FUBOCO: Structure Synthesis of Basic Op-Amps by FUnctional BlOck COmposition

INGA ABEL and HELMUT GRAEB, Technical University of Munich, Chair of Electronic Design Automation, Germany

This paper presents a method to automatically synthesize the structure of an operational amplifier. It is positioned between approaches with fixed design plans and a small search space of structures and approaches with generic structural production rules and a large search space with technically impractical structures. The presented approach develops a hierarchical composition graph based on functional blocks that spans a search space of thousands of technically meaningful structure variants for single-output, fully-differential and complementary operational amplifiers. The search algorithm is a combined heuristic and enumerative process. The evaluation is based on circuit sizing with a library of behavioral equations of functional blocks. Formalizing the knowledge of functional blocks in op-amps for structural synthesis and sizing inherently reduces the search space and lessens the number of created topologies not fulfilling the specifications. Experimental results for the three op-amp classes are presented. An outlook how this method can be extended to multi-stage op-amps is given.

Additional Key Words and Phrases: Analog circuit design, CMOS, operational amplifiers,

ACM Reference Format:
Inga Abel and Helmut Graeb. 2024. FUBOCO: Structure Synthesis of Basic Op-Amps by FUnctional BlOck COmposition. 1, 1 (October 2024), 24 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Automation of structural synthesis of analog circuits has not gained much attention from industry. A reason for this might be that the current approaches are following two major paths, none of which seems to be attractive from the industrial point of view. One path of structural synthesis is a specification-based selection of one or few netlists from a library [10, 12, 14–16, 19]. This is an automation of what happens in practice, as every company has a set of, say, 20 to 30 different netlists for op-amps, from which is initially chosen according to the specification. As an experienced designer can do this selection instantly, there is little gain in design time or design quality. The other path of structural synthesis builds up the netlist by combining modules of transistors and transistor groups while satisfying Kirchhoff laws and basic voltage/current conversion at the module interfaces [6, 7, 9, 11, 20, 21, 32]. Here, a plethora of variants is created, from which the promising ones are selected only after symbolic analysis and complete sizing. Designers see an unnecessary variety of intermediate solutions that they would never have created. In the presented approach, a new path for analog structural synthesis is started, which lies in the middle between these two poles. It does more than selecting from 20 to 30 available alternatives. And it creates a much smaller number of intermediate solutions that fail the requirements.

Authors' address: Inga Abel, inga.abel@tum.de; Helmut Graeb, helmut.graeb@tum.de, Technical University of Munich, Chair of Electronic Design Automation, Arcisstr. 21, Munich, Germany.

Manuscript submitted to ACM

Early topology synthesis approaches were [11, 15, 19]. While [11] was tested on different capacitor structures, as LC-structures or switch-capacitors, [15, 19] were developed for operational amplifiers. In [15], a top level abstraction of a circuit is implemented and hierarchically transformed into a transistor level structure with a fixed design plan for evaluation. [19] has a library of 64 topologies and fixed design plans. Synthesis approaches based on genetic algorithms followed to overcome the topology dependence of the early approaches [6, 9, 14, 20]. They are based on libraries containing simple transistor structures. With these methods, many different circuit topologies are created. However, many topologies are redundant. To size the topologies and evaluate their practicability, simulations are used. To lessen the number of redundant op-amp topologies, graph-grammar based approaches were developed [7, 21, 32]. With strict grammar rules and isomorphism techniques, the number of redundant topologies are reduced. The evaluation and sizing of the topologies takes place after synthesizing all possible topologies. Thus, much computation time is spent for circuits which cannot fulfill the given specifications from a visual inspection of a designer. Other approaches to lessen the number of created topologies were rule-based topology synthesis algorithms which closely implemented designer knowledge [10, 12, 16].

Different to previous methods, the presented synthesis approach contains a comprehensive computer-oriented systematic of op-amp functional building blocks like, e.g., amplification stages with their internal transconductances, loads and biases. The functional blocks form a composition graph. Thus, it overcomes the topology dependency in [12, 15, 19]. But it is still close enough to the op-amp structure such that redundant or impracticable topologies are avoided. This makes algorithms to remove these structures as in [21, 32] unnecessary. A novel algorithm for structural synthesis iteratively composes op-amp netlists from these basic functional blocks. For given specifications, the netlists are evaluated using an analytical equation-based sizing method similar to [3]. A library was developed storing analytical behavior models for every functional block in an op-amp [4]. Useless configurations are avoided in an early phase of the structural synthesis process. For instance, there exist performance features which degrade if a second stage is added to a one-stage op-amp topology, e.g., area, phase margin. In this case, the corresponding two-stage op-amp topology is not created if the one-stage op-amp already indicates to fail these specifications. Introducing design equations into the structural synthesis process obviates the need for symbolic analysis [11] or circuit simulation [20, 31] in evaluating intermediate solutions. This is achieved by a rigorous adoption of the practical creation process of analog designers along the functional block hierarchy of an operational amplifier.

This paper is an extended version of [1]. [1] presented a prototype of the synthesis tool, creating up to 34 op-amp topologies on amplification stages and evaluating the topologies through sizing. In this paper, we present the functional block composition of several thousand op-amp variants, building up the amplification stages by their subblocks, starting on transistor level with basic one-transistor functional blocks. New compared to [1], it presents:

- An algorithm to synthesize every functional block of an op-amp by its functional subblocks starting on transistor level up to whole op-amp topology (Sec. 3).
- An algorithm to synthesize a functional block based bias circuit for an op-amp topology (Sec. 4).
- An structural synthesis algorithm for op-amps featuring thousands of variations (Sec. 5 and Sec. 6).
- Experimental results (Sec. 7) featuring three op-amp types: single-output, fully-differential, complementary.
- An outlook how this method can be extended to three-stage op-amp topologies (Sec. 8).

With the enhanced version of the synthesis method, up to 3912 op-amps can now be synthesized. The number of created op-amps depends on the type of specification. Broad specifications result in many synthesized topology alternatives, while strict specifications lead to a small set of synthesized netlists.

HL 5	Op-amps	single-output (SO), fully-differential (FD), complementary (Comp)
HL 4	Op-amp subblocks	op-amp bias (b_O), amplification stage (a)
HL 3	Amplification stage subblock	load (l), transconductance (tc), stage bias (b_s)
HL 2	Structures	voltage bias (vb), current bias (cb), differential pair (dp),
HL 1	Devices	normal transistor (nt), diode transistor (dt), capacitor (cap)

Fig. 1. Hierarchical structure of functional blocks in op-amps

The goal of this approach is to perform op-amp synthesis using a formalized, computer-oriented description of the fundamentals of op-amp design. The hierarchical structure of the approach allows a straightforward extension to further functional blocks and multi-stage op-amp architectures. This allows not only to create suitable op-amps for given specifications sets, but also to create large circuit libraries containing only practicable solutions. Hence, the created libraries provide large data sets for future machine learning projects.

In the following, an overview of the functional blocks in op-amps is given (Sec. 2). Sec. 3 describes the generic algorithm to synthesis each functional block in Sec. 2 based on its subblocks. Sec. 4 describes the synthesis of the op-amp bias circuit. Sec. 5 describes the creation of whole op-amp topologies based on the previous described algorithms. The whole FUBOCO synthesis process is presented in Sec. 6. Experimental results featuring three op-amp types are presented in Sec. 7. Sec. 8 discusses the extension of the method to three-stage op-amps. The paper ends with a conclusion (Sec. 9).

2 FUNCTIONAL BLOCKS IN OP-AMPS

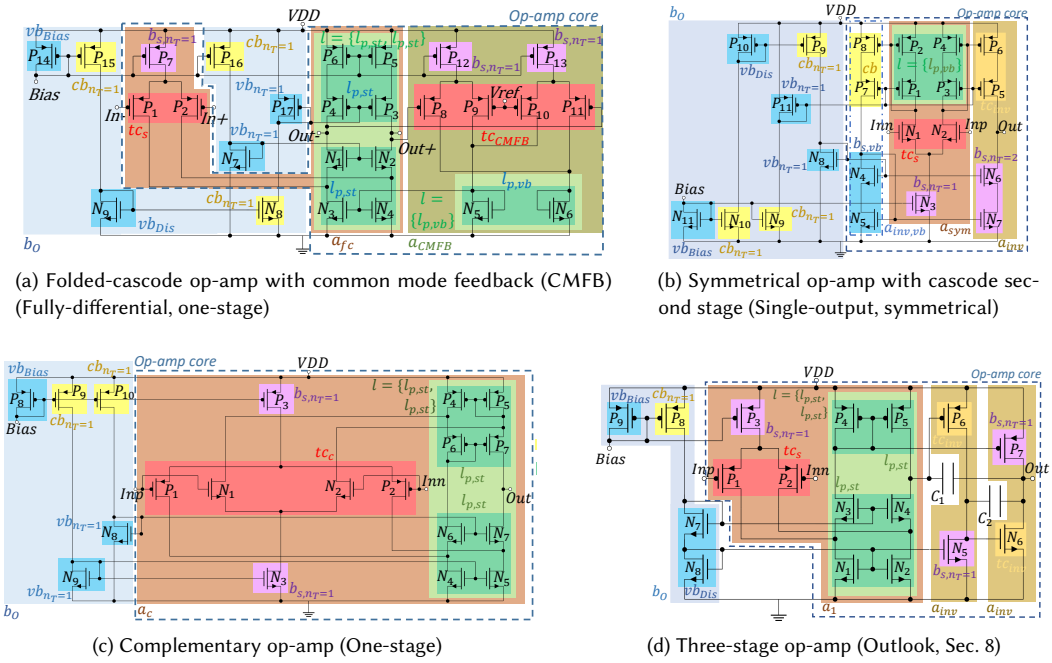
Every op-amp consists of a set of transistor blocks which can be characterized by their function and are therefore called functional blocks in the following. This paper presents a hierarchical structuring of these functional blocks and a computer-oriented representation of the fundamental principles of op-amp design composition as described in standard works [8, 13, 18, 26, 27, 29]. An overview of the basic functional blocks structured on five hierarchical levels is given in Fig. 1. Examples are shown using four different op-amp topologies in Fig. 2. A more detailed description is given in [2].

Hierarchy level 1 (HL 1) consists of devices, e.g., *capacitors cap* and transistors. Transistors are further divided into groups according to their self-connection. *Normal transistors* (nt) are transistors without any self-connection, e.g., Fig. 2a, P_1. *Diode transistors* (dt) are transistors with their gates connected to their drains, e.g., Fig. 2a P_{14}.

Hierarchy level 2 (HL 2) consists of transistor structures, e.g., *voltage bias* (vb, Fig. 2a vb_{Bias}, vb_{Dis}), *current bias* (cb, Fig. 2a $cb_{nT=1}$) and *differential pair* (dp, Fig. 2b N_1, N_2). The voltage and current biases can be simple, e.g. Fig. 2a $vb_{nT=1}$, or cascode (Fig. 2d vb_{Dis}).

Hierarchy level 3 (HL 3) consists of the amplification stage subblocks which are the *transconductance* (tc, Fig. 2, red), the *in-stage load* (l, Fig. 2, light green) and the *stage bias* (b_s, Fig. 2, violet). In the following, we abbreviate the in-stage load to load. For the transconductance, two main types exist: non-inverting (tc_{ninv}, Fig. 2b tc_s) and inverting (tc_{inv}, Fig. 2b). The non-inverting transconductance can be further divided into three types: simple (tc_s, Fig. 2b), complementary (tc_c, Fig. 2c), and common-mode feedback (CMFB) (tc_{CMFB}, Fig. 2a). The load consists of one or two load parts (l_p, Fig. 2, dark green).

