few new sizing rules have to be added. In theory, the recognition algorithm corresponds to a search for subgraph isomorphisms, which is known to be NP-complete. The subgraphs correspond to the schematics of the building block set defined according to Figs. 2 and 3. The computational cost of the recognition algorithm is only low as the number of transistors and building blocks in practical analog circuits is rather small, and we only search for pairs of elements. In a given circuit schematic, the recognition algorithm detects all building blocks that correspond to the respective elements in the library L, going through the hierarchy from simple building blocks on low levels to more complex ones on higher levels.

First, a mathematical representation of the presented building block library is given. The library L containing all building blocks shown in Figs. 2 and 3 is a strictly ordered set. With the exception of a single transistor, all elements of L consist of other elements of L. Therefore, we can group the building blocks into different hierarchy levels with a single transistor at the bottom. Set L can also be formulated as the union of all sublibraries L_i which contain all building blocks on level i each

$$L = \bigcup_{i=0}^{h_{\max}} L_i. \tag{1}$$

Here, $h_{\max}$ is the number of the highest hierarchy level, which is three. To be able to classify the building blocks, we introduce the set of types Y

$$Y \subseteq YT \times YS. \tag{2}$$

Set YT is the set of transistor types, YS the set of structural types of building blocks in our library

$$YT = \{\text{NMOS}, \text{PMOS}, \text{npn}, \text{pnp}, \ldots\} \tag{3}$$

$$YS = \{\text{transistor}, \text{vr I}, \text{vr II}, \ldots, \text{DS}\}. \tag{4}$$

For example, an NMOS dp is of type (NMOS, dp). Set Y contains all types of building blocks that can be found in the presented library and is therefore a subset of $YT \times YS$. This definition also includes hybrid types, e.g., a DS consisting of bipolar and CMOS transistors. The type of a library element $l \in L$ is denoted by $l.type$. Additionally, $l.trantyp$ and $l.strtyp$ are used to denote transistor (from set YT) and structural types (from set YS). Furthermore, for all library elements above level 0, $l(1) \in L$ and $l(2) \in L$ denote the subelements which l consists of. Index 1 denotes the left or upper subelement of the building blocks in Figs. 2 and 3, index 2 the right or lower subelement. Accordingly, the subelements of library elements on hierarchy level 1 have been marked with their index, (1) or (2), in Figs. 2 and 3.

Second, a formal representation of a circuit netlist and its contents is given. A netlist contains the elements in a circuit and their pin connections. Any circuit element or building block that is included in the hierarchical library in Figs. 2 and 3 will be called a "module" in the following.

We define M as the set of all modules m_μ in a circuit

$$M = \{m_\mu | \mu = 1, 2, \ldots, |M|\}. \tag{5}$$

After the circuit netlist has been read in, set M contains only building blocks on level 0, i.e., single CMOS or bipolar transistors. During the recognition process, building blocks consisting of two or more transistors will be added to M. Set M can be considered a union of subsets. For instance, like set L, set M arises from the union of subsets that contain modules from a single hierarchy level each

$$M = \bigcup_{i=0}^{h_{\max}} M_i. \tag{6}$$

In addition, M can be formulated as the union of subsets that only contain modules of a certain type as defined by set Y

$$M = M_{(\text{NMOS},\text{trans})} \cup M_{(\text{PMOS},\text{trans})} \cup \cdots \cup M_{(\text{pnp},\text{DS})}. \tag{7}$$

We will make use of all three forms of representation.

Next, we define the set N of nets connecting the modules

$$N = \{n_\nu | \nu = 1, 2, \ldots, |N|\}. \tag{8}$$

The set of pins P_μ of a single module m_μ is defined as

$$P_\mu = \{m_\mu.p_\psi | \psi = 1, 2, \ldots, |P_\mu|\}, \quad p_\psi \in \{d, g, s, c, b, e, \ldots\}. \tag{9}$$

Note that the names of the pins are used for illustration here. For instance, $m_1.p_\psi = m_1.d$ could denote the drain pin of a CMOS transistor named m_1. For building blocks on higher hierarchy levels, special pin names like "common source" are used. Set P of all pins is the union of all sets P_μ

$$P = \bigcup_{\mu=1}^{|M|} P_\mu. \tag{10}$$

A formal representation of the circuit netlist results from the heterogeneous relation

$$C \subseteq P \times N. \tag{11}$$

Relation C describes the pin-to-net connections of all circuit elements. In graph notation, graph $G_C = (P \cup N, C)$ is a bipartite graph, i.e., there are only edges between vertices of set P and N. Since set P is the union of all sets P_μ referring to the pins of module m_μ, we can define relation C as the union of all relations C_μ which refer to the connections between the pins of single modules m_μ and the nets

$$C_\mu \subseteq P_\mu \times N, \qquad C = \bigcup_{\mu=1}^{|M|} C_\mu. \tag{12}$$

To check if two building blocks m_κ and m_λ form a new building block, the union of their circuit relations $C_{\kappa,\lambda}$ is built

$$C_{\kappa,\lambda} := C_\kappa \cup C_\lambda. \tag{13}$$

A corresponding relation $C_{l(1),l(2)}$ referring to the circuit relation of the two submodules of a library element l can

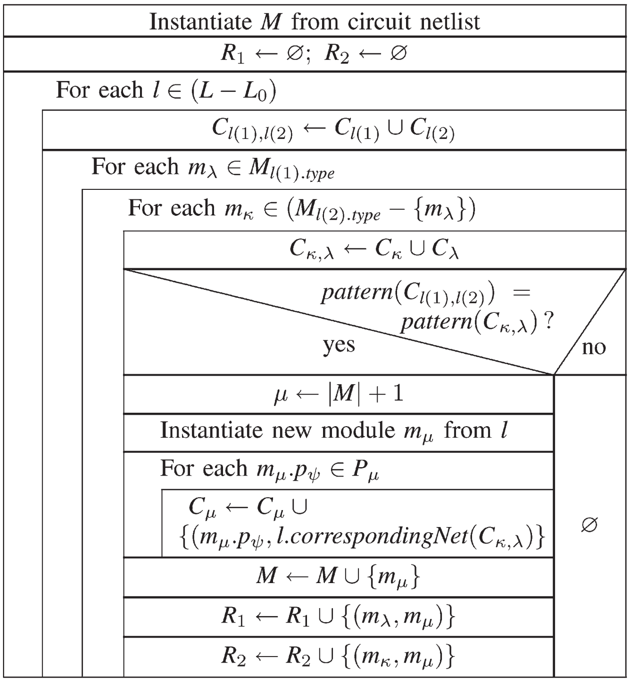

Fig. 5. Building block recognition algorithm.

be built the same way. If the patterns of $C_{\kappa,\lambda}$ and $C_{l(1),l(2)}$ match, m_κ and m_λ form a new building block m_μ that has the structure of library element l. For each library element, only appropriate candidates are checked. For instance, when NMOS CCMs are searched, only NMOS simple current mirrors and NMOS ls that were detected before are checked.

The recognition algorithm is shown in Fig. 5. At the beginning, set M is initialized with all circuit elements contained in the netlist, i.e., available modules from level 0. In addition, two relations R_1 and R_2 are initialized. Every time a new module m_μ is detected, a new ordered pair will be added to both relations. The upper or left submodule m_λ (according to Figs. 2 and 3), i.e., submodule with index 1 and the new module m_μ are stored as an ordered pair in R_1. The lower or right submodule m_κ (with index 2) and the new module m_μ are stored as an ordered pair in R_2. This information is needed to handle recognition ambiguities (Section IV). The outermost loop iterates bottom–up through all library elements $l \in L - L_0$, i.e., library elements that consist of at least two submodules. Thus, each library element is only examined once during the whole process. It does not matter in what order the library elements on each hierarchy level are examined, since the submodules of newly recognized modules will not be removed from M during the recognition process. Relation $C_{l(1),l(2)}$ is built for each library element. In the inner loops, all possible pairs of appropriate modules m_κ and m_λ are examined. If the pattern $C_{\kappa,\lambda}$ matches the pattern $C_{l(1),l(2)}$, the pair (m_κ, m_λ) forms a building block, and a new module m_μ with $\mu = |M| + 1$ is instantiated and added to set M. Consequently, the new module can be recognized as part of other building blocks in the next run of the outer loop. Its submodules are also kept in M, since a module can be contained in several building blocks in some cases. In addition, it cannot be determined beforehand if the detected building block is intended, and removing a submodule would rule out other possible building blocks containing that submodule.