Hierarchy level 4 (HL 4) consists of op-amp subblocks, which are the *amplification stages* (a), and the *op-amp bias* (b_O). Two types of amplification stages exist: non-inverting (a_{ninv}, Fig. 2a a_{fc}), and inverting (a_{inv}, Fig. 2b). The non-inverting amplification stages are further divided into simple first stage (a_s), complementary first stage (a_c, Fig. 2c), telescopic



(a) Folded-cascode op-amp with common mode feedback (CMFB) (Fully-differential, one-stage)

(b) Symmetrical op-amp with cascode second stage (Single-output, symmetrical)

(c) Complementary op-amp (One-stage)

(d) Three-stage op-amp (Outlook, Sec. 8)

Fig. 2. Example topologies synthesizable with the presented method, colored background: functional blocks of HL 2 - 4

first stage (a_{tel}), folded-cascode first stage (a_{fc}, Fig. 2a), symmetrical first stage (a_{sym}, Fig. 2b) and common-mode feedback stage (a_{CMFB}, Fig. 2a).

Hierarchy level 5 (HL 5) consists of the op-amp itself. Three types of op-amps are subject of this paper: single-output (SO), fully-differential (FD) and complementary (Comp). Complementary op-amps are defined as op-amps having a pmos and nmos differential pair forming the input stage.

In the following, we will distinguish between the *core* of an op-amp, containing the amplification stages and the capacitors, and the *op-amp bias* b_O, containing all transistors needed to bias the structures in the op-amp core (Fig. 2).

3 SYNTHESIS OF FUNCTIONAL BLOCKS EXCEPT OP-AMP BIAS

The hierarchical structure of functional blocks allows the synthesis of structural implementations of a functional bock based on its subblocks. This section presents a new generic algorithm which covers all blocks in Fig. 1 except the op-amp bias b_O. A separate algorithm to synthesize the op-amp bias is presented in Sec. 4.

3.1 Data Structure

A generic approach to synthesis of op-amp functional blocks requires the transition from a functional, i.e., behavioral, description of a block, to a transistor implementation, i.e., to a structural description, of a block. In this transition, different and new pins may arise. In this work, this transistion is implemented by representing functional blocks as instances with specific sets of pins. The set of pins varies for different functional block (implementation) types. Each functional block type has its own specific set of pins. The functional block *voltage bias* for instance has two implementation

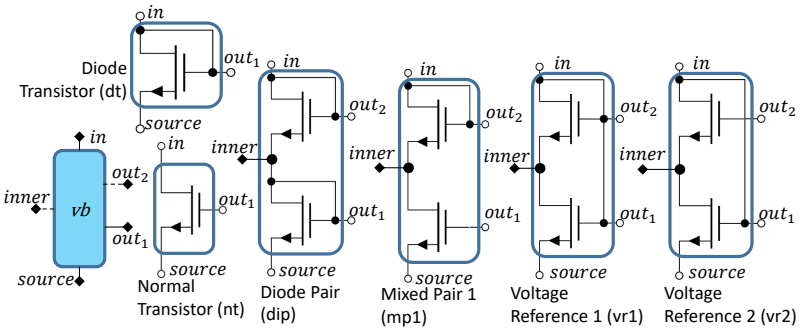


Fig. 3. Voltage bias instance and corresponding structural implementations

$S_1, \ldots, S_i$:	Structural implementation sets of $FB_1, \ldots, FB_i$ ($2, \ldots, i$ optional)
R_c (optional):	Characteristic connections of FB_{new}
$\mathcal{R}_f$ (optional):	Rules s_{new} must fulfill
R_a (optional):	Additional connections s_{new} can have
$P_{FB_{new}}$:	Pin set of s_{new}

Fig. 4. Input of Algorithm 1, FB_j: jth functional block, s_j: a structural implementation of FB_j

types, i.e., simple and cascode (Fig. 3). If the voltage bias is simple, its instance has three pins: $in, out_1, source$. If the voltage bias is cascode, its instance has two additional pins: $inner, out_2$. The pins define the generic pin sets for the two implementation types. They cover all possible implementation sets, even if an implementation as for instance *vr1* happens to connect two pins. This data structure provides exchangeability and flexibility in the synthesis process.

3.2 Generic Algorithm to Synthesize a Functional Block (Except Op-Amp Bias)

Alg. 1 creates for every functional block FB_{new} in Fig. 1 (but the op-amp bias) a set of structural implementations (instances) S_{new}.

3.2.1 Input. The input of the algorithm is defined in Fig. 4 and comprises the following sets.

Structural implementations $S_1, S_2, .., S_i$ of the functional subblocks of FB_{new}: i is the number of functional subblocks that are combined to build FB_{new}. E.g., a cascode voltage bias in Fig. 3 consists of two functional blocks, hence there are two sets S_1, S_2. Each set consists of normal and diode transistors $nt \in NT, dt \in DT$ having the same doping Φ ($S_1 : NT_\Phi, DT_\Phi; S_2 : NT_\Phi, DT_\Phi;$). Further input defines which combinations of these are allowed in a cascode voltage bias.

Characteristic connections R_c state how the instances in $S_1, .., S_i$ are connected to each other. This input is optional as no characteristic connections are provided for functional blocks consisting of one instance, e.g., simple voltage bias. To create a cascode voltage bias, R_c contains that the drain of a transistor $s_1 \in S_1$ must be connected to the source of a transistor $s_2 \in S_2$ ($R_c : s_1.drain \leftrightarrow s_2.source$).

Functional block rules $\mathcal{R}_f$ define wanted and unwanted connections independent of the characteristic connections. This is used to verify that a combination of different instances of functional subblocks is an implementation of FB_{new}. E.g., $\mathcal{R}_f$ of a cascode voltage bias (Fig. 3) contains that the pin *in*, i.e., $s_2.drain$, must be connected to one of the gates of its subblocks ($\mathcal{R}_f : s_2.drain \leftrightarrow (s_1.gate \vee s_2.gate)$).

Algorithm 1 Synthesis of a functional block except op-amp bias

Require: Compare Fig. 4

```
1:  S_new := { } // The set of structural implementations of FB_new is empty
2:  for all s_1 ∈ S_1 do
3:    for all s_2 ∈ S_2 do
4:      ....
5:      for all s_i ∈ S_i do
6:        c_new := createConnections(s_1, s_2, ..., s_i, R_c)
7:        if fulfillesRules(R_f,c_new) then
8:          s_new := createNewInstance(c_new, P_fb_new)
9:          S_new := S_new ∪ {s_new}
10:       end if
11:       for all r_a ∈ R_a do
12:         c_new := createConnections(c_new, r_a)
13:         if fulfillesRules(R_f,c_new) then
14:           s_new := createNewInstance(c_new, P_fb_new)
15:           S_new := S_new ∪ {s_new}
16:         end if
17:       end for
18:       ...
19:     end for
20:   end for
21: end for
22: return S_new
```

$\mathcal{R}_f$ also contains *basic structural rules* of analog building blocks as, e.g., that no transistor drain $t_m.drain$ is allowed to be connected to another transistor drain $t_n.drain$ of the same doping Φ:

$$\forall_{t_m,t_n \in T_\Phi} t_m.drain \nleftrightarrow t_n.drain \tag{1}$$

T_Φ are all transistors in the newly created implementation s_{new} of FB_{new} with doping type Φ. If the diode pair (dip) in Fig. 3 would have a connection between out_1 and out_2 it would not be a valid structural implementation of a voltage bias.

Additional connections R_a formulate additional optional connections to the connections in R_c. In the $vr1$-implementation of a cascode voltage bias (Fig. 3), the gates of the transistor are additionally connected. In the voltage reference 2, the gate of the lower transistor is additionally connected to the drain of the transistor above ($R_a : \{s_1.gate \leftrightarrow s_2.gate\}, \{s_2.drain \leftrightarrow s_1.gate\}$).

3.2.2 Algorithm. To synthesize all structural implementations of a functional block, the algorithm iterates over the sets of structural implementations $S_1, S_2, .., S_i$ to create all possible combinations. For each combination, a subcircuit c_{new} is created, consisting of $s_1 \in S_1$, $s_2 \in S_2$, ..., $s_i \in S_i$ having the required connections in R_c. It is checked if c_{new} fulfills the rules in $\mathcal{R}_f$. If that is the case, a new instance s_{new} is created being a valid structural implementation of FB_{new}. To create all implementations of a cascode voltage bias, every combination of diode and normal transistor with same doping having a drain-source connection is created. Because of R_f, only the diode pair and the mixed pair are recognized as valid structural implementations.

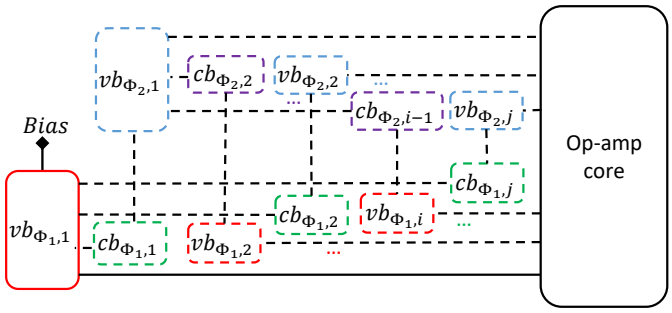


Fig. 5. Schematic overview of a bias

If a set R_a is provided, subcircuits having the defined additional connections are also created. If these circuits fulfill $\mathcal{R}_f$, the respective instances are created. Voltage reference 1 and voltage reference 2 are thus created as cascode voltage bias. Further transistor structures are created based on R_a, but are discarded as they do not fulfill $\mathcal{R}_f$.

4 SYNTHESIS OF THE OP-AMP BIAS

While Alg. 1 is used to synthesize the *op-amp core*, i.e., the amplification stages and capacitor structures with their connections, the op-amp bias b_O is created with Alg. 2.

4.1 Structure of an Op-Amp Bias

An op-amp bias b_O consists of n voltage biases (vb) and $n-1$ current biases (cb) [2]. Its generic structure is defined in Fig. 5. The voltage biases supply the op-amp core with the required voltage potentials. The current biases supply these voltage biases with a current. One voltage bias of each doping ($vb_{\Phi_1,1}$, $vb_{\Phi_2,1}$) supplies these current biases, called *distributor voltage bias* (vb_{Dis}) in the following. A single current input pin p_{Bias} remains for the user-specified bias input current. The number of voltage biases forming the bias depends on the position of the transistors in the op-amp core needing voltage supply. Five different position types are distinguished:

(1) *Improved Wilson current biases* are cascode current biases which have a diode transistor at the source and a normal transistor at the output. This type of current bias can only be connected to one implementation of a voltage bias (Fig. 3 $mp1$). Together, they form an improved Wilson current mirror (Fig. 2c $P_4 - P_7$). For each Wilson current bias in an op-amp, the specific voltage bias must be created (Alg. 2, Line 3).
(2) *Cascode current biases* can be biased by a cascode voltage bias (Fig. 2d, $N_1 - N_4, N_7, N_8$).
(3) *Simple current biases with the source connected to the supply-voltage rail* are biased by simple voltage biases (Fig. 2a, $P_7 - P_{14}$)
(4) *Simple current biases not connected to a supply-voltage rail* must be biased by an additional simple voltage bias. This is, e.g., the case in wide-swing cascode current mirrors (Fig. 2b, $N_4 - N_7$)).

A cascode current bias with doping Φ is not always biased by a cascode voltage bias. If additional single transistors of doping Φ are in the circuit needing voltage supply, the cascode current bias might be supplied by two simple voltage biases (Fig. 2c, $N_3 - N_9$).

Algorithm 2 Synthesis of op-amp bias

Require: Set of transistors without voltage supply sorted according their doping and position in the op-amp core $T_{un} := T_{un,1,\Phi_1} \cup T_{un,2,\Phi_1} \cup T_{un,1,\Phi_2} \cup T_{un,2,\Phi_2}$, Set of functional blocks forming the op-amp-core $op = \{a_1, ...\}$

1: $VB_{\Phi_1} := \{\}$ //The set of voltage biases of doping Φ_1
2: $VB_{\Phi_2} := \{\}$ //The set of voltage biases of doping Φ_2
3: $VB_{\Phi_1,iw}, VB_{\Phi_2,iw}$:=createImprovedWilsonVoltageBiases(T_{un})
4: $VB_{\Phi_1,add}$:= createAdditionalVoltageBiases($T_{un,1,\Phi_1}$, $T_{un,2,\Phi_1}$) //Alg. 3
5: $VB_{\Phi_1} := VB_{\Phi_1} \cup VB_{\Phi_1,iw} \cup VB_{\Phi_1,add}$
6: $VB_{\Phi_2,add}$:= createAdditionalVoltageBiases($T_{un,1,\Phi_2}$, $T_{un,2,\Phi_2}$) //Alg. 3
7: $VB_{\Phi_2} := VB_{\Phi_2} \cup VB_{\Phi_2,iw} \cup VB_{\Phi_2,add}$
8: setBiasPin(VB_{Φ_1}, VB_{Φ_1})
9: CB, VB_{dis} := createCurrentBiases(VB_{Φ_1}, VB_{Φ_2})//Alg. 4
10: $VB := VB_{\Phi_1} \cup VB_{\Phi_2} \cup VB_{dis}$
11: **return** CB, VB

Algorithm 3 Creating additional voltage biases

Require: Set of transistors without voltage supply sorted according their doping and position in the op-amp core $T_{un,1,\Phi}$, $T_{un,2,\Phi}$

1: $VB_\Phi := \{\}$ //The set of voltage biases of doping Φ
2: **if** $T_{un,1,\Phi} \cup T_{un,2,\Phi} = \{cb_{1,n_T=2}, cb_{2,n_T=2}, ...\}$ **then**
3: vb_Φ := createCascodeVoltageBias($T_{un,1,\Phi}$, $P_{un,2,\Phi}$)
4: $VB_\Phi := VB_{\Phi_1} \cup \{vb_\Phi\}$
5: **else**
6: $vb_{1,\Phi}$:= createSimpleVoltageBias($T_{un,1,\Phi}$)
7: $vb_{2,\Phi}$:= createSimpleVoltageBias($T_{un,2,\Phi}$)
8: $VB_\Phi := VB_\Phi \cup \{vb_{1,\Phi}\} \cup \{vb_{2,\Phi}\}$
9: **end if**
10: **return** VB_Φ

4.2 Generic Algorithm to Synthesize the Op-Amp Bias b_O

Input of Alg. 2 are the transistors of the op-amp core needing voltage supply T_{un} and the functional blocks of the op-amp core $op = \{a_1, ...\}$. The transistors in T_{un} are sorted according to their doping Φ_1, Φ_2 and their position in the op-amp core. $T_{un,1}$ contains transistors connected with their source to a supply voltage rail, $T_{un,2}$ the remaining transistors. To create the bias b_O of the fully-differential op-amp (Fig. 2a), $T_{un,p,1} = \{P_5, P_6, P_7, P_{12}, P_{13}\}$, $T_{un,p,2} = \{P_3, P_4\}$, $T_{un,n,1} = \{\}$, $T_{un,n,2} = \{N_1, N_2\}$.

The algorithm creates and connects the improved Wilson voltage biases for all Wilson current biases in the op-amp core (Alg. 2, Line 3). For the folded-cascode op-amp (Fig. 2a), no Wilson voltage bias is created as there is no Wilson current bias in the circuit.

To supply the remaining cascode and simple current biases, voltage biases are created with Alg. 3. If all remaining transistors of doping Φ in $T_{in,\Phi}$ are part of cascode current biases, a cascode voltage bias is created and connected to supply these transistors (Line 3). Otherwise, simple voltage biases are created to connect the transistors in the two sets $T_{un,1,\Phi}$, $T_{un,2,\Phi}$ (Line 6, 7). If one of the sets is empty, the corresponding voltage bias is not created. For the folded-cascode op-amp (Fig. 2a), an three voltage biases are created: P_{14} for $T_{un,p,1}$, P_{17} for $T_{un,p,2}$, N_7 for $T_{un,n,2}$. Also P_3, P_5 and P_4, P_6 form cascode current biases, as $T_{un,p,1}$ contains additional transistors not being part of a cascode current bias (P_7, P_{12}, P_{13}), no cascode voltage bias is created.

Algorithm 4 Adding current biases

Require: Set of voltage biases being of one doping with the bias pin included $VB_{\Phi_{Bias}}$, set of voltage biases being of the other doping $VB_{\Phi_{Other}}$

1: CB := { } //The set of current biases
2: VB_{Dis} := { } //Set od distributor voltage bias
3: **if** $|VB_{\Phi_{Bias}}| > 1$ **then**
4: vb_{Dis} := findDistributorVoltageBias($VB_{\Phi_{Other}}$)
5: $CB_{\Phi_{Other}}$:= createCurrentBiases(vb_{Dis}, $VB_{\Phi_{Bias}}$)
6: $CB := CB \cup CB_{\Phi_{Other}}$
7: **if** $VB_{\Phi_{Other}} \cap vb_{Dis} = \{\ \}$ **then**
8: $VB_{Dis} := VB_{Dis} \cup vb_{Dis}$
9: **end if**
10: **end if**
11: **if** $VB_{\Phi_{Other}} \neq \emptyset$ **then**
12: $CB_{\Phi_{Bias}}$:= createCurrentBiases(vb_{Bias}, $VB_{\Phi_{Other}}$)
13: $CB := CB \cup CB_{\Phi_{Bias}}$
14: **end if**
15: **return** CB, VB_{Dis}

From the created voltage biases, one voltage bias is chosen to be connected to the current bias input pin p_{Bias} (Alg. 2, Line 8). In the selection process, a single voltage bias is preferred to a cascode one, which is preferred to an improved Wilson voltage bias. This voltage bias vb_{Bias} is already set to be a distributor voltage bias for the later created current biases of the same doping (Fig. 5). For the folded-cascode op-amp (Fig. 2a), P_{14} is chosen as vb_{Bias}. Different to P_{17}, N_7, it has a source connection to the supply-voltage rail.

Alg. 4 creates the current biases of the op-amp bias b_O. Its input are the sets of voltage biases ordered according to their doping $VB_{\Phi_{Bias}}, VB_{\Phi_{Other}}$. The algorithm first creates all current biases of doping Φ_{Other}. Current biases of this doping are only needed when the number of voltage biases having the doping Φ_{Bias} is larger than one. As the current biases must be connected to a *distributor voltage bias*, a respective voltage bias must be selected in $VB_{\Phi_{Other}}$ using the same criteria as for vb_{Bias}. If $VB_{\Phi_{Other}}$ is empty or contains only voltage biases supplying transistors in $T_{un,2}$ (Fig. 2a, N_7), a simple voltage bias is created to be the distributor voltage bias vb_{Dis} (Fig. 2a, N_9). If voltage biases of doping Φ_{Other} were created during the synthesis process, current biases of doping Φ_{Bias} are created. The distributor voltage bias for them is vb_{Bias}. For the folded-cascode op-amp (Fig. 2a), $VB_{\Phi_{Bias}} = \{P_{14}, P_{17}\}$, $VB_{\Phi_{Other}} = \{N_7\}$. P_{14} is connected to the bias pin. To bias P_{17}, the current bias N_8 is created and added to $VB_{\Phi_{Other}}$. As $VB_{\Phi_{Other}}$ does not have a voltage bias connected to the supply-voltage rail, P_9 is created as distributor voltage bias of Φ_{Other}. Two current biases P_{15}, P_{16} are created to bias $N_7, N_9 \in VB_{\Phi_{Other}}$.

5 FUNCTIONAL BLOCK COMPOSITION (FUBOCO) OF A COMPLETE OP-AMP TOPOLOGY

The described algorithms (Algs. 1 - 4) are combined to synthesize complete op-amp topologies. Fig. 6 shows the composition rules to synthesize all implementations of functional blocks for single-output one-stage and two-stage op-amps. The composition rules for additional functional blocks for single-output symmetrical op-amps (e.g Fig. 2b), fully-differential op-amps (e.g. Fig. 2a) and complementary op-amps (e.g. Fig. 2c) are given in the Appendix. Structural examples synthesizable for each functional block type are given in [2]. An outlook on three-stage op-amp synthesis is given in Sec. 8.

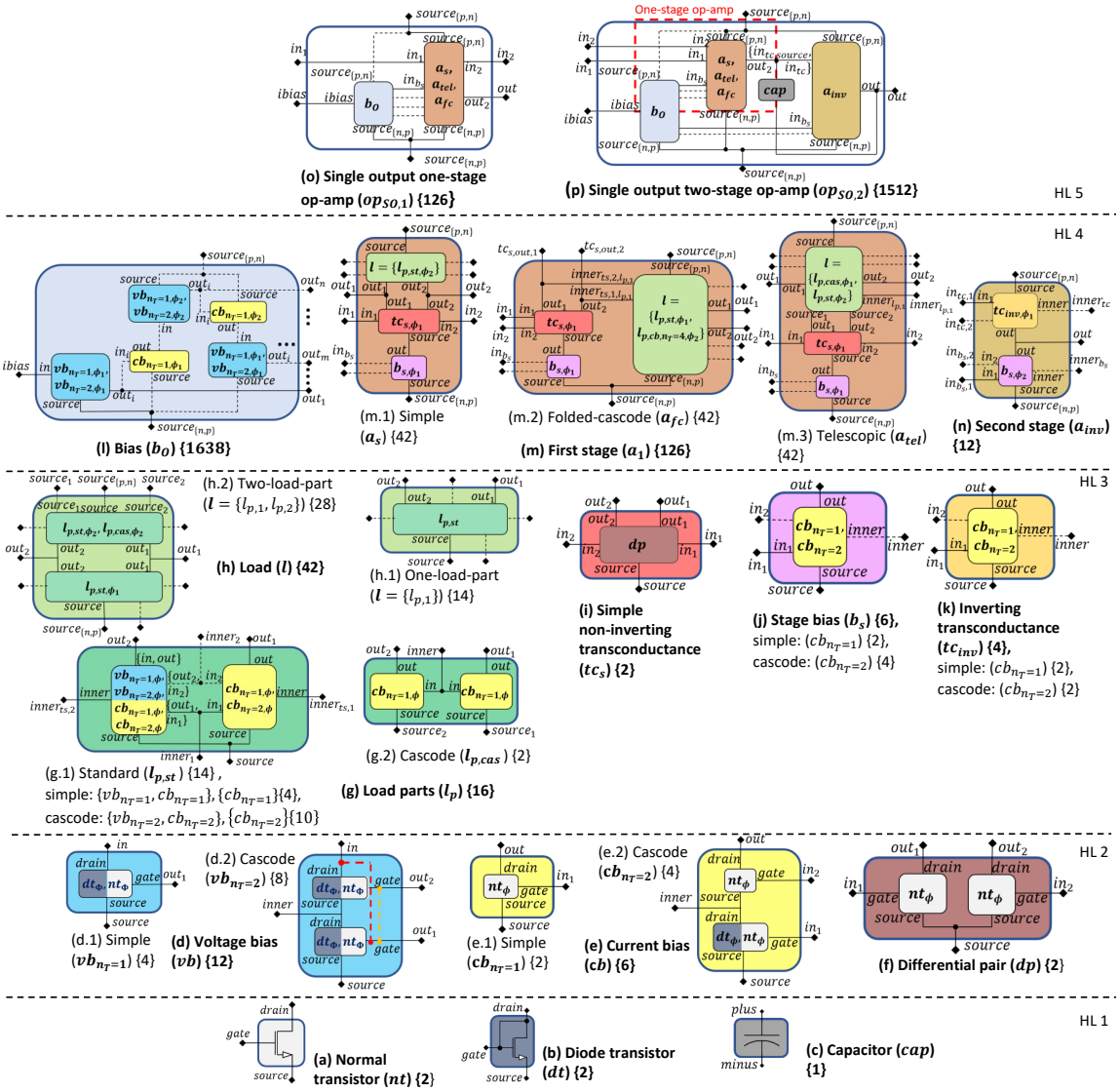


Fig. 6. Functional block composition rules for single-output op-amps. $\{n\}$ denotes the respective number of synthesizable structural implementations

5.1 Hierarchy Level 1: Devices

For every device type, instances are created (Fig 6a-c). For transistors, it is differentiated between n- and p-doping (Φ_n, Φ_p).