Next, the pins of the new module are connected to the appropriate nets of its submodules. For each library element, a specific mapping which picks out the proper net from $C_{\kappa,\lambda}$ exists. For instance, in a simple current mirror, the common source pin will be connected to the common net of the source pins of both transistors. The function which returns this net is called *correspondingNet*. Finally, m_μ is added to set M, and the relations R_1 and R_2 are extended by one ordered pair each, consisting of the first or second submodules, respectively, as well as the new module.

Banks are treated differently. If a pair of equal building blocks (e.g., a pair of simple current mirrors) shares one or two common driving submodules (e.g., the driving transistor of a simple current mirror), these two building blocks belong to a bank. If one of those building blocks is already part of a bank, the other building block will be added to that bank. If not, a new bank will be instantiated from these two building blocks. All modules belonging to the same bank are stored in the same set. The recognition of banks is performed after the whole recognition process including the arbitration of assignment ambiguities.

The algorithm in Fig. 5 works for any hierarchy of building blocks organized like the one shown in Figs. 2 and 3. The hierarchical library representation allows a redundancy-free storage of sizing rules. Since each building block on level $i > 0$ consists of building blocks from lower hierarchy levels, only the additional rules for the current building block have to be stored in the libraries.

After the automatic bottom–up recognition of building blocks, the sizing rules will be assigned top–down. In Sections V and VI, sizing rules for basic building blocks in CMOS as well as in bipolar transistor technology are presented.

However, the number of detected building blocks in each of the presented circuits would be significantly higher without employing additional rules to resolve possible conflicts. The recognition algorithm only checks type and connectivity to determine if a pair of modules forms a building block. This could result in modules being recognized as part of different building blocks at the same time which is not correct in certain cases. The next task is to decide which building blocks to prefer in such cases. This will be explained in the following.

IV. Arbitration of Assignment Ambiguities

The algorithm in Fig. 5 describes how building blocks in a circuit are detected. All possible building blocks are detected purely from a structural analysis. In particular, on hierarchy level 1, where the structural type is always "transistor," it is possible that one transistor is part of several building blocks at the same time. In some cases, this is intended, e.g., a transistor can be part of a voltage reference I as well as of an ls. In a simple current mirror bank, the driving transistor is contained in several simple current mirrors that form the bank. However, in most cases, only one of the detected building blocks

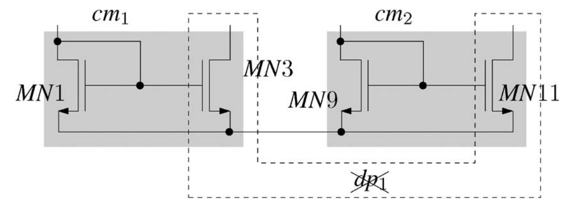

Fig. 6. Two simple current mirrors preferred over a dp.

fulfills the intended function, and the other one has to be discarded to avoid the assignment of not applicable sizing rules. In this section, a new heuristic methodology for the arbitration of such assignment ambiguities will be presented.

Fig. 8 shows a folded cascode circuit. All recognized building blocks above level 0 are shaded. An extract of it is shown in Fig. 6. Transistors $MN1$ and $MN3$, as well as $MN9$ and $MN11$, form simple current mirrors which are recognized by the recognition algorithm shown in Fig. 5. An additional dp consisting of $MN3$ and $MN11$ is detected as well since those two transistors are solely connected via their source pins. However, a transistor cannot be part of a simple current mirror and a dp at the same time. In fact, seven dps can be recognized in the folded cascode circuit, but only transistors $MN7$ and $MN8$ actually do fulfill the function of a dp.

A way to prevent false recognition results could be to check for the library elements in an order such that building blocks which usually appear more frequently than others are recognized first and to remove the submodules of each recognized building block from set M. However, a module could no longer be contained in more than one building block, and the detection of building blocks which consist of submodules from different hierarchy levels would become very unlikely. A buffered current mirror (BCM), for example, is modeled as a combination of a Darlington configuration II and a single transistor. The single transistor would probably be recognized as part of a building block on hierarchy level 1. Then, it would be removed from set M and could no longer be recognized as part of a BCM located on level 2.

We propose a different approach. First, all potential pairs are identified. In a second step, ambiguities are resolved. For this purpose, we define a dominance relation S which determines which building blocks are to be preferred over others if a module has been detected as part of more than one building block. Additionally, we define a rule given in (15) that—if it is obeyed—ensures that there are no ambiguities in the final recognition result. The homogeneous relation S and its elements $s \in S$ are defined as follows:

$$S \subseteq (YS \times \{1,2\})^2 \quad s := ((y,i),(z,j)). \tag{14}$$

The first component of each tuple of s, i.e., y or z, denotes the structural type of a library element. The second component denotes whether submodule 1 or 2 is referred to. If $((y,i),(z,j)) \in S$, this means that (y,i) is dominated by (z,j).

Relations R_1 and R_2 in Fig. 5 contain the so-called "recognition information." For each module m_μ that has been recognized as part of a building block m_λ, an ordered pair (m_μ, m_λ) is stored in R_1 if m_μ is submodule 1 of m_λ or in R_2 if m_μ is submodule 2 of m_λ.

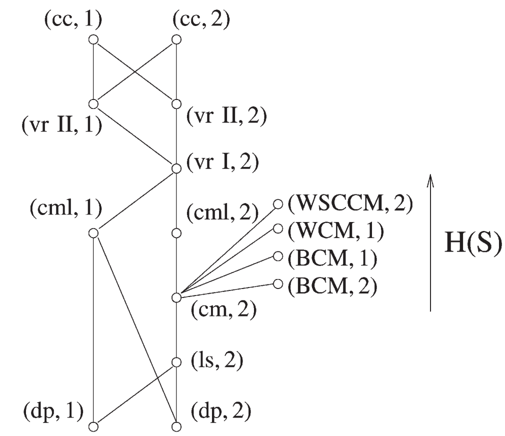

Fig. 7. Hasse diagram of dominance relation S.

With R_1, R_2, $R := R_1 \cup R_2$, and S, the dominance of certain building blocks over others can be formalized as follows:

$$\underset{m_\mu \in M}{\forall} \quad \underset{m_\kappa, m_\lambda \in \mathrm{des}(m_\mu)_\mathrm{R}}{\forall} \quad \underset{i,j \in \{1,2\}}{\forall} \Bigg[((m_\kappa.strtyp, i),(m_\lambda.strtyp, j)) \in S \wedge \underset{x \in \mathrm{des}^*(m_\mu)_\mathrm{R}}{\exists} (x, m_\lambda) \in R_j \longrightarrow \neg \underset{y \in \mathrm{des}^*(m_\mu)_\mathrm{R}}{\exists} (y, m_\kappa) \in R_i \Bigg]. \tag{15}$$

In (15), $\mathrm{des}(m_\mu)_\mathrm{R}$ denotes the set of descendants of m_μ in R, i.e., all modules m_μ is contained in, either as direct submodule or as part of another submodule. Additionally, $\mathrm{des}^*(m_\mu)_\mathrm{R} := \mathrm{des}(m_\mu)_\mathrm{R} \cup \{m_\mu\}$. The rule reads: For every module $m_\mu \in M$ and all modules m_κ and m_λ it is contained in, and all i, $j \in \{1,2\}$, it holds that if $(m_\kappa.strtyp, i)$ is dominated by $(m_\lambda.strtyp, j)$ and there is at least one module x being m_μ itself or a module m_μ is contained in that is submodule j of m_λ, there must not be a module y being m_μ itself or a module m_μ is contained in that is submodule i of m_κ. Thus, each m_κ that violates the rule and all modules it is contained in have to be removed from set M. The appendix includes an algorithm (12) that removes all such building blocks, so that the rule will eventually be observed. For instance, since $((\mathrm{dp},1),(\mathrm{cm},2)) \in S$ and transistor $MN3$ in Fig. 6 is both submodule 2 of simple current mirror cm_1 and submodule 1 of dp dp_1, this dp will be removed from the overall recognition result.