5.2 Hierarchy Level 2: Structures

Two types of *voltage bias (vb)* implementations (Fig. 6d) are synthesized by Alg. 1. For the *simple voltage bias* ($vb_{n_T=1}$, Fig. 6d.1), only one set implementations is inputted in the algorithm consisting of normal and diode transistors ($S_1 : NT, DT$). The synthesis of *cascode voltage bias* implementations ($vb_{n_T=2}$, Fig. 6d.2) is discussed in Sec. 3.

Current biases (cb, Fig 6e) are synthesized similar to the voltage biases. *Simple current biases* ($cb_{n_T=1}$, Fig. 6e.1) only consist of normal transistors ($S_1 : NT$, e.g. Fig. 2a P_{15}). The *cascode variant* ($cb_{n_T=2}$, Fig. 6e.2) consists either of a diode and a normal transistor (e.g. Fig. 2c P_5, P_7) or two normal transistors (e.g. Fig. 2b P_7, P_8) having a drain-source connection ($S_1 : NT_\Phi, DT_\Phi; S_2 : NT_\Phi; R_c : s_1.drain \leftrightarrow s_2.source$).

Differential pairs (dp, Fig 6f) are created using two normal transistors (nt) of the same doping Φ connected at the sources (e.g., Fig. 2c, N_1, N_2) ($S_1 : NT_\Phi; S_2 : NT_\Phi; R_c : s_1.source \leftrightarrow s_2.source;$).

5.3 Hierarchy Level 3: Amplification Stage Subblocks

Two different types of *load parts (l_p)* are synthesized for single-output op-amps (Fig. 6g):

The *standard load part* ($l_{p,st}$, (Fig 6g.1) is synthesized based on either of two current biases of the same implementation (e.g. Fig. 2c $N_4 - N_7$), or of a voltage and a current bias (e.g. Fig. 2c $P_4 - P_7$). The voltage and current biases are either simple (e.g. Fig. 2d $P_1 - P_2$) or cascode (e.g. Fig. 2a $P_3 - P_6$). The functional subblocks s_1, s_2 are connected at their sources. Also the inputs are connected or, in cases of a load part based on a voltage and a current bias, the outputs are connected to the inputs.

$S_1 : CB_\Phi, VB_\Phi;\ \ S_2 : CB_\Phi;$
$R_c : n_{T,fb_1} = n_{T,fb_2}, s_1.source \leftrightarrow s_2.source, (s_1.in_1 \vee s_1.out_1) \leftrightarrow s_2.in_1, \forall_{n_{T,s_1}=2}(s_1.in_2 \vee s_1.out_2) \leftrightarrow s_2.in_2;$

Cascode load part implementations ($l_{p,cas}$, Fig. 6g.2) are only relevant for the synthesis of telescopic op-amps. The load part is synthesized based on simple current biases (cb) being only connected at the input pins ($S_1 : CB_{n_T=1,\Phi}; S_2 : CB_{n_T=1,\Phi}; R_c : s_1.in_1 \leftrightarrow s_2.in_1;$).

Loads (l, Fig. 6h) are synthesized using either one (Fig. 2b, $P_1 - P_4$) or two load parts (Fig. 2a, $P_3 - P_6, N_1 - N_4$).

One-load-part loads implementations ($l = \{l_{p,1}\}$, Fig. 6h.1) are created with standard load parts ($S_1 : L_{p,st}$).

Two-load-part loads ($l = \{l_{p,1}, l_{p,2}\}$, Fig. 6h.2) consist of two standard load parts l_{p,st,Φ_1}, l_{p,st,Φ_2} with different doping (e.g. Fig. 2a $N_1 - N_4$, $P_3 - P_6$) or, iff the op-amp has a telescopic first stage, a standard load part l_{p,st,Φ_1} and a cascode load parts l_{p,cas,Φ_2}. They are connected at the load part outputs. ($S_1 : L_{p,st,\Phi_1}; S_2 : L_{p,st,\Phi_2}, L_{p,cas,\Phi_2}; R_c : s_1.out_1 \leftrightarrow s_2.out_1, s_1.out_2 \leftrightarrow s_2.out_2;$)

Simple non-inverting transconductances (tc_s, Fig. 6i) are synthesized based on one differential pair ($S_1 : DP$) (e.g. Fig. 2a P_1, P_2).

Stage biases (b_s, Fig. 6j) are created based on simple (Fig. 2a P_7) or cascode current biases ($S_1 : CB$).

Inverting transconductances (tc_{inv}, Fig. 6k) are based on current biases CB ($S_1 : CB;$), which are simple or cascode. No connection between the first input pin and the inner pin is allowed ($\mathcal{R}_f : s_1.in_1 \nleftrightarrow s_1.inner$).

5.4 Hierarchy Level 4: Amplification Stages

The topology-specific *op-amp bias* b_O (Fig. 6l) is synthesized using Alg. 2 after the amplification stages and capacaitors are created and connected. The bias consists of voltage and current biases (e.g. Fig. 2a $P_{14} - P_{17}$, $N_7 - N_9$).

Three different types of *first stage* implementations (a_1, Fig. 6m) are supported for simple op-amps:

Simple first stages (a_s, Fig. 6m.1) are synthesized based on a one-load-part load, a simple non-inverting transconductance and a stage bias ($S_1 : TC_{s,\Phi_1}; S_2 : B_{s,\Phi_1}; S_3 : L = \{L_{p,st}\}$). The load is of different doping Φ_2 than the transconductance and stage bias (Φ_1). The transconductance's source is connected to the output of the stage bias, while its outputs are connected to the outputs of the load ($R_c : s_1.source \leftrightarrow s_2.out$, $s_1.out_1 \leftrightarrow s_3.out_1$,$s_1.out_2 \leftrightarrow s_3.out_2$;).

Folded-cascode first stage implementations (a_{fc}, Fig. 6m.2) are synthesized with loads consisting of two standard load parts. One load part $l_{p,cb,n_T=4,\Phi_2}$ of the two-load-part load consists of current biases, has four transistors and a different doping than tc_s. This load part is connected with its inner pins of the current biases $inner_{ts_1,l_{p,1}}$, $inner_{ts_2,l_{p,1}}$ to the output pins of the transconductance.

$S_1 : TC_{s,\Phi_1}$; $S_2 : B_{s,\Phi_1}$; $S_3 : L = \{L_{p,st,\Phi_1}, L_{p,cb,n_T=4,\Phi_2}\}$;
$R_c : s_1.source \leftrightarrow s_2.out$, $s_1.out_1 \leftrightarrow s_3.inner_{ts_1,l_{p,1}}$,$s_1.out_2 \leftrightarrow s_3.inner_{ts_2,l_{p,1}}$;

Telescopic first stages (a_{tel}, Fig. 6m.3) are created with a two-load part load consisting of a cascode and a standard load part $L = \{L_{p,cas,\Phi_1}, L_{p,st,\Phi_2}\}$

$S_1 : TC_{s,\Phi_1}$; $S_2 : B_{s,\Phi_1}$; $S_3 : L = \{L_{p,cas,\Phi_1},L_{p,st,\Phi_2}\}$;
$R_c : s_1.source \leftrightarrow s_2.out$,$s_1.out_1 \leftrightarrow s_3.source_1$,$s_1.out_2 \leftrightarrow s_3.source_2$;

Second stages (a_{inv}, Fig. 6n) are composed of an inverting transconductance tc_{inv} and a stage bias b_s of different doping ($S_1 : TC_{inv,\Phi_1}; S_2 : B_{s,\Phi_2}$). They are connected at their outputs ($R_c : s_1.out \leftrightarrow s_2.out$,).

5.5 Hierarchy Level 5: Op-Amps

Two types of *single output op-amps* (op_{SO}) are supported:

One-stage op-amp implementations ($op_{SO,1}$, Fig. 6n) are created by synthesizing different first stage implementations ($S_1 : A_s, A_{tel}, A_{fc}$) with Alg. 1. The bias circuit b_O is synthesized with Alg. 2.

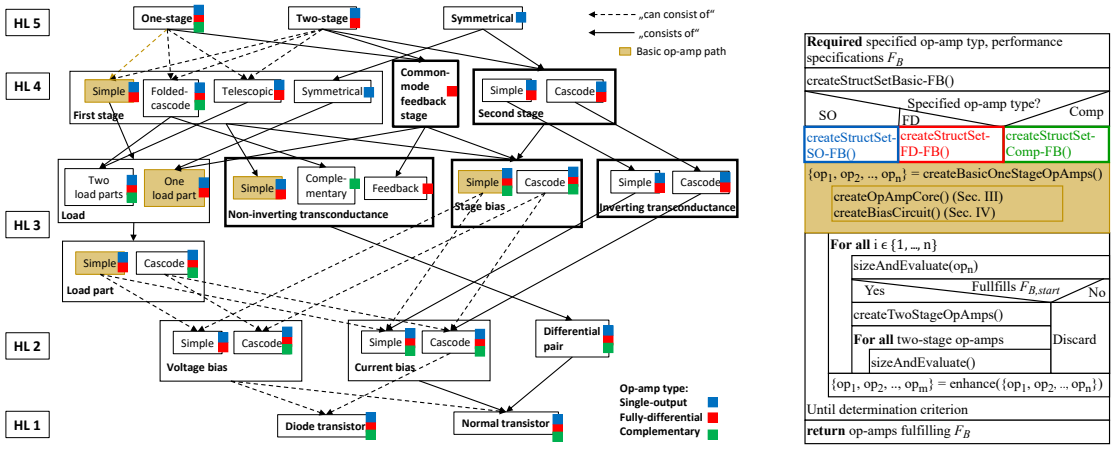
Two-stage op-amps ($op_{SO,2}$, Fig. 6p) are created by adding a second stage a_{inv} and a capacitor cap to the first stage of the one-stage op-amp using Alg. 1 ($S_1 : A_s, A_{tel}, A_{fc}; S_2 : cap; S_3 : A_{inv}$;). The capacitor is connected between the output of the first stage and the second stage, the output of the first stage is connected to the input of the second stage ($R_c : s_1.out_2 \leftrightarrow s_2.plus$,$s_1.out_2 \leftrightarrow s_3.in_{tc,1}$,$s_2.minus \leftrightarrow s_3.out$;). The bias b_O is synthesized with Alg. 2.