Relation S can be considered a set of rules to avoid unintended recognitions. We assume that the examined circuit is free of design faults. However, S could be extended so that it served as a rulebook for analog circuit design. Relation S is a strict order relation, i.e., it is asymmetric and transitive. A descriptive graphic representation of a strict order relation is a Hasse diagram which arises from the arrow diagram of the relation omitting all transitive edges. The Hasse diagram of S is shown in Fig. 7. The cascode pair is not included in

S since it does not produce any sizing rules on its own, and the main purpose of S is to avoid the assignment of wrong sizing rules.

After resolving conflicts applying the algorithm in Fig. 12, the function of some recognized building blocks might still be "uncertain." This is because the function of some building blocks is not clear if they are not recognized as part of a larger building block. This holds for the dp, Darlington configuration II and the single transistor, if it has not been recognized as part of any building block at all. For instance, the operating region of transistors that have not been recognized as part of a building block cannot be determined automatically. The cascode pair will not be classified as "uncertain" since it does not produce any sizing rules. An algorithm to collect all uncertain building blocks in a set is shown in the Appendix in Fig. 13. No sizing rules are assigned automatically to these uncertain building blocks. Instead, these building blocks are collected in a set and provided to the designer for further actions.

Experimental results in Section VII show the importance of the arbitration of assignment ambiguities to produce a meaningful recognition result.

V. Sizing Rules for CMOS Transistor Building Blocks

For each basic CMOS or bipolar building block according to Figs. 2 and 3, a set of sizing rules can be given. The types of these rules are shown in Fig. 1. In correspondence to Fig. 1, a rule will be labeled with FG or FE if it is a Geometric or Electrical constraint concerning Function, and with RG or RE if it is a Geometric or Electrical constraint concerning Robustness.

This section deals with sizing rules for CMOS transistor building blocks. Section VI treats bipolar transistor building blocks. All rules are presented for NMOS building blocks but can be formulated accordingly for their PMOS counterparts. Only building blocks that deliver sizing rules on their own are discussed. The constants that are introduced in the following are technology specific and have to be determined only once for each technology.

A CMOS transistor's behavior can be described using the Shichman–Hodges model [40]. The transistor's drain current is given by

$$i_d = \begin{cases} 0, & \text{if } v_{\mathrm{gs}} \le 0 \\ K\frac{W}{L}\left[(v_{\mathrm{gs}} - V_{\mathrm{th}}) - \frac{v_{\mathrm{ds}}}{2}\right] & \\ \quad \cdot v_{\mathrm{ds}}(1 + \frac{\lambda}{L} v_{\mathrm{ds}}), & \text{if } 0 \le v_{\mathrm{ds}} < v_{\mathrm{gs}} - V_{\mathrm{th}} \\ \frac{1}{2}K\frac{W}{L}(v_{\mathrm{gs}} - V_{\mathrm{th}})^2 & \\ \quad \cdot(1 + \frac{\lambda}{L} v_{ds}), & \text{if } v_{\mathrm{gs}} - V_{\mathrm{th}} \le v_{\mathrm{ds}}. \end{cases} \tag{16}$$

Here, W and L are the transistor's width and length, $K = \mu_{\mathrm{Si}} C_{\mathrm{ox}}$ with μ_{Si} being the electron mobility and C_{ox} the oxide capacity, V_{th} is the threshold voltage, and λ the channel length modulation coefficient. The gate current is very small and is therefore neglected. The design parameters are the transistor geometries W and L.

A. Building Blocks on Hierarchy Level 0

1) vccs: A transistor working as a voltage-controlled current source (vccs) has to operate in saturation. Furthermore, from [40]–[42], it follows that the drain-source current variation depends on variations of channel width and length, threshold voltage, electron mobility, and specific gate oxide capacitance with factors $1/w^2$, $1/l^2$, and $1/(w \cdot l)$. Additionally, $1/f$ noise is also proportional to $1/(w \cdot l)$. In [41], the variation of the drain current is given by

$$\frac{\sigma_{i_d}^2}{i_d^2} = \frac{A_K}{W \cdot L} + \frac{\sigma_W^2}{W^2} + \frac{\sigma_L^2}{L^2} + \frac{4}{(v_{\mathrm{gs}} - V_{\mathrm{th}})} + \frac{A_K}{W \cdot L}. \tag{17}$$

Hence, for robustness, certain minimum values for width, length, and area are required which are significantly larger than $L_{\min}$ and $W_{\min}$ from the basic technology.

Thus, the sizing rules for a vccs can be summarized as follows:

$$\text{FE1}: \quad v_{\mathrm{ds}} - (v_{\mathrm{gs}} - V_{\mathrm{th}}) \ge V_{\mathrm{SAT}_{\min}} \tag{18}$$

$$\text{FE2}: \quad v_{\mathrm{ds}} \ge 0 \tag{19}$$

$$\text{FE3}: \quad v_{\mathrm{gs}} - V_{\mathrm{th}} \ge 0 \tag{20}$$

$$\text{RG1}: \quad w \cdot l \ge A_{\min_{\mathrm{SAT}}} \tag{21}$$

$$\text{RG2}: \quad w \ge W_{\min_{\mathrm{SAT}}} \tag{22}$$

$$\text{RG3}: \quad l \ge L_{\min_{\mathrm{SAT}}}. \tag{23}$$

2) vcres: A transistor as a voltage-controlled resistor (vcres) operates in the linear region, hence

$$\text{FE1}: \quad (v_{\mathrm{gs}} - V_{\mathrm{th}}) - v_{\mathrm{ds}} \ge V_{\mathrm{lin}_{\min}} \tag{24}$$

$$\text{FE2}: \quad v_{\mathrm{ds}} \ge 0 \tag{25}$$

$$\text{FE3}: \quad v_{\mathrm{gs}} - V_{\mathrm{th}} \ge 0. \tag{26}$$

To ensure that a vcres operates in the deep ohmic region, $V_{\mathrm{lin}_{\min}}$ has to be sufficiently large.

B. Building Blocks on Hierarchy Level 1

1) Simple Current Mirror (cm): The function of a "cm" is to produce a constant ratio between the drain currents of the two transistors (gate currents are assumed to be zero)

$$\frac{i_{d_2}}{i_{d_1}} = \frac{(w_2/l_2)}{(w_1/l_1)}. \tag{27}$$

To keep the influence of the drain-source voltage low, both transistors are vccs. To avoid systematic mismatch due to channel length modulation, the drain-source voltage difference needs to be small, and the transistor lengths have to be equal. To avoid mismatch due to local process variations, the effective gate-source voltage has to be sufficiently large, hence

$$\text{FG}: \quad l_1 = l_2 \tag{28}$$

$$\text{FE}: \quad |v_{\mathrm{ds}_2} - v_{\mathrm{ds}_1}| \le \Delta V_{\mathrm{ds}_{\max(\mathrm{cm})}} \tag{29}$$

$$\text{RE}: \quad v_{\mathrm{gs}_{1,2}} - V_{\mathrm{th}_{1,2}} \ge V_{\mathrm{gs}_{\min}}. \tag{30}$$

2) ls: Basically, the level shifting function can be processed using a single transistor, e.g., a source follower or a transistor with its drain and gate terminal connected (diode connection). The function of the presented building block ls is to provide a constant differential voltage between—or equal voltages at—the two transistors' source pins. Both transistors are vccs. To avoid systematic mismatch, the lengths of the two transistors have to be equal. Additionally, RE of a cm holds.

$$\text{FG}: \quad l_1 = l_2 \tag{31}$$

$$\text{RE}: \quad v_{\text{gs}_{1,2}} - V_{\text{th}_{1,2}} \geq V_{\text{gs}_{\min}}. \tag{32}$$

3) dp: A dp transforms a gate-source voltage difference into a drain current difference. To reduce the influence of the drain-source voltage, both transistors work as vccs. The FE and FG of a cm hold analogously for the dp. For symmetry reasons, both transistors must have equal widths and lengths. To ensure that the drain-source current difference changes linearly with the gate-source voltage difference, the difference of the gate-source voltages may not exceed a certain value (RE).