5.6 Number of implementations per functional block

Fig. 6 gives the maximum number of structural implementations per functional block currently supported by the synthesis algorithm. It includes n-type as well as p-type implementations and can be controlled by adding/removing rules to the set of functional block rules R_f and adding/removing functional block implementation from S_i. E.g., removing the diode transistors from S_1 of the cascode current bias leads to only two implementation of the cascode current bias instead of four, reducing also the number of synthesized stage biases and thus also the number of first and second stages as well as overall op-amp topologies.

6 OVERVIEW OF THE COMPLETE FUBOCO SYNTHESIS PROCESS

Fig. 7 gives an overview of the overall synthesis algorithm. The parts of the hierarchical functional block composition graph (Fig. 7a) marked with blue dots represent an abstraction of the composition graph for single-output op-amps (SO) in Fig. 6. The parts marked with red and green dots represent the functional blocks needed to synthesize fully-differential



(a) Functional block composition graph (without op-amp bias). HL1 - HL2: basic functional blocks, HL3 - HL4, thick frame: basic functional blocks of a specific op-amp type.

(b) Structure synthesis algorithm.

Fig. 7. Overview of the FUBOCO synthesis process.

(FD, red) and complementary op-amps (Comp, green). Details of the supplementary functional blocks are given in the Appendix. The op-amp type to be synthesized and its performance requirements F_B are specified by the user.

6.1 Functional Block Composition (FUBOCO) Graph

The composition graph (Fig. 7a) defines how to compose each functional block on level x from a set of functional blocks on level x-1 or x by combination, starting from the functional blocks on the 5th hierarchy level, i.e., one-stage, two-stage, or symmetrical op-amp. The usage of a functional block is either strict ("consists of"), or a selection of one out of many ("can consist of"). The new composition graph differs from the design plan-based structural synthesis approaches [12, 19], which feature a single solution path through such a composition graph based on if-then-else decisions, and from the structural synthesis approaches that use local structural changes with a nearly open-end process. The FUBOCO graph instead provides a large search space of yet only technically meaningful structures and uses a fast equation-based sizing process for an optimization-based selection process.

Each op-amp type features its own part of the functional block composition graph for synthesis. The respective blocks are marked with the respective color in Fig. 7a. The common-mode feedback stage (CMFB stage) for instance is only used in fully-differential op-amps (red), a folded cascode first stage is used for all three types of op-amps (blue, red, green). One-stage, two-stage and symmetrical types of single-output op-amps are considered, one- and two-stage versions of fully-differential op-amps can be synthesized, and one-stage complementary op-amps are supported. An outlook how three-stage op-amps can be integrated in the composition graph is given in Sec. 8.

6.2 Synthesis Algorithm

The synthesis algorithm, sketched in Fig. 7b, features a combination of enumerative and generative approach to structure synthesis. The evaluation of created structural op-amp variants is based on optimization over behavorial equations and is particularly fast.

Three groups of functional blocks are distinguished and specifically treated in the structural synthesis process:

Basic functional blocks are all functional blocks of HL 1 - HL 2 (Fig. 7a). They are part of many other functional blocks, e.g. non-inverting and inverting transconductance, load and stage bias. The number of implementations per functional block is small (2 - 12 for each type of functional block).

Basic functional blocks of a specific op-amp type are the functional blocks of HL 3 - HL 4 framed with thick lines. Not all functional blocks are part of every op-amp type, such that the functional blocks can be further divided into op-amp type specific groups having overlaps. The stage bias for instance is part of every group, the common-mode feedback stage is only relevant for fully-differential op-amps. The number of implementations per functional blocks is small (2 - 12).

Functional blocks with many implementations are the op-amps themselves, the first stages, loads and load parts. The number of implementations per functional block is high and varies between 24 for the load parts and 318 for the first stages.

Based on the methods presented in the preceding sections, the algorithm creates all implementations of basic functional blocks and of op-amp type-specific functional blocks upfront and stores them in a library (parts marked with blue, red, green dots in Fig. 7b). Functional blocks with a large number of structural implementation are only created on-demand when they are part of a topology. This provides a good comprise between computation time and memory usage.

The algorithm creates a set of basic one-stage op-amps with a low number of transistors. This set is marked with golden background in Fig. 7a and refers to the part similarly colored in the synthesis algorithm in Fig. 7b. The topologies are evaluated based on sizing (Sec. 6.3). Some op-amp characteristics F_{start} degenerate or do not change if a second stage is added to a one-stage op-amp. If a one-stage op-amp does not fulfill the corresponding specifications $F_{B,start}$, its two-stage versions would also not fulfill the specifications. Hence, two-stage variants are only created if these performance specifications are satisfied. Otherwise, a one-stage op-amp variant and its potential two-stage variants are discarded, thus bounding and reducing the search tree from irrelevant branches.

One-stage op-amps are enhanced by changing their stage bias and/or load. A new set of one-stage op-amps and their two-stage variants is configured and evaluated. The synthesis process ends if either the simplest op-amp fulfilling the specification is found or all op-amp topologies are enumerated.

Please note that complementary op-amps and symmetrical op-amps are synthesized slightly different. For complementary op-amps, only one-stage op-amps are currently considered. Symmetrical op-amps exist only as two-stage variants, such that the differentiation between one- and two-stage variant is not made.

6.3 Topology Sizing and Evaluation

Topologies are evaluated with an equation-based sizing method. Based on standard equations, as e.g. in [18], a behavioral model based on analytical equations for each functional block in Fig. 1 was developed leading to a hierarchical performance equation library [4]. The open-loop gain $A_{D,0}$ for examples is calculated by the multiplication of the open-loop gain of the op-amp stages n:

$$A_{D,0} = \prod_{i=1}^{n} A_{D,i};\ A_{D,i} = gm_i \cdot R_{out} \tag{2}$$

The open-loop gain of an amplification stage $A_{D,i}$ is calculated by the input transconductance of the stage gm_i and its output resistance R_{out}. As the structure of the op-amp and its amplification stages is known through the composition graph (Fig. 6), the equations for an op-amps are automatically instantiated as described in [4].

The optimization approach within initial sizing method is an enhanced version of [3]. Two groups of performance variables are defined. The group F_{start} contains all performance variables which depend only on design variables of the first stage or degrade by adding a second stage to the one-stage op-amp. These are the transistor gate-area f_D, the power consumption f_P, the maximum and minimum common-mode input voltage $f_{v_{cm,max}}, f_{v_{cm,min}}$, the common-mode rejection ratio f_{CMRR} and the phase margin f_{PM}.

$$F_{start} = \{f_D, f_P, f_{v_{cm,max}}, f_{v_{cm,min}}, f_{CMRR}, f_{PM}\} \tag{3}$$

The group F_{end} contains the remaining performance variables. These are the open-loop gain $f_{A_{D,0}}$, the slew rate f_{SR}, the unity-gain bandwidth f_{GBW} and the maximum and minimum output voltage swing $f_{v_{out,max}}, f_{v_{out,min}}$:

$$F_{end} = \{f_{A_{D,0}}, f_{SR}, f_{GBW}, f_{v_{out,max}}, f_{v_{out,min}}\} \tag{4}$$

The performance variables in the set F_{start} must fulfill the user-given constraint values $F_{B,start}$. Otherwise the topology is discarded from the synthesis process. If it was a one-stage op-amp, its two-stage variants are not considered in the synthesis process.

The variables of F_{end} are optimized towards over-fulfillment of their specifications $F_{B,end}$. If this succeeds, all performance variables $F = F_{start} \cup F_{end}$ are further optimized. In this work, the optimization algorithm is based on constraint programming and does not seek to reach an optimum but is terminated when the optimization progress slows down. Experiments have shown that after an optimization time of around one minute per structural variant the progress slows down. We therefore set the time limit for one optimization run to one minute.

The analytical equation-based sizing method allows the emulation of the manual sizing process during topology evaluation, this is different to other approaches, e.g. [20, 21], using numerical, simulation-based sizing methods [5, 23, 31].

7 EXPERIMENTAL RESULTS

We tested the synthesis tool with seven different specification sets (Table 1). FUBOCO creates for each set all topologies that fulfill the specifications. The user specifies the op-amp type (single-output, fully-differential, complementary) and the performance requirements in the specification set. Additional design knowledge is not required. All supported implementations of every functional block were allowed in the synthesis process (Sec. 5.6).

Specs 1, Specs 2 and *Specs 3* (Table 1) specify single-output as op-amp type. *Specs 1* has a high gain requirement likely to exclude many topologies. *Specs 2* demands a smaller quiescent power and gate-area. Also the requirements for slew rate and unity-gain bandwidth are more challenging. *Specs 3* requires an even smaller quiescent power. The requirements for the gain are less strict. A high phase margin is required. *Specs 4* and *Specs 5* are specified for fully-differential op-amps. In *Specs 4*, the most demanding specification is the gain being comparatively high. *Specs 5* demands a smaller area and quiescent power and a higher unity gain-bandwidth and slew rate. The op-amp type with *Specs 6* and *Specs 7* is complementary. *Specs 6* are satisfiable by many topologies variants of complementary op-amps. In *Specs 7*, the allowed area and quiescent power is reduced. The requirements for unity-gain bandwidth and slew rate increase.

Please note that only the gate-area of every transistor $t \in T$, i.e. $\sum_{t \in T} W_t L_t$, is considered as area constraint. However, also other models including more layout aspects can be used in the calculation.

Table 1. Specifications; Specs 1, Specs 4, Specs 5: general Specs; Specs 2, Specs 3, Specs 5, Specs 7: sophisticated Specs

Specs #	1	2	3	4	5	6	7
Op-amp type	Single-output			Fully-differential		Complementary	
Bias current (μA)	10	100	10	100	100	100	100
Load Capacity (pF)	20	20	20	20	20	20	20
Supply voltage (V)	5	5	5	5	5	5	5
Gate-area ($10^3\mu m^2$)	≤ 15	≤ 5	≤ 5	≤ 50	≤ 20	≤ 15	≤ 5
Quiescent power (mW)	≤ 15	≤ 8	≤ 5	≤ 25	≤ 15	≤ 10	≤ 5
Phase Margin (°)	≥ 60	≥ 60	≥ 80	≥ 60	≥ 60	≥ 60	≥ 60
CMRR (dB)	≥ 70	≥ 70	≥ 70	≥ 80	≥ 80	≥ 80	≥ 80
CMIR (V)	2-3	1.5-3.5	1.5-3.5	2-3	2-3	-	-
Open-loop gain (dB)	≥ 80	≥ 70	≥ 45	≥ 70	≥ 60	≥ 70	≥ 70
Unity-gain bandwidth (MHz)	≥ 2.5	≥ 10	≥ 10	≥ 2.5	≥ 10	≥ 2.5	≥ 10
Slew rate ($\frac{V}{\mu s}$)	≥ 3.5	≥ 20	≥ 20	≥ 3.5	≥ 15	≥ 3.5	≥ 15
Output voltage swing (V)	1.5-3.5	1.5-3.5	1-4	2-3	1.5-3.5	1.5-3.5	1-4

Table 2. Number of created topologies by FUBOCO and resulting runtime; brackets: max. # supported topologies

Specs #	1	2	3	4	5	6	7
# one-stage op-amps fulfilling $F_{B,start}$	144 (210)	96 (210)	81 (210)	39 (72)	37 (72)	36 (36)	36 (36)
# created op-amp topologies	1728 (2940)	1152 (2940)	972 (2940)	468 (936)	444 (936)	36 (36)	36 (36)
# op-amps fulfilling F_B	228 (2940)	71 (2940)	54 (2940)	34 (936)	2 (936)	10 (36)	6 (36)
Runtime	21 h	16 h	14.5 h	8 h	8 h	35 min	30 min

7.1 Synthesized Topologies

Table 2 shows the number of topologies created by FUBOCO for the different specifications. FUBOCO currently supports 2940 single-output topologies of which 210 are one-stage topologies, 936 fully-differential topologies (72 one-stage/864 two-stage) and 36 complementary op-amps. Not all topologies are created in each run, as the syntheses process does not consider two-stage op-amps if its one-stage variant fail the specifications in $F_{B,start}$ (Sec. 6.2).