$$\text{FG1}: \quad l_1 = l_2 \tag{33}$$

$$\text{FG2}: \quad w_1 = w_2 \tag{34}$$

$$\text{FE}: \quad |v_{\text{ds}_2} - v_{\text{ds}_1}| \leq \Delta V_{\text{ds}_{\max(\text{dp})}} \tag{35}$$

$$\text{RE}: \quad |v_{\text{gs}_2} - v_{\text{gs}_1}| \leq \Delta V_{\text{gs}_{\max}}. \tag{36}$$

4) cc: This building block can be found in voltage-controlled oscillators functioning as a negative resistor, for instance, but it can also form a simple memory consisting of two transistors. Symmetry reasons require

$$\text{FG1}: \quad l_1 = l_2 \tag{37}$$

$$\text{FG2}: \quad w_1 = w_2. \tag{38}$$

C. Building Blocks on Hierarchy Level 2

1) CCM: A CCM consists of a cm together with an ls to reduce the influence of the drain-source voltage of the driven transistor of the cm on the current ratio. It has a higher output impedance than a cm. The ls has to produce equal voltages at its source pins to obtain equal drain-source voltages at the transistors of the cm. Thus, the ls's transistors have to have the same widths as the current mirror's

$$\text{FG1}: \quad w_{\text{ls}(1)} = w_{\text{cm}(1)} \tag{39}$$

$$\text{FG2}: \quad w_{\text{ls}(2)} = w_{\text{cm}(2)}. \tag{40}$$

It is useful to use these absolute equalities instead of ratios between the transistor widths, because this reduces the number of degrees of freedom from four to two instead of from four to three. The fewer the design parameters, the faster the optimizer manages to find a solution where all specifications are met. The current ratio of a CCM depends on the ratio of the simple current mirror's transistor widths.

2) 4TCM: A four-transistor current mirror (4TCM) consists of a "vr I" and a "cml." It has the same advantage over a cm as the CCM and the additional advantage of a lower drain-source voltage drop, which is important for lower supply voltages. The two upper transistors are also recognized as ls; hence, the respective sizing rules have to be fulfilled by those two transistors. Additionally, the symmetry requirements of the CCM hold for the transistor widths. The two lower transistors operate as vcres, and the difference of their drain-source voltages needs to be small

$$\text{FG1}: \quad w_{\text{vrI}(1)} = w_{\text{vrI}(2)} \tag{41}$$

$$\text{FG2}: \quad w_{\text{cml}(1)} = w_{\text{cml}(2)} \tag{42}$$

$$\text{FE}: \quad |v_{\text{ds}_{\text{vrI}(2)}} - v_{\text{ds}_{\text{cml}(2)}}| \leq V_{\text{ds}_{\max(4\text{TCM})}}. \tag{43}$$

3) WSCCM: A wide swing CCM (WSCCM) consists of a "vr II" and a "cp." This type of current mirror is usually driven by a diode-connected CMOS transistor or a "vr I." The voltage drop along the two transistors on the left is just the v_{gs} of the lower left transistor, in contrast to the CCM where it is the sum of the two left transistors' v_{gs}'s. For the WSCCM, the sizing rules do not arise from its submodules. In contrast, the sizing rules for a CCM and its submodules can be taken over, i.e., the upper two transistors have to fulfill the sizing rules of an ls, the lower two transistors those of a cm.

4) WCM: A WCM consists of a cm and a single transistor and has also a higher output impedance than a cm. For the lower two transistors, the rules for a cm apply, but the roles of driving and driven transistors are reversed. Hence, the current ratio is given by

$$\frac{i_2}{i_1} = \frac{w_{\text{cm}(1)}}{w_{\text{cm}(2)}}. \tag{44}$$

In addition to the sizing rules for a cm for the two lower transistors, the third transistor has to operate as vccs.

5) IWCM: In the WCM, a mismatch between the simple current mirror's drain-source voltages occurs. To remedy this, a fourth transistor is added. An improved WCM (IWCM) consists of a cm together with an ls. Thus, the sizing rules are the same as for the CCM. The current ratio is determined by the cm. As for the WCM, the roles of driving and driven transistors are reversed in the cm.

D. Building Blocks on Hierarchy Level 3

1) DS: A DS consists of an arbitrary current mirror ("CM") and a dp. It does not produce any new sizing rules itself, but as soon as a dp is recognized as part of a DS, the sizing rules for a dp apply in any case, and the dp does not have to be classified as uncertain.

VI. Sizing Rules for Bipolar Transistor Building Blocks

In this section, sizing rules for bipolar transistor building blocks will be derived. All sizing rules will be presented for n-p-n transistors but hold analogously for p-n-p transistors. As for CMOS transistor building blocks, only building blocks that deliver sizing rules on their own will be discussed, and the values of all constants are technology specific.

In [43], the collector current of a bipolar transistor is given by

$$i_c = I_S e^{\frac{v_{\rm be}}{V_T}} \left(1 + \frac{v_{\rm ce}}{V_A}\right). \quad (45)$$

Here, $V_T = (k_{BT}/q_0)$ is the thermal voltage with k_B being the Boltzmann constant, T the room temperature, and q_0 the electron charge, and V_A is the early voltage. The saturation current is denoted by I_S. It depends on the transistor area A. The transistor area is the design parameter in bipolar transistor technology. Instead of scaling transistors, it is recommended to use identical transistors for each building block and to connect transistors in parallel, e.g., to produce a certain current ratio. The number of transistors connected in parallel will be denoted by N in the following.

In contrast to CMOS transistors whose gate current is usually neglected, the base current of a bipolar transistor has to be taken into account. The forward current gain β is given by

$$\beta = \frac{i_c}{i_b}. \quad (46)$$

The value of β depends on $v_{\rm be}$ [34]. For all presented building blocks, $\beta = \beta_{\max}$ has to be fulfilled. This is only the case if $v_{\rm be}$ stays in a range bounded by $V_{\rm be_{min}}$ and $V_{\rm be_{max}}$ where the slope of i_c is equal to that of i_b. This leads to the following additional sizing rule for all presented bipolar transistor building blocks:

$$\text{RE}: \qquad V_{\rm be_{min}} \leq v_{\rm be} \leq V_{\rm be_{max}}. \quad (47)$$

A. Building Blocks on Hierarchy Level 0

1) Transistor in Forward Active Region: All transistors contained in the library in Fig. 3 have to operate in the forward active region. Hence, $\nu_{\rm be}$ has to be larger than the cut-in voltage $V_{\rm ci}$ of the base-emitter diode, and the collector-emitter voltage has to be sufficiently larger than the difference between the base-emitter voltage and $V_{\rm ci}$.

$$\text{FE1}: \qquad \nu_{\rm be} - V_{\rm ci} \geq 0 \quad (48)$$

$$\text{FE2}: \qquad \nu_{\rm ce} - (\nu_{\rm be} - V_{\rm ci}) \geq V_{\rm ce_{SAT}}. \quad (49)$$

B. Building Blocks on Hierarchy Level 1

1) Simple Current Mirror (cm): A simple current mirror produces a constant ratio between the collector currents which depends on the ratio of the transistor areas which can be adjusted by the ratio between the number of transistors connected in parallel, which is denoted by N_2/N_1

$$\frac{i_{c_2}}{i_{c_1}} = \frac{A_2}{A_1} = \frac{N_2}{N_1}. \quad (50)$$

Due to base currents, the collector currents are smaller than the overall current flowing into the left branch of the current mirror. This error is proportional to $1/\beta$. To keep it low, β has to be maximal; hence, (47) has to be fulfilled. Both transistors have to operate in the forward active region. Since the collector current depends on the ratio between the collector-emitter voltage and the early voltage, the difference of the collector-emitter voltages has to be small

$$\text{FE}: \qquad |v_{\rm ce_2} - v_{\rm ce_1}| \leq \Delta V_{\rm ce_{max(cm)}}. \quad (51)$$

2) ls: This building block works as its CMOS counterpart (see Section V-B2). Both transistors have to operate in the forward active region, and the β of both transistors has to be maximal; thus, (47) has to be fulfilled.

3) dp: The function of a dp is to produce a difference in the collector currents i_{c_1} and i_{c_2} dependent on the base-emitter voltages of the two transistors. To keep the base currents small, both transistors have to operate in the forward active region, and their β has to be maximal; thus, (47) has to be fulfilled. For symmetry reasons, the number of transistors in parallel on both sides has to be the same, and the difference of the collector-emitter voltages has to be small to reduce the influence of the early effect.

$$\text{FG}: \qquad N_1 = N_2 \quad (52)$$

$$\text{FE}: \qquad |v_{\rm ce_2} - v_{\rm ce_1}| \leq \Delta V_{\rm ce_{max(dp)}}. \quad (53)$$

Similar to the CMOS dp, the difference of the base-emitter voltages has to be small as well.

$$\text{RE}: \qquad |v_{\rm be_2} - v_{\rm be_1}| \leq \Delta V_{\rm be_{max(dp)}}. \quad (54)$$

However, the region where Δi_c changes approximately linearly with $\Delta v_{\rm be}$ is much smaller than for the CMOS dp. To remedy this, resistors can be added at the transistors' emitter pins. The value of $V_{\rm be_{max(dp)}}$ in (54) depends on the value of these resistors and the current flowing into the common-emitter terminal.

4) Darlington Configuration I/II (dc I/II): These configurations are also called Darlington pairs. Both transistors have to operate in the forward active region, and for both, (47) has to be fulfilled to ensure proper function.

5) cc: This building block works as its CMOS counterpart. The number of transistors connected in parallel on both sides has to be the same

$$\text{FG}: \qquad N_1 = N_2. \quad (55)$$

C. Building Blocks on Hierarchy Level 2

1) BCM: A BCM is a modification of a cm to reduce the error in the current ratio caused by the base current. This error is only proportional to $1/\beta^2$ in a BCM. We used a "dc II" to model it as a pair of a two-transistor building block and a single transistor. For the two lower transistors, the sizing rules are the same as for a cm; for the third transistor, (47)–(49) have to be fulfilled.

D. Further Building Blocks on Hierarchy Levels 2 and 3

For the remaining building blocks, the same additional geometric sizing rules as for their CMOS counterparts apply. Note that "equal transistor widths" has to be replaced by "equal number of transistors connected in parallel."

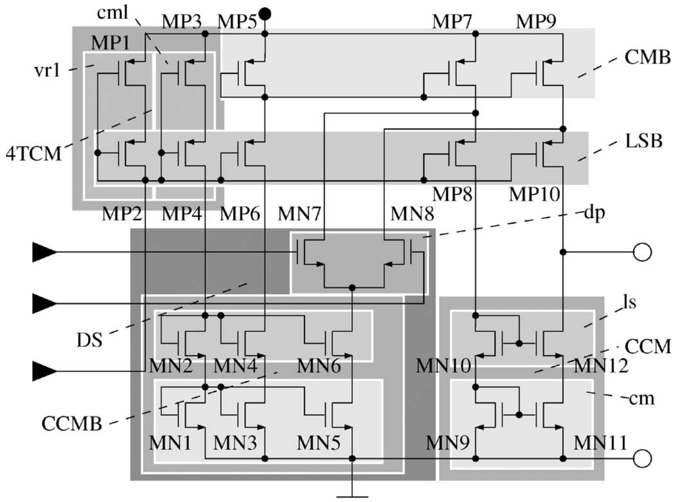

Fig. 8. Detected building blocks in a folded cascode OpAmp [37].

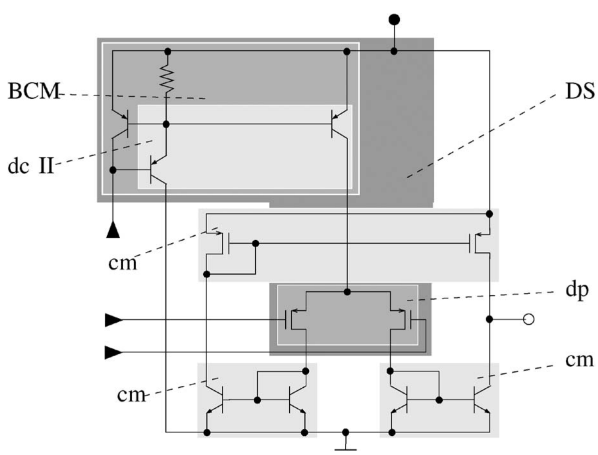

Fig. 9. Detected building blocks in a BiCMOS OpAmp [34].

VII. Results and Applications

A. Structure Recognition

In Figs. 8–10, three operational amplifiers are shown. The automatic building block recognition as shown in Fig. 5 leads to the detected building blocks shaded in Figs. 8–10.

Table II summarizes the number of detected building blocks on the different levels of hierarchy. It contains two further operational amplifiers which are not shown.

These results already present the final result of the subcircuit recognition method, including the arbitration of assignment ambiguities presented in Section IV. Table III shows the number of building blocks that were removed and would have been detected wrongly otherwise. The last column shows the number of uncertain building blocks identified in the circuits. The results clearly show the importance of the arbitration of assignment ambiguities.

B. Sizing Rules

Although the list of generic sizing rules for each building block is fairly small, the overall number of sizing rules for each circuit in the examined circuits is quite large. Table IV shows the total number of sizing rules that are established for each of the circuits. It can easily be seen that the automatic construction of these rules on circuit level, as presented here, is of large benefit even for small circuits.

Please note that the small number of generic sizing rules is a result of the presented hierarchical building block libraries on transistor-pair level. If the generic rules would be established on circuit level, the preparatory effort for analog synthesis would be significantly higher.

The inequality part of sizing rules has to be satisfied during the design process, e.g., sizing and design centering. The equality part of sizing rules leads to a reduction of the complexity of the design process, because it reduces the number of free design parameters. Both together enable a reliable design process.

C. Automatic Circuit Sizing and Design Centering

Automatic sizing means that circuit parameters are tuned in order to fulfill the performance specifications. The task of design centering is to tune circuit parameters in order to maximize the parametric yield (i.e., percentage of circuits satisfying specified performance bounds) with respect to manufacturing tolerances. For circuit sizing and design centering, we used WiCkeD [45]. Any other sizing tool could be used as well. Using the examples shown in Figs. 8–10, we illustrate that sizing rules are of large benefit for automatic circuit sizing and design centering.

Both automatic sizing and design centering were performed in two variants: In the first one, sizing rules were not considered; in the second one, they were considered. In the latter case, a feasibility optimization had to be performed first to find a design point where no sizing rules were violated. The initial design point was the same for both variants. Experimental results show that, with consideration of sizing rules, a higher yield was achieved at the end of the optimization process, mostly at a lower simulation cost.

The results for the three circuits are shown in Tables V–VII. At the initial design point, the specifications were not met, and several sizing rules were violated. It was always possible to find a design point where no sizing rules were violated.

In the circuit in Fig. 8, the number of design parameters was 14. At the beginning, 12 sizing rules were violated. When sizing rules were considered, the feasibility optimization needed 272 simulations, and automatic sizing required another 420 simulations. When no sizing rules were considered, no design point where the specifications were met was found at all. Thus, the consideration of sizing rules was crucial for this circuit. From the point that was found with consideration of sizing rules, design centering was performed, again once with consideration of sizing rules and once without. The results show that, from this point, design centering could be performed successfully in both cases, and a high yield was achieved. Table V summarizes the results for this circuit.

For the circuit in Fig. 9, the number of design parameters was eight, and there were two sizing rules violated at the beginning. The feasibility optimization required 27 simulations. For both variants, a design point where the specifications were met was

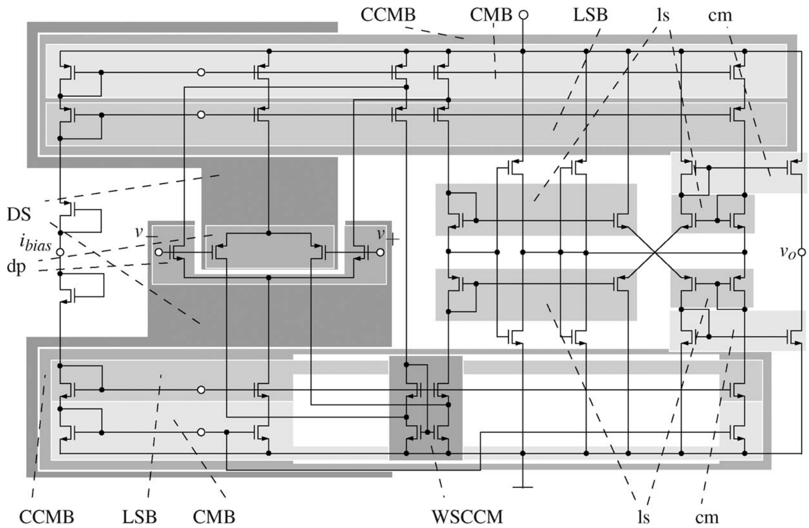

Fig. 10. Detected building blocks in a CMOS buffer amplifier [44].