Table 3 gives an overview of the composition of amplification stages in the synthesized topologies for *Specs 1* - *Specs 5*. Topologies with a large structural variety are outputted for general specifications (*Specs 1*, *Specs 4*), a smaller variety results for more specific specifications (*Specs 2*, *Specs 3*, *Specs 5*).

For *Specs 1*, the largest number of op-amps were created. Many one-stage op-amps fullfilled $F_{B,start}$ (144), such that in total 1728 op-amps were created. 228 of the 1728 topologies fulfilled *Specs 1*. Due to the high gain requirement, most of topologies are two-stage op-amps with a simple or a folded-cascode first stage (Table 3). The one-stage op-amps are either formed with a folded-cascode or telescopic first stage. The 60 symmetrical op-amps have all a cascode second stage. An example of an op-amp fulfilling the specifications is the symmetrical op-amp in Fig. 2b.

As *Specs 2* is more strict, the number of topologies fulfilling the set is much smaller: 96 one-stage op-amps fulfill $F_{B,start}$ leading to 1152 created op-amps in total. 92 of the 1152 topologies fulfill all specifications. The set is dominated by telescopic and folded-cascode one-stage op-amps and symmetrical op-amps (Table 3). Due to the strong area constraint, folded-cascode two-stage op-amps do not longer fulfill the specifications. The small number of two-stage op-amps have all a simple first stage. As the constraints on the input voltage are more demanding, all topologies in the set have a simple voltage bias as stage bias of the first stage. The symmetrical op-amp (Fig. 2b) is also a valid topology for *Specs 2*.

Table 3. Amplification stage composition of the resulting topologies

First stage type	simple a_s		folded-cascode a_{fc}		telescopic a_{tel}		symmetrical a_{sym}	total #
# stages	1	2	1	2	1	2	-	topologies
Specs 1	0	40	30	68	30	0	60	228
Specs 2	0	5	18	0	20	0	28	71
Specs 3	24	0	0	0	3	0	27	54
Specs 4	0	0	11	22	1	0	-	34
Specs 5	0	0	2	0	0	0	-	2

The set of topologies fulfilling *Specs 3* is even smaller. As 81 one-stage op-amps fulfilled the specifications in $F_{B,start}$, 972 op-amps in total were created. 54 topologies fulfilled all specifications F_B. Only one-stage op-amps and symmetrical op-amps fulfill the specifications due to the high phase margin constraint (Table 3). The one-stage op-amps have mostly a simple first stage. The three telescopic op-amps have all a simple current mirror as one of the load parts. The second stages in the symmetrical op-amps are mostly simple.

For *Specs 4*, 468 of the 936 fully-differential op-amp topologies were created in total, as 39 one-stage op-amps fulfilled the specifications in $F_{B,start}$. 34 topologie fulfilled all specifications. These are mainly folded-cascode one-stage and two-stage op-amps (Table 3). Also one telescopic one-stage op-amp fulfilled the specifications. An example of a fully-differential op-amp topology fulfilling *Specs 4* is shown in Fig. 2a. It is a folded-cascode one-stage op-amp with a pmos differential stage.

The number of topologies fulfilling *Specs 5* is smallest in this scenario: Similar to *Specs 4*, 37 topologies fulfilled $F_{B,start}$ leading to 444 created op-amps. Only two folded cascode one-stage topologies fulfilled all specifications (Table 3). One is the topology shown in Fig. 2a. The other one is an NMOS version of it. The NMOS version has a cascode stage bias in the first stage instead of a simple one.

As currently only one-stage complementary op-amps are supported by FUBOCO, all 36 topologies are created for every run specified for complementary op-amps. 10 topologies fulfill all specifications in *Specs 6*. They only vary in their type of loads and stage biases as they all have a folded-cascode first stage. The loads are mainly loads with one load part being a current mirror and the other load part containing two current biases (Fig. 2c). Various types of current mirrors as load fulfill the specifications. All types of stage biases appear in the topologies.

Six complementary op-amp topologies fulfill *Specs 7*. They do not differ much from the op-amps fulfilling *Specs 6*. The simple stage bias in the first stage dominates the set. The topology in Fig. 2c also fulfills *Specs 7*.

Please note that topologies might overfill specifications with quite high margin, as only one specification bound is given per performance feature. This can be omitted by using an upper and lower bound for each performance feature, e.g., a maximum and minimum open-loop gain requirements. This would reduce the resulting topologies to only topologies performing very close to the specifications. For specifications with low demands, e.g. *Specs 1*, a smaller number of topologies would be outputted.

7.2 Synthesis runtime

The Experiments were run on an Intel® Core™ i5-7500 CPU@3.4GHz with 32 GB RAM. The sizing of an op-amp with its optimization time in the range of minutes per circuit is the biggest time constraint of the synthesis tool. In contrast, the creation of a topology is in the range of milliseconds.

Table 4. Comparison to simulation results; Deviation in per cent; (i/j), i: # of topologies fulfilling the specs, j. total # of topologies tested

Specs #	1	2	3	4	5	6	7
Quiescent power	8% (25/30)	7% (28/28)	6% (21/23)	4% (10/10)	4% (0/2)	20% (9/10)	6% (6/6)
Phase Margin	24% (25/30)	11% (28/28)	6% (16/23)	3% (10/10)	3% (2/2)	16% (10/10)	15% (5/6)
CMRR	20% (30/30)	27% (26/28)	21% (22/23)	19% (10/10)	16% (2/2)	18% (10/10)	11% (6/6)
CMIR	(30/30)	(27/28)	(23/23)	(10/10)	(2/2)	-	-
Open-loop gain	16% (19/30)	25% (21/28)	12% (22/23)	15% (9/10)	12% (2/2)	25% (9/10)	11% (6/6)
Unity-gain bandwidth	29% (21/30)	25% (22/28)	19% (19/23)	31% (5/10)	19% (2/2)	32% (9/10)	32% (6/6)
Slew rate	15% (25/30)	24% (15/28)	16% (17/23)	20% (10/10)	26% (2/2)	22% (6/10)	36% (3/6)
Output voltage swing	(24/30)	(23/28)	(19/23)	(10/10)	(2/2)	(7/10)	(4/6)
All *Specs*	(6/30)	(8/28)	(6/23)	(5/10)	(0/2)	(2/10)	(2/6)

The runtime highly varies with the number of created op-amp topologies (Table 2). For a single-output op-amp specification set, many topologies can be synthesized and sized (2940). For such specifications, the run time of the synthesis tool is quite long (14 h - 22 h). For complementary op-amps with only 36 topologies supported, the runtime is smaller ($\sim$ 30 min). The requirements of the specifications have a great influence on the runtime. If they are more strict, the runtime decreases for two reasons:

- Strict specifications lessen the number of two-stage op-amps which are created and sized. Many one-stage op-amps do not fulfill the specifications in $F_{B,start}$. Thus, two-stage op-amps based on them are not created.
- The constraint-programming solver does not optimize circuits if early results show that the circuit does not fulfill the specifications.

As many topologies were created for *Specs 1* (1728), 21 h were needed to evaluate them. For *Specs 2*, only 1152 topologies were created at a runtime of around 16 h. For *Specs 3*, the number of created topologies was 972, and the overall runtime 14.5 h. For fully-differential op-amps, the runtime for both specification sets is equal. The total number of one-stage op-amps is smaller compared to single-output op-amps, such that the number of one-stage op-amps fulfilling F_{start} does not vary as much. For complementary op-amps, only one-stage op-amps are supported such that always 36 op-amps are created and evaluated. For more challenging specs (*Specs 7*), the runtime reduces further, as the number of topologies increases for which the sizings are not further optimized as the topologies will not fulfill the specifications.

As the runtime for specifications with many created topologies is quite long, future work remains in reducing the runtime. We are currently working on paralleling the synthesis process, sizing several op-amps at the same time. Depending on the CPU, this reduces the runtime significantly.

7.3 Sizing and Evaluation Results

A selection of the outputted topologies fulfilling all specifications were simulated to analyze how the circuits sized with analytical equations agree with simulation results (Table 4). A BSIM3v3 transistor models was used for simulations. Table 4 shows the average deviations between analytically calculated and simulated values. The deviations agree well with the expectations of a designer, who expects a deviation below 30%. The largest deviation occurs for the unity-gain bandwidth. Its value is overestimated in the sizing process as it depends linearly on the input conductance of the first stage in the analytical equations [4, 18].

Table 5. Results of the sizing method in [4] for the three-stage op-amp Fig. 2d

(a) Performance values; M: sizing method; S: simulation

Constraints	Spec.	M	S
Bias current (μA)	10	10	10
Load Capacity (pF)	20	20	20
Supply voltage (V)	5	5	5
Gate-area (10^3 μm^2)	≤15	8.3	-
Quiescent power (mW)	≤20	15	17
Phase Margin (°)	≥60	-	50
CMRR (dB)	≥70	-	180
CMIR	2-3	0-4.1	0.4-4
Open-loop gain (dB)	≥60	88	73
Unity-gain bandwidth (MHz)	≥2.5	2.8	1.9
Slew rate ($\frac{V}{\mu s}$)	≥2.5	2.6	8.4
Output voltage swing (V)	1-4	0.3-4.3	0.2-4.2

(b) Device sizes

Variable	Value (μm/fF)
$W_{P_{1,2}};L_{P_{1,2}}$	6;3
$W_{P_3}; L_{P_3}$	13;6
$W_{P_{4,5}}; L_{P_{4,5}}$	15;1
$W_{P_6}; L_{P_6}$	157;9
$W_{P_7}; L_{P_7}$	256;1
$W_{P_8}; L_{P_8}$	203;6
$W_{P_9}; L_{P_9}$	104;6
$W_{N_{1,2}};L_{N_{1,2}}$	319;3
$W_{N_{3,4}};L_{N_{3,4}}$	222;3
$W_{N_5}; L_{N_5}$	189;3
$W_{N_6}; L_{N_6}$	570;1
$W_{N_7}; L_{N_7}$	37;3
$W_{N_8}; L_{N_8}$	41;3
C_1	5
C_2	100

The absolute number of circuits fulfilling a specification respective all specs is also given. For many specification sets, already a number of topologies fulfill all specifications. Other circuits need subsequent optimization [22] to fulfill all specifications.

8 OUTLOOK: SYNTHESIS OF MULTI-STAGE OP-AMPS

The hierarchical structure of the approach can be extended to other op-amp types, e.g., multistage op-amps. Fig. 2d shows such an op-amp. To synthesize the op-amp and similar topologies automatically following input is derived for Alg. 1:

$S_1 : OP_{so,2,\backslash b_O \backslash cap}$, $S_2 : A_{inv}$, $S_3 : cap$, $S_4 : cap$; R_c :$s_1.out \leftrightarrow s_2.in_{tc,1}$,$s_1.inner_{1,l_{p,st,a_1}} \leftrightarrow s_2.in_{b_{s,1}}$,
$s_1.inner_{2,l_{p,st,a_1}} \leftrightarrow s_2.in_{b_{s,2}}$,$s_1.out_{a_1} \leftrightarrow s_3.plus$,$s_1.out_{a_2} \leftrightarrow s_4.plus$,$s_2.out \leftrightarrow s_3.minus$,$s_2.out \leftrightarrow s_4.minus$;
R_f :$s_1.\Phi_{tc,a_1} \neq s_2.\Phi_{tc}$,$s_1.\Phi_{tc,a_2} \neq s_2.\Phi_{tc}$;

The inputted functional blocks are a set of single output two-stage op-amps without their bias circuit b_O and compensation capacitor cap, $OP_{so,2,\backslash b_O \backslash cap} = \{A_1, A_2\}$, a set of inverting stages A_{inv}, which form the third stage of the op-amps and two sets of capacitors cap. The inverting stage is connected as third stage to the output of the two-stage op-amp. The stage bias of the third stage is biased by one of the load parts of the first stage l_{p,st,a_1}. The capacitors are connected between the outputs of first and third stage, and the outputs of second and third stage. The topologies are only valid if the transconductance of the third stage has different doping as the transconductance of first and second stage.