TABLE II
NUMBER OF DETECTED BUILDING BLOCKS FOR SEVERAL OPERATIONAL AMPLIFIERS

Circuit	level of hierarchy 0	1	2	3	Total
Fig. 9	22	15	8	2	47
Fig. 10	11	5	1	1	18
Fig. 11	38	22	11	4	75
OP-27	19	11	3	2	35
MI buffer	31	12	5	2	50

TABLE III
RECOGNITION RESULTS INCLUDING REMOVED AND UNCERTAIN BUILDING BLOCKS

Circuit	#building blocks recognized	#building blocks removed	#"uncertain" building blocks
Fig. 9	47	7	0
Fig. 10	18	1	0
Fig. 11	75	34	6
OP-27	35	11	0
MI buffer	50	27	1

TABLE IV
NUMBER OF SIZING RULES FOR THE GIVEN CIRCUIT EXAMPLES

Circuit	#Equalities	#Inequalities	Total
Fig. 9	23	186	209
Fig. 10	6	61	67
Fig. 11	47	261	308
OP-27	25	83	108
MI buffer	21	164	185

TABLE V
RESULTS FOR THE CIRCUIT IN FIG. 8

	sizing rules considered? no	yes
#sims. for feasibility optimization	0	272
#sims. for automatic sizing	failed	420
#sims. for design centering	3428	3844
overall yield	99.46%	99.58%

TABLE VI
RESULTS FOR THE CIRCUIT IN FIG. 9

	sizing rules considered? no	yes
#sims. for feasibility optimization	0	27
#sims. for automatic sizing	154	145
#sims. for design centering after automatic sizing w/o sizing rules	failed	2475
overall yield	-	99.98%
#sims. for design centering after automatic sizing with sizing rules	failed	1365
overall yield	-	> 99.99%

TABLE VII
RESULTS FOR THE CIRCUIT IN FIG. 10

	sizing rules considered? no	yes	yes (+ 6 "vccs")
#sims. for feasibility optimization	0	27	31
#sims. for automatic sizing	failed	427	623
#sims. for design centering	1113	953	886
overall yield	45.2%	46.7%	71.1%

found. When no sizing rules were considered, seven sizing rules were violated after automatic sizing.

After automatic sizing, we compared the two design points that were found by automatic sizing considering the robustness of the circuit. For this purpose, a sweep over the supply voltage V_{dd} and the operating temperature ϑ was performed. Exemplarily, we present the sweep result for the power supply rejection ratio (PSRR) over the supply voltage which is shown in Fig. 11. In both cases, the specification of 80 dB was met at the nominal supply voltage. When V_{dd} dropped, the PSRR diminished—first smoothly, later drastically—to a value of less than 20 dB at V_{min}, when sizing rules were not considered. When sizing rules were considered, the PSRR was only slightly reduced and stayed above 80 dB throughout the specified operating range of V_{dd}. Hence, the design was much less sensitive to variations of operating conditions.

Next, design centering was performed from the two design points that were found by automatic sizing. Once again, this was done once with consideration of sizing rules and once without. When no sizing rules were considered for design centering,

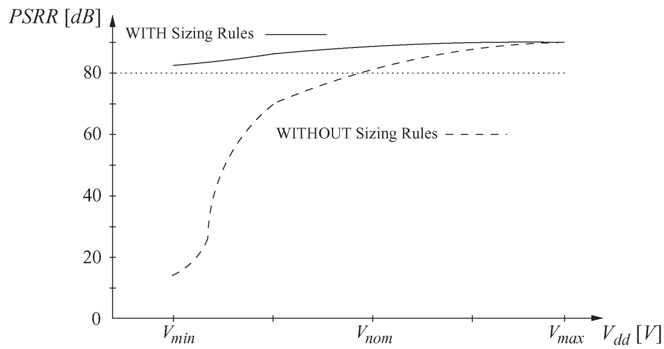

Fig. 11. PSRR over supply voltage for the circuit in Fig. 9.

the algorithm terminated prematurely with a linearization error. Hence, for the BiCMOS amplifier, the consideration of sizing rules was crucial for design centering.

When sizing rules were considered for design centering, a high yield was achieved for both design points that were found by automatic sizing. In the design centering run that started from the point that was found by automatic sizing with consideration of sizing rules, the yield was nearly 100%. In the run where sizing rules were not considered for automatic sizing, the yield was slightly lower but the simulation cost was about 80% higher. Table VI shows the results for this circuit.

The number of design parameters for the circuit in Fig. 10 was quite high, namely, 26. The specified performance bounds were also high, making it harder to fulfill the specifications. The number of sizing rules violated at the beginning was 21. Once again, automatic sizing without consideration of sizing rules failed, and the specifications could not be met. Only a low yield could be achieved for both variants, and the algorithm for design centering terminated early.

As shown in Table III, there are six uncertain building blocks in this circuit. These are six transistors which were not detected as part of a building block defined in our library, so that no operating region could be assigned automatically. Table VII summarizes the results for this circuit. The last column shows the result when we assigned the sizing rules for a vccs to all six transistors that were classified as uncertain. The resulting yield was significantly higher than in the other two cases. Hence, it was advantageous to consider additional sizing rules for these six transistors.

The results for all three circuits show that the consideration of sizing rules was crucial. The consideration of sizing rules lead to more robust designs mostly at lower simulation effort.

D. RSM

Response surface modeling (RSM) serves to replace performance evaluation by computationally expensive circuit simulation models with cheaper performance evaluation by analytical functions. In practical applications, RSM has to select basis functions for the analytical model, select test points where the "true" circuit is evaluated, and compute the coefficients of the analytical model. Another problem is the definition region of the analytical model.

Sizing rules provide an accurate and technically relevant definition region of an analytical model. The region where test points have to be simulated is much smaller than the original region defined by simple box constraints. In addition, the performance behavior is near to linear in the region where sizing rules are satisfied. This results in an increased accuracy of the analytical models [46].

E. Analog Placement

The building block recognition part of the sizing rules method is applied in order to automatically provide placement constraints for analog cells in industry. This enables analog layout designers to create the necessary placement specifications for analog cells starting just from the circuit schematic.

VIII. Conclusion

In this paper, an efficient method for capturing design knowledge on transistor-pair level has been developed. We introduced two generic hierarchical libraries of CMOS and bipolar transistor building blocks. A new algorithm to automatically detect building blocks and set up corresponding sizing rules in any circuit with an advanced arbitration of assignment ambiguities was developed.

The large number of sizing rules of a circuit clearly shows that these cannot be established manually for each circuit. The threshold values in the generic sizing rules have to be given just once for a technology.

The significance of the presented method follows from applications to circuit sizing, design centering, RSM, and analog placement. Sizing rules particularly make sure that automatic circuit sizing and design centering lead to technically meaningful and robust results in CMOS—as well as in BiCMOS—and bipolar transistor circuits.

IX. Future Work

In the future, the presented method will be extended to additional rules and library elements, as well as to other types of circuits like CMOS logic blocks.