To evaluate the topologies, the sizing method in [4] must be adopted to three-stage op-amps. One of the biggest restrictions in automatic sizing of multi-stage op-amps is the phase margin. The generation of good models is still a research issue in analog design [30]. The sizing results for the three-stage op-amp (Table 5) were generated with an enhanced version of [4] without an accurate model for the phase margin. Instead common stability constraints as in [17, 28] were used. The compensation capacitors were manually adopted to improve the stability of the op-amp.

As the sizing tool provides good values for all specifications not affected by the changed capacitor values (Table 5), the synthesis tool can be well used for the synthesis of multi-stage op-amps. The results will improve when adequate models for the phase margin of multi-stage op-amps found.

The circuit shown in Fig. 2d is a simple version of a three-stage op-amp. Many other types of multi-stage op-amps with more advanced frequency compensation techniques exist [17, 24, 25, 28]. They can be added analogously to the synthesis algorithm increasing the number of supported topologies and functional blocks. However, as the research in this area is still ongoing and an adequate model for the phase margin is missing, this remains future work.

The mentioned multi-stage extension increases the underlying topology library from 3912 to 6852 practically meaningful op-amps. It shows that the library can be easily extended, which is also useful for machine learning-based approaches needing large data sets.

9 CONCLUSION

This paper presented a method for structural synthesis of op-amps by a hierarchical composition graph of functional blocks. The method emulates the manual design process. It uses knowledge described in standard analog design works [8, 13, 18, 26, 27, 29] and makes it computationally accessible. A generic algorithm to compose a functional block from blocks from the same or lower level of abstraction was presented. Rule sets for the hierarchical composition of all functional blocks span a search space of several thousand variants. It is searched by a heuristic mixture of static and dynamic structure exploration and a fast sizing method using behavioral equations [4]. The method currently supports one- and two-stage op-amps. An outlook is given how also multi-stage op-amps can be supported.

As the functional block representation of op-amps can be similarly encoded as in [9], future research based on this method can be carried out in the area of machine learning. Future applications lies in combining the structural synthesis with layout synthesis.

ACKNOWLEDGMENTS

The authors would like to thank the Cusanuswerk for partly funding this work.

A APPENDIX: IMPLEMENTATION RULES FOR OP-AMP FUNCTIONAL BLOCKS

In the following, functional blocks and their composition rules are presented needed to support also symmetrical single-output op-amps (Sec. A.1), fully-differential op-amps (Sec. A.2) and complementary op-amps (Sec. A.3).

A.1 Symmetrical Op-Amp ($op_{SO,sym}$)

Fig. 8 shows the functional blocks and their composition rules for symmetrical op-amps.

Voltage bias load part implementations ($l_{p,vb}$, Fig. 8a, HL 3) are synthesized based on two identical voltage bias implementations (vb, Fig. 2b $P_1 - P_4$) ($S_1 : VB_\Phi$; $S_2 : VB_\Phi$). The two voltage biases are connected at their sources ($R_c : s_1 = s_2, s_1.source \leftrightarrow s_2.source$). $\mathcal{R}_f : s_1.out_1 \leftrightarrow s_1.in$ omits mixed pairs (Fig. 3) as voltage bias.

Symmetrical non-inverting stages (a_{sym}, Fig. 8b, HL 4) feature one load part consisting of voltage biases l_{p,vb,Φ_2} as load. The outputs of the simple transconductance tc_s are connected to the input pins of the load.

$S_1 : TC_{s,\Phi_1}$; $S_2 : B_{s,\Phi_1}$; $S_3 : L = \{L_{p,vb,\Phi_2}\}$; $R_c : s_1.source \leftrightarrow s_2.out, s_1.out_1 \leftrightarrow s_3.in_1, s_1.out_2 \leftrightarrow s_3.in_2$;

Symmetrical op-amps ($op_{SO,sym}$) (Fig. 8c, HL 5) are synthesized by adding an inverting stage a_{inv} and an inverting stage with a voltage bias as stage bias $a_{inv,vb}$ to the outputs of a symmetrical non-inverting stage a_{sym}. The

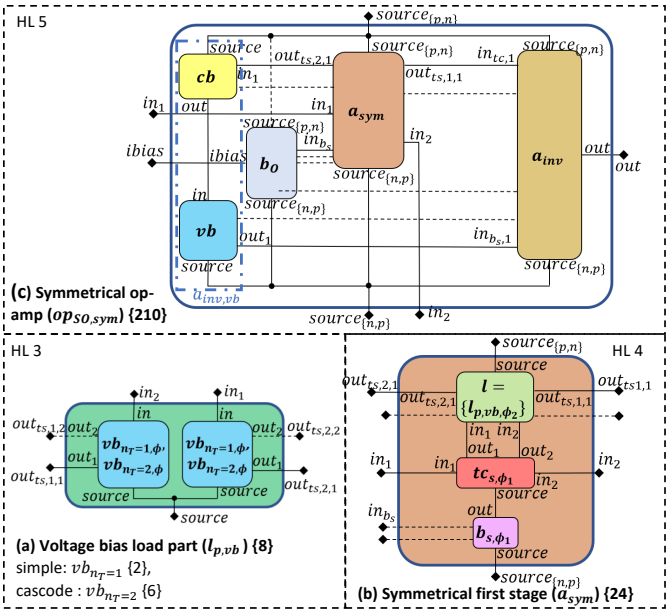


Fig. 8. Composition rules for additional functional blocks needed to synthesize symmetrical op-amps, $\{n\}$ denotes the respective number of synthesizable structural implementations

transconductances tc_{inv}, $tc_{inv,vb}$ of the inverting stages a_{inv}, $a_{inv,vb}$ must have the same doping as the load part of the non-inverting stages. The number of transistors must be equal in tc_{inv}, $tc_{inv,vb}$. The transistor sum in tc_{inv}, $tc_{inv,vb}$ must be greater than or equal to the number of transistors in the load part $l_{p,vb}$ of a_{sym}. The two stage biases $b_{s,inv}$, $b_{s,inv,vb}$ of a_{inv}, $a_{inv,vb}$ must form a current mirror cm.

$S_1 : A_{sym,l_{p,vb}.\Phi=\Phi_1}$; $S_2 : A_{inv,tc_{inv}.\Phi=\Phi_1}$; $S_3 : A_{inv,vb,tc_{inv}.\Phi=\Phi_1}$; $R_c : n_{T,tc_{inv}} = n_{T,tc_{inv,vb}}$,

$n_{T,l_{p,vb}} \leq (n_{T,tc_{inv}} + n_{T,tc_{inv,vb}}), s_1.out_{ts,1,1} \leftrightarrow s_2.in_{tc,1}, s_1.out_{ts,2,1} \leftrightarrow s_3.in_{tc,1}, s_2.in_{b_s,1} \leftrightarrow s_3.outb_s, 1,$

$\forall_{n_{T,l_{p,vb}}=2, n_{T,tc_{inv}}=2} s_2.in_{tc,2} \leftrightarrow s_3.in_{tc,2}, \forall_{n_{T,l_{p,vb}}=4, n_{T,tc_{inv}}=2} [(s_1.out_{ts,1,2} \leftrightarrow s_2.in_{tc,2}) \wedge (s_1.out_{ts,2,2} \leftrightarrow s_3.in_{tc,2}))],$

$n_{T,b_{s,inv}} \geq n_{T,b_{s,inv,vb}}, \forall_{n_{T,b_{s,inv,vb}}=2} s_2.in_{b_s,2} \leftrightarrow s_3.out_{b_s,2}$; $\mathcal{R}_f : \mathcal{R}_{f,b_{s,inv},b_{s,inv,vb}} = \mathcal{R}_{f,cm}$

The bias b_O is generated with Alg. 2.

A.2 Fully-Differential Op-Amp (op_{FD})

Fig. 9 shows additional functional blocks and their composition rules for fully differential op-amps.

Common-mode feedback transconductance implementations (tc_{CMFB}, Fig. 9a, HL 3) are generated based on two differential pairs with equal doping Φ ($S_1 : DP_\Phi$; $S_2 : DP_\Phi$, e.g, Fig. 2a, $P_8 - P_{11}$). One input of both differential pairs is connected. Also, the outputs are connected ($R_c : s_1.in_2 \leftrightarrow s_2.in_1$, $s_1.out_1 \leftrightarrow s_2.out_2$, $s_1.out_2 \leftrightarrow s_2.out_1$).

Common-mode feedback stages (a_{CMFB}, Fig. 9b, HL 4) are created based on a common-mode feedback transconductance tc_{CMFB}. The load consists of one load part with two simple voltage biases $l_{p,vb,n_T=2}$. Two identical stage biases are connected with their outputs to the respective source pins of the transconductance.

$S_1 : TC_{CMFB,\Phi_1}$; $S_2 : L = \{L_{p,vb,n_T=2,\Phi_2}\}$; $S_3 : B_{s,\Phi_1}$; $S_4 : B_{s,\Phi_1}$;

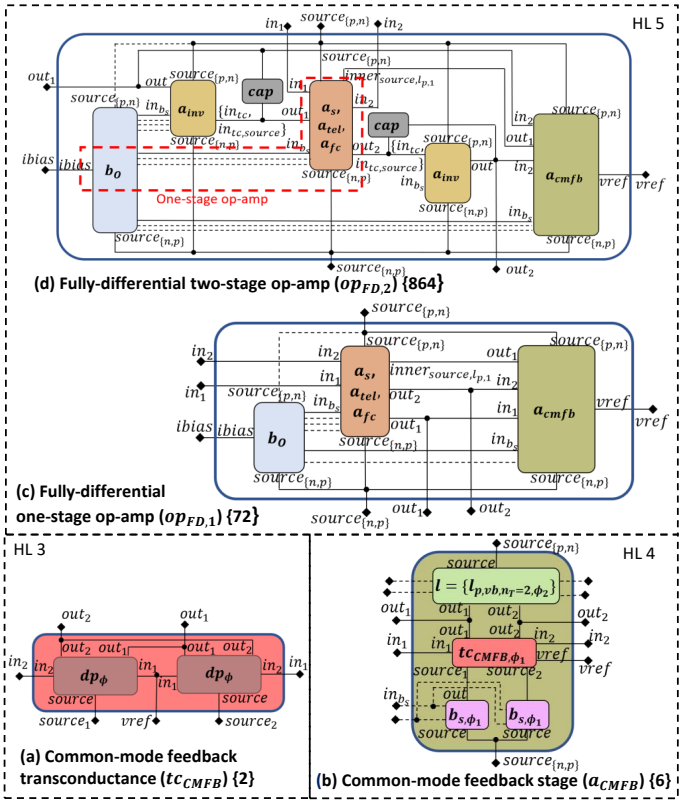


Fig. 9. Composition rules for additional functional blocks needed to synthesize fully-differential op-amps, $\{n\}$ denotes the respective number of synthesizable structural implementations

$$R_c: \; s_1.out_1 \leftrightarrow s_2.in_1, s_1.out_2 \leftrightarrow s_2.in_2, s_1.source_1 \leftrightarrow s_3.out, s_1.source_2 \leftrightarrow s_4.out, s_3 = s_4,$$

The algorithm supports one and two-stage fully-differential op-amps. *Fully-differential one-stage op-amp* implementations ($op_{FD,1}$, Fig. 9c, HL 5) are generated based on a first stage and a feedback stage as input to Alg. 1. The transconductances of the first stage and the feedback stage have the same doping Φ_1. The outputs of the first stage are connected to the inputs of the common-mode feedback stage. The output of the feedback stage is connected to the gates of the transistors at the source of the load part of the first stage having the same doping Φ_2 as the load of the feedback stage.

$$S_1: A_{s,tel,fc,tc_{ninv}.\Phi=\Phi_1}; \; S_2: A_{CMFB,tc_{ninv}.\Phi=\Phi_1}; \; R_c: s_1.out_1 \leftrightarrow s_2.in_1, s_1.out_2 \leftrightarrow s_2.in_2, s_1.inner_{1,l_{p,\Phi_2}} \leftrightarrow s_2.out_1;$$

The bias b_O is generated with Alg. 2.