Appendix
Algorithm for Arbitration of Assignment Ambiguities

Fig. 12 shows the algorithm to resolve conflicts after the process of structure recognition (see Section IV). First, set F is instantiated as a copy of M. Relation R is the union of R_1 and R_2. In the outer loop, all modules in F are checked. For the current module m_μ, set E of m_μ itself and all modules in F that m_μ is contained in—either as a submodule of building block $m_{\mu'}$ or on a lower hierarchy level—is generated. A module m_μ is contained in another module $m_{\mu'}$ if the expression $m_\mu R^+ m_{\mu'}$ is true, with relation R^+ being the transitive closure of R. Relation U contains all possible pairs of different modules in E without the identity relation I_E, i.e., no pairs that consist of the same element of E twice. In the following loops, each possible pair (u, v) with $u \neq v$ and both u and v contained in U and R_1 or R_2 are examined. The second element of u or v,

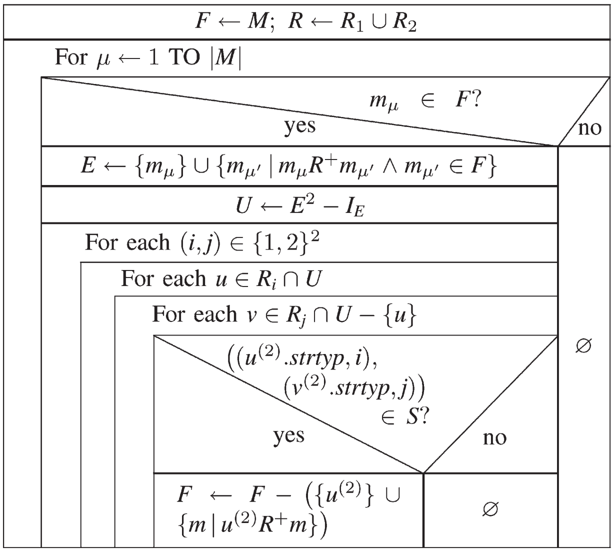

Fig. 12. Algorithm for arbitration of assignment ambiguities.

which is denoted by $u^{(2)}$ or $v^{(2)}$, denotes the module that $u^{(1)}$ or $v^{(1)}$, i.e., the first element of u or v, is contained in. To find out if m_μ is allowed to be contained in both $u^{(2)}$ and $v^{(2)}$, it is checked if the structural type of $v^{(2)}$ dominates the structural type of $u^{(2)}$ considering if their submodules $u^{(1)}$ and $v^{(1)}$ are submodule number 1 or 2, i.e., if u and v are in R_1 or R_2. This is done by checking if $((u^{(2)}.strtyp, i), (v^{(2)}.strtyp, j)) \in S$. If this is the case, module $u^{(2)}$ and all modules that $u^{(2)}$ is contained in are removed from F.

Example: Using the example in Fig. 6 from Section IV, it will be described how a conflict is resolved. After the recognition process, set M is given by

$$M = \{MN1, MN3, MN9, MN11, \mathrm{cm}_1, \mathrm{cm}_2, \mathrm{dp}_1\}.$$

Relations R_1 and R_2 look as follows:

$$R_1 = \{(MN1, \mathrm{cm}_1), (MN3, \mathrm{dp}_1), (MN9, \mathrm{cm}_2)\}$$

$$R_2 = \{(MN3, \mathrm{cm}_1), (MN11, \mathrm{cm}_2), (MN11, \mathrm{dp}_1)\}.$$

When $MN3$ is examined using the algorithm shown in Fig. 12, set E is instantiated as

$$E = \{MN3, \mathrm{cm}_1, \mathrm{dp}_1\}$$

since $MN3$ is both contained in cm_1 and dp_1. Thus, relation U is instantiated as

$$U = \{(MN3, \mathrm{cm}_1), (MN3, \mathrm{dp}_1), (\mathrm{cm}_1, \mathrm{dp}_1), (\mathrm{cm}_1, MN3), (\mathrm{dp}_1, MN3), (\mathrm{dp}_1, \mathrm{cm}_1)\}.$$

For $(i,j) = (1,2)$ or $(i,j) = (2,1)$ and $u = (MN3, \mathrm{dp}_1)$, $R_i \cap U$ and $R_j \cap U - \{u\}$ are both not empty; therefore, the inner if-clause will be executed. In the former case

$$R_1 \cap U = \{(MN3, \mathrm{dp}_1)\} \quad R_2 \cap U - \{u\} = \{(MN3, \mathrm{cm}_1)\}.$$

Thus, it is checked if $((\mathrm{dp}, 1), (\mathrm{cm}, 2)) \in S$. From the Hasse diagram in Fig. 7, it can be seen that this is true, i.e., if a module was both recognized as submodule 2 of a cm and submodule 1 of a dp, cm_1 is the dominating building block, and dp_1 will be removed from set F of all detected building blocks. In the latter case, it is checked if $((\mathrm{cm}, 2), (\mathrm{dp}, 1)) \in S$ which is false because the dp does not dominate the simple current mirror. When $MN11$ is examined later, there will be no further conflict to resolve, since dp_1 will no longer be contained in F.

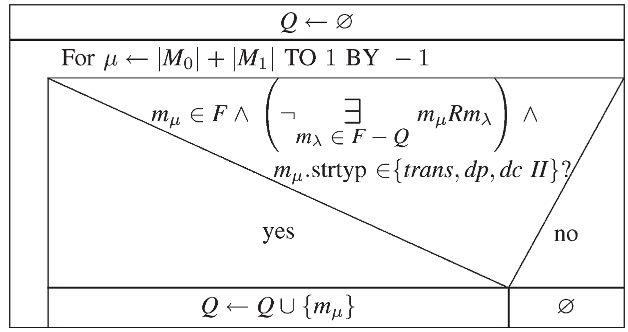

Fig. 13. Algorithm to determine uncertain building blocks.

Fig. 13 shows the algorithm to classify building blocks as uncertain. All these blocks are stored in a set Q that is empty at the beginning. The algorithm runs top–down through all detected building blocks on the lowest two hierarchy levels. It adds all dps, Darlington configurations II, and single transistors to Q that are not part of another building block that was not classified as uncertain before.

REFERENCES

[1] M. Degrauwe *et al.*, "IDAC: An interactive design tool for analog CMOS circuits," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 6, pp. 1106–1116, Dec. 1987.
[2] R. Harjani, R. Rutenbar, and L. Carley, "OASYS: A framework for analog circuit synthesis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 8, no. 12, pp. 1247–1266, Dec. 1989.
[3] H. Koh, C. Sequin, and P. Gray, "OPASYN: A compiler for CMOS operational amplifiers," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 9, no. 2, pp. 113–125, Feb. 1990.
[4] J. Harvey, M. Elmasry, and B. Leung, "STAIC: An interactive framework for synthesizing CMOS and BiCMOS analog circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 11, no. 11, pp. 1402–1417, Nov. 1992.
[5] R. Brayton, G. Hachtel, and A. Sangiovanni-Vincentelli, "A survey of optimization techniques for integrated-circuit design," *Proc. IEEE*, vol. 69, no. 10, pp. 1334–1363, Oct. 1981.
[6] M. Lightner, T. Trick, and R. Zug, "Circuit optimization and design," *Circuit Analysis, Simulation and Design*, vol. 3, pt. 2, A. Ruehli, Ed. Amsterdam, The Netherlands: North Holland, 1987, pp. 333–391.
[7] W. Nye, D. Riley, A. Sangiovanni-Vincentelli, and A. Tits, "DELIGHT. SPICE: An optimization-based system for the design of integrated circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 7, no. 4, pp. 501–519, Apr. 1988.
[8] S. Director and G. Hachtel, "The simplicial approximation approach to design centering," *IEEE Trans. Circuits Syst.*, vol. CAS-24, no. 7, pp. 363–372, Jul. 1977.
[9] K. Antreich and R. Koblitz, "Design centering by yield prediction," *IEEE Trans. Circuits Syst.*, vol. CAS-29, no. 2, pp. 88–95, Feb. 1982.
[10] R. Spence and R. S. Soin, *Tolerance Design of Electronic Circuits*. Reading, MA: Addison-Wesley, 1988.
[11] S. Director, W. Maly, and A. Strojwas, *VLSI Design for Manufacturing: Yield Enhancement*. Norwell, MA: Kluwer, 1990.
[12] P. Feldmann and S. Director, "Integrated circuit quality optimization using surface integrals," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 12, no. 12, pp. 1868–1879, Dec. 1993.
[13] J. C. Zhang and M. A. Styblinski, *Yield and Variability Optimization of Integrated Circuits*. Norwell, MA: Kluwer, 1995.

[14] M. del Mar Hershenson, S. P. Boyd, and T. H. Lee, "Optimal design of a CMOS Op-Amp via geometric programming," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 20, no. 1, pp. 1–21, Jan. 2001.