Fully-differential two-stage op-amps ($op_{FD,2}$, Fig. 9d, HL 5) are synthesized by adding an inverting stage and a capacitor to each output of a first stage. The inverting stages are symmetrical. Their outputs are input to the feedback stage. The output of the feedback stage is fed to the gates of the transistors at the source of the load part of the first stage having the same doping as the load of the feedback circuit. The transconductances of the feedback circuit and the first stage have the same doping. The input to Alg. 1 is:

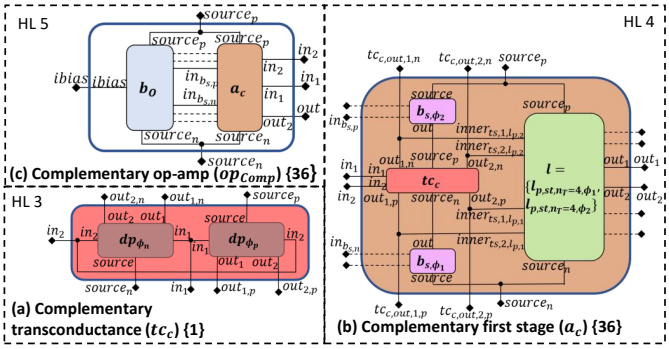


Fig. 10. Composition rules for additional functional blocks needed to synthesize complementary op-amps, $\{n\}$ denotes the respective number of synthesizable structural implementations

$S_1 : A_{s,tel,fc,tc_{ninv}.\Phi=\Phi_1}$; $S_2 : cap$; $S_3 : cap$; $S_4 : A_{inv}$; $S_5 : A_{inv}$; $S_6 : A_{CMFB,tc_{ninv}.\Phi=\Phi_1}$;
$R_c : s_4 = s_5,s_1.out_1 \leftrightarrow s_2.plus,s_1.out_2 \leftrightarrow s_3.plus,s_1.out_1 \leftrightarrow s_4.in_{tc,1},s_1.out_2 \leftrightarrow s_5.in_{tc,1},s_2.minus \leftrightarrow s_4.out$,
$s_3.minus \leftrightarrow s_5.out,s_2.out \leftrightarrow s_6.in_1,s_3.out \leftrightarrow s_6.in_2,s_1.inner_{1,l_{p,\Phi_2}} \leftrightarrow s_6.out_1$;

The bias b_O is generated with Alg. 2.

A.3 Complementary Op-Amp (op_{comp})

Fig. 10 shows additional functional blocks and their composition rules for complementary op-amps.

Complementary transconductances (tc_c, Fig. 10a, HL 3) are synthesized based on two differential pairs ($S_1 : DP_{\Phi_1}, S_2 : DP_{\Phi_2}$) having different dopings Φ_1, Φ_2 (Fig. 2c, N_1, N_2, P_1, P_2). Both inputs of the differential pairs are connected ($R_c : s_1.in_1 \leftrightarrow s_2.in_1$, $s_1.in_2 \leftrightarrow s_2.in_2$).

Complementary first stages (a_c, Fig. 10b, HL 4) are created based on a complementary transconductance tc_c. As tc_c has two source pins of different doping, two stage biases $b_{s,\Phi_1}, b_{s,\Phi_2}$ of different doping are connected with their outputs to the sources of the same doping. The stage biases should be symmetrical, i.e., have the same structural implementation by different doping. The load is a two-load-part load with eight transistors. The inner pins $inner_{ts_i,l_{p,j}}$ of current or voltage biases in the load are connected to the output of the transconductance having a different doping as the load part.

$S_1 : TC_c$; $S_2 : B_{s,\Phi_1}$; $S_3 : B_{s,\Phi_2}$; $S_4 : L_{n_{lp}=2}$ R_c :sym(s_2, s_3),$s_1.source_{\Phi_1} \leftrightarrow s_2.out$,$s_1.source_{\Phi_2} \leftrightarrow s_3.out$,
$s_1.out_{1,\Phi_1} \leftrightarrow s_4.inner_{ts_1,l_{p,2}}$,$s_1.out_{2,\Phi_1} \leftrightarrow s_4.inner_{ts_2,l_{p,2}}$,$s_1.out_{1,\Phi_2} \leftrightarrow s_4.inner_{ts_1,l_{p,1}}$,$s_1.out_{2,\Phi_2} \leftrightarrow s_4.inner_{ts_2,l_{p,1}}$;

Complementary op-amps (op_{Comp}) (Fig. 6s) are synthesized using complementary first stages (S_1: A_c). The topology specific op-amp bias is created with Alg. 2.

REFERENCES

[1] I. Abel and H. Graeb. 2020. Structural Synthesis of Operational Amplifiers Based on Functional Block Modeling. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*.

[2] Inga Abel, Maximilian Neuner, and Helmut Graeb. [n.d.]. A Functional Block Decomposition Method for Automatic Op-Amp Design. ([n. d.]). https://arxiv.org/abs/2012.09051 (Dec, 2020).

[3] Inga Abel, Maximilian Neuner, and Helmut Graeb. 2021. COPRICSI: COnstraint-PRogrammed Initial Circuit SIzing. *Integration* 76 (2021).

[4] Inga Abel, Maximilian Neuner, and Helmut Graeb. 2021. A Hierarchical Performance Equation Library for Basic Op-Amp Design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2021), 1–1. https://doi.org/10.1109/TCAD.2021.3101691

[5] Kurt Antreich, Helmut Graeb, and C. Wieser. 1994. Circuit analysis and optimization driven by worst-case distances. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (1994).

[6] F. H. Bennett, M. A. Keane, D. Andre, and J. R. Koza. 1999. Automatic Synthesis of the Topology and Sizing for Analog Electrical Circuits using Genetic Programming . In *EUROGEN workshop in Jyvdskyld, Finland.*

[7] A. Das and R. Vemuri. 2009. A graph grammar based approach to automated multi-objective analog circuit design. In *Design, Automation Test in Europe Conference Exhibition.*

[8] Ken Martin David Johns. 1997. *Analog Integrated Circuit Design.* John Wiley and Sons.

[9] Miguel Duarte-Villaseñor, Esteban Tlelo-Cuautle, and Luis de la Fraga. 2012. Binary Genetic Encoding for the Synthesis of Mixed-Mode Circuit Topologies. *Circuits Systems and Signal Processing* 31 (06 2012). https://doi.org/10.1007/s00034-011-9353-2

[10] C. Ferent and A. Doboli. 2014. Novel circuit topology synthesis method using circuit feature mining and symbolic comparison. In *Design, Automation Test in Europe Conference Exhibition (DATE).*

[11] F. Fernandez, Á. Rodríguez-Vázquez, J. L. Huertas, and G. G. E. Gielen. 1998. Structural Synthesis and Optimization of Analog Circuits. In *Symbolic Analysis Techniques: Applications to Analog Design Automation.*

[12] A. Gerlach, J. Scheible, T. Rosahl, and F. Eitrich. 2017. A generic topology selection method for analog circuits with embedded circuit sizing demonstrated on the OTA example. In *Design, Automation Test in Europe Conference Exhibition (DATE).*

[13] Paul R. Gray, Robert G. Meyer, Paul J. Hurst, and Stephen H. Lewis. 2001. *Analysis and Design of Analog Integrated Circuits* (4th ed.). John Wiley & Sons, Inc., USA.

[14] D. Guilherme, J. Guilherme, and N. Horta. 2010. Automatic topology selection and sizing of Class-D loop-filters for minimizing distortion. In *2010 XIth International Workshop on Symbolic and Numerical Methods, Modeling and Applications to Circuit Design (SM2ACD).*

[15] R. Harjani, Rob A. Rutenbar, and L. Carley. 1989. OASYS: A Framework for Analog Circuit Synthesis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (1989).

[16] F. Jiao and A. Doboli. 2015. A low-voltage, low-power amplifier created by reasoning-based, systematic topology synthesis. In *2015 IEEE International Symposium on Circuits and Systems (ISCAS).*

[17] Ka Nang Leung and P. K. T. Mok. 2001. Analysis of multistage amplifier-frequency compensation. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 48, 9 (2001).

[18] Kenneth R. Laker and Willy M. C. Sansen. 1994. *Design of analog integrated circuits and systems.* McGraw-Hill.

[19] P. C. Maulik, L. R. Carley, and R. A. Rutenbar. 1995. Integer programming based topology selection of cell-level analog circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 14, 4 (April 1995).

[20] T. McConaghy, P. Palmers, M. Steyaert, and G. G. E. Gielen. 2009. Variation-Aware Structural Synthesis of Analog Circuits via Hierarchical Building Blocks and Structural Homotopy. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 28, 9 (2009).

[21] M. Meissner and L. Hedrich. 2015. FEATS: Framework for Explorative Analog Topology Synthesis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2015).

[22] MunEDA 2009. *WiCkeD.* MunEDA. www.muneda.com

[23] Emil S. Ochotta, Rob A. Rutenbar, and L. Richard Carley. 1996. Synthesis of High-Performance Analog Circuits in ASTRX/OBLX. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (1996).

[24] G. Palmisano and G. Palumbo. 1995. An optimized compensation strategy for two-stage CMOS op amps. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 42, 3 (1995).

[25] G. Palmisano and G. Palumbo. 1997. A compensation strategy for two-stage CMOS opamps based on current buffer. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 44, 3 (1997).

[26] Gaetano Palumbo and Salvatore Pennisi. 2002. *Feedback Amplifiers* (1st ed.). Springer US, USA.

[27] Douglas R. Holberg Phillip E. Allan. 2012. *CMOS Analog Circuit Design.* Oxford University Press.

[28] J. Ramos, Xiaohong Peng, M. Steyaert, and W. Sansen. 2003. Three stage amplifier frequency compensation. In *ESSCIRC 2004 - 29th European Solid-State Circuits Conference (IEEE Cat. No.03EX705).*

[29] Behzad Razavi. 2002. *Design of Analog CMOS Integrated Circuits.* Tata McGraw-Hill.

[30] Guoyong Shi. 2017. Topological Approach to Symbolic Pole–Zero Extraction Incorporating Design Knowledge. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 36, 11 (2017).

[31] Esteban Tlelo-Cuautle, Martín Alejandro Valencia-Ponce, and Luis Gerardo de la Fraga. 2020. Sizing CMOS Amplifiers by PSO and MOL to Improve DC Operating Point Conditions. *Electronics* 9, 6 (2020).

[32] Z. Zhao and L. Zhang. 2020. An Automated Topology Synthesis Framework for Analog Integrated Circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 39, 12 (2020).