[15] G. Gielen, G. Debyser, K. Lampaert, F. Leyn, K. Swings, G. Van Der Plas, W. Sansen, D. Leenaerts, P. Veselinovic, and W. Van Bokhoven, "An analogue module generator for mixed analogue/digital ASIC design," *Int. J. Circuit Theory Appl.*, vol. 23, no. 4, pp. 269–283, 1995.

[16] G. Gielen, H. C. Walscharts, and W. C. Sansen, "Analog circuit design optimization based on symbolic simulation and simulated annealing," *IEEE J. Solid-State Circuits*, vol. 25, no. 3, pp. 707–713, Jun. 1990.

[17] E. S. Ochotta, R. A. Rutenbar, and L. R. Carley, "Synthesis of high-performance analog circuits in ASTRX/OBLX," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 15, no. 3, pp. 273–294, Mar. 1996.

[18] E. Ochotta, T. Mukherjee, R. A. Rutenbar, and L. R. Carley, *Practical Synthesis of High-Performance Analog Circuits*. Norwell, MA: Kluwer, 1998.

[19] K. Antreich, H. Graeb, and C. Wieser, "Circuit analysis and optimization driven by worst-case distances," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 13, no. 1, pp. 57–71, Jan. 1994.

[20] A. Dharchoudhury and S. M. Kang, "Worst-case analysis and optimization of VLSI circuit performances," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 14, no. 4, pp. 481–492, Apr. 1995.

[21] D. M. Binkley, C. E. Hopper, S. D. Tucker, B. C. Moss, J. M. Rochelle, and D. P. Foty, "A CAD methodology for optimizing transistor current and sizing in analog CMOS design," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 22, no. 2, pp. 225–237, Feb. 2003.

[22] M. del Mar Hershenson, S. P. Boyd, and T. H. Lee, "GPCAD: A tool for CMOS Op-Amp synthesis," in *Proc. IEEE/ACM ICCAD*, 1998, pp. 296–303.

[23] G. Debyser and G. Gielen, "Efficient analog circuit synthesis with simultaneous yield and robustness optimization," in *Proc. IEEE/ACM ICCAD*, 1998, pp. 308–311.

[24] T. Mukherjee, L. R. Carley, and R. A. Rutenbar, "Efficient handling of operating range and manufacturing line variations in analog cell synthesis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 19, no. 8, pp. 825–839, Aug. 2000.

[25] R. Phelps, M. Krasnicki, R. A. Rutenbar, L. R. Carley, and J. R. Hellums, "Anaconda: Simulation-based synthesis of analog circuits via stochastic pattern search," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 19, no. 6, pp. 703–717, Jun. 2000.

[26] K. Antreich, J. Eckmueller, H. Graeb, M. Pronath, F. Schenkel, R. Schwencker, and S. Zizala, "WiCkeD: Analog circuit synthesis incorporating mismatch," in *Proc. IEEE CICC*, May 2000, pp. 511–514.

[27] N. Rubanov, "SubIslands: The probabilistic match assignment algorithm for subcircuit recognition," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 22, no. 1, pp. 26–38, Jan. 2003.

[28] P. Mandal and V. Visvanathan, "CMOS Op-Amp sizing using a geometric programming formulation," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 20, no. 1, pp. 22–38, Jan. 2001.

[29] N. Dhanwada, A. Nunez-Aldana, and R. Vemuri, "Hierarchical constraint transformation using directed interval search for analog system synthesis," in *Proc. DATE*, 1999, pp. 328–335.

[30] T. Sripramong and C. Toumazou, "The invention of CMOS amplifiers using genetic programming and current-flow analysis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 21, no. 11, pp. 1237–1252, Nov. 2002.

[31] H. Graeb, S. Zizala, J. Eckmueller, and K. Antreich, "The sizing rules method for analog integrated circuit design," in *Proc. IEEE/ACM ICCAD*, 2001, pp. 343–349.

[32] T. Massier, H. Graeb, and U. Schlichtmann, "Sizing rules for bipolar analog circuit design," in *Proc. DATE*, 2008, pp. 140–145.

[33] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*. Hoboken, NJ: Wiley, 1997.

[34] K. R. Laker and W. Sansen, *Design of Analog Integrated Circuits and Systems*. New York: McGraw-Hill, 1994.

[35] M. Ismail and T. Fiez, "Analog VLSI Signal and Information Processing," in *Electrical and Computer Engineering*. New York: McGraw-Hill, 1994.

[36] P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 3rd ed. Hoboken, NJ: Wiley, 1993.

[37] P. E. Allen and D. R. Holberg, *CMOS Analog Circuit Design*. London, U.K.: Oxford Univ. Press, 1987.

[38] H. F. Mattson, *Discrete Mathematics with Applications*. New York: Wiley, 1993.

[39] Object Management Group, *OMG Unified Modeling Language Specification*, 1999, Object Management Group, Inc.

[40] H. Shichman and D. Hodges, "Modeling and simulation of insulated-gate field-effect transistor switching circuits," *IEEE J. Solid-State Circuits*, vol. SSC-3, no. 3, pp. 285–289, Sep. 1968.

[41] K. Lakshmikumar, R. Hadaway, and M. Copeland, "Characterization and modeling of mismatch in MOS transistors for precision analog design," *IEEE J. Solid-State Circuits*, vol. SSC-21, no. 6, pp. 1057–1066, Dec. 1986.

[42] M. Pelgrom, A. Duinmaijer, and A. Welbers, "Matching properties of MOS transistors," *IEEE J. Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1440, Oct. 1989.

[43] R. C. Jaeger, *Microelectronic Circuit Design*. New York: McGraw-Hill, 2004.

[44] J. Fisher and R. Koch, "A highly linear CMOS buffer amplifier," *IEEE J. Solid-State Circuits*, vol. SSC-22, no. 3, pp. 330–334, Jun. 1987.

[45] *MunEDA*, 2007. [Online]. Available: http://www.muneda.com

[46] S. Zizala, J. Eckmueller, and H. Graeb, "Fast calculation of analog circuits' feasibility regions by low level functional measures," in *Proc. IEEE ICECS*, 1998, pp. 85–88.

Tobias Massier received the Dipl.Ing. degree in electrical engineering and information technology from the Technische Universität München, Munich, Germany, in 2002, where he has been working toward the Dr. Ing. degree in the Department of Electrical Engineering and Information Technology, Institute for Electronic Design Automation, since 2002.

His research interests include computer-aided design of analog integrated circuits and systems, particularly structural analysis and sizing of analog circuits.

Helmut Graeb (M'02–SM'03) received the Dipl.Ing. and Dr. Ing. degrees from the Technische Universität München (TUM), Munich, Germany, in 1986 and 1993, respectively.

From 1986 to 1987, he was with Siemens Corporation, Munich, where he was involved in the design of DRAMs. Since 1987, he has been with the Institute of Electronic Design Automation, TUM, where he has been the Head of a research group since 1992. He has published more than 90 papers, six of which were nominated for best papers. His research interests are design automation for analog and mixed-signal circuits.

Dr. Graeb has, for instance, served as a member or Chair of the Analog Program Subcommittees of the International Conference on Computer-Aided Design, Design Automation Conference, and Design Automation and Test in Europe conferences, as the Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS" PART II: ANALOG AND DIGITAL SIGNAL PROCESSING and IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, and as a member of the Technical Advisory Board of MunEDA GmbH, Munich. He was the recipient of the 2004 Best Teaching Award of the TUM Electrical Engineering Faculty Students Association and the third prize of the 1996 Munich Business Plan Contest.

Ulf Schlichtmann (S'88–M'90) received the Dipl.Ing. and Dr. Ing. degrees in electrical engineering and information technology from the Technische Universität München (TUM), Munich, Germany, in 1990 and 1995, respectively.

From 1994 to 2003, he was with the Semiconductor Group, Siemens AG which, in 1999, became Infineon Technologies AG, where he held various technical and management positions in design automation, design libraries, IP reuse, and product development. Since 2003, he has been with TUM as a Professor and the Head of the Institute for Electronic Design Automation. His research interests are in computer-aided design of electronic circuits and systems, with special emphasis on designing robust systems.