

Method of Generating Unique Elementary Circuit Topologies

D. Shahhosseini, E. Zailer, *Student Member, IEEE*, L. Behjat, *Member, IEEE*, and
L. Belostotski, *Senior Member, IEEE*

Abstract—Designing analog circuits with new topologies is often very challenging, as it requires not only circuit design expertise but also an intuition of how various elementary circuits may work when put together to form a larger circuit. In this paper, we present a method of generating all functional elementary circuit topologies. The paper uses combinatorics to ensure that all unique circuit topologies are generated and stored in a database. This database contains 582 two-transistor and 56,280 three-transistor functional and unique elementary circuit topologies. It is envisioned that the circuit topologies stored in the database can save design time and assist designers by both offering previously unknown circuit topologies and providing circuit topologies for further optimizations. To give an example of how this vision can be used in practice, a search for all amplifier circuits was conducted that resulted in 5,177 circuit topologies, some previously unknown, out of 56,862 three-transistor elementary circuit topologies.

Index Terms—Computer-Aided Design, Analog Circuit Synthesis, Integrated Circuit Design, Mathematical programming

I. INTRODUCTION

Analog circuit designers usually use a handful of known elementary circuits, containing just a few transistors, to design more complicated systems. However, as the design process is based on the experience of the designer, topologies that can produce better results may be overlooked. While, as argued in [1], approximately 1 hour of expert design time is required per each transistor in a circuit, it is not possible to estimate the time for ingenuity, expertise and work required to come up with new and unique topologies. To make the matter more urgent, the scaling of CMOS has increased the demand for new circuit topologies, which are more suitable for low voltage power supplies of scaled CMOS [2]–[5]. In this paper, we present a methodology that can assist analog designers in their search for new topologies.

Some notable works in the field of analog circuit synthesis include [6] and [7]. In [6], a Tabu-search-based algorithm is used to generate system architectures from basic building blocks such as operational amplifiers (op-amps), resistors, capacitors, etc. The system is described by functional HDL

and several topologies which fit certain criteria are generated. Transistors sizes of the op-amps are then optimized to meet specific constraints described by HDL and the best resulting circuit is chosen. In [6] the focus is on the algorithm and cost functions used to perform the evaluation versus topology generation. In addition, a generated topology can produce many circuits with equivalent signal-flow, which is listed as a limitation of the algorithm. The example used to show the possible use of their algorithm is the automatic generation of a low frequency audio circuit. The main focus of [6] is the high level synthesis not circuit topology generation. Also as the authors use Tabu search, there can be no guarantee that all possible topologies are generated. In [7], the authors address many of the limitation of the system level architecture design with proposed solutions. Basic building blocks such as amplifiers and current sources are represented by bipartite graphs and reduced into abstract blocks allowing for equivalent PMOS-NMOS circuits to be represented by a single block. The abstract building blocks are organized in a library and are used in combination to form new topologies. The transistor sizing is then handled by commercial tools.

There have been only a few attempts to find all elementary 2- and 3-transistor circuit topologies [8]–[11]. In [8]–[10], a technique is proposed to generate all possible topologies for circuits with 2 transistors by hand. Linear analog circuits were represented by small-signal 2-port networks, and a transistor was considered as a voltage controlled current source (VCCS). The circuit's 2-port network was considered as a graph and each component for the 2-port was modeled as a branch of the graph. Once an individual transistor graph was determined, all topologically different graphs were generated by hand. This method relies heavily on hand calculations and therefore is not suitable for circuit topologies with more than 2 transistors.

In [11], a systematic methodology to design all possible low noise amplifiers for ultra wideband receivers is proposed. The method in essence is an automated version of the work presented in [10], where transistors are represented as VCCS, and topologies are generated using graphs. This is the first methodology in automating the design of circuit topologies. However, it has limitations in practical applications including the following: (i) the methodology can only be used to produce amplifiers, (ii) the system equations need to be solved symbolically, which can result in very large runtime, (iii) the methodology does not recognize situations when two or more circuits are identical, i.e. the internal transistor connections are the same but the transistor labels are switched. As we will show, the duplications result in

Manuscript received on XXXX XX, 201X. This work was supported by the National Research Council of Canada, the University of Calgary, the Natural Sciences and Engineering Research Council of Canada (NSERC) grants RGPIN/358707-2013 and RT731053, Alberta Innovates-Technology Futures, computing resources for this work have been provided by Compute/Calcul Canada and CMC Microsystems.

D. Shahhosseini, E. Zailer, L. Behjat, and L. Belostotski are with Department of Electrical and Computer Engineering, University of Calgary. E-mails: delaram.shahhosseini@gmail.com, ezailer@ucalgary.ca, laleh@ucalgary.ca, lbelosto@ucalgary.ca

producing several thousands duplicated circuit topologies that have to be stored and analyzed unnecessarily. In addition to [11], recent symbolic analysis reported in [12]–[15] was also used to identify new circuits. In symbolic analysis, equations describing the circuit operation, and design parameters are produced. However, exhaustive search of the complete space of available circuit topologies was not demonstrated so far with such methods.

The work reported in this paper is considered as an extension of the works reported in [8]–[11]. Similarly to [8]–[11], this paper makes the following assumptions:

- circuit topologies are determined based on their equivalent small-signal representations, which assume that transistors are biased in saturation or subthreshold and which are topologically the same regardless whether NMOS or PMOS transistors are used. This assumption eliminates some circuits, such as ring oscillators operating in a large-signal mode and circuits where transistors are biased in triode.

- only single-ended circuits are considered as differential circuits are constructed by placing a common mode network in an otherwise grounded node of a two identical single-ended circuits. Since a differential circuit is built with single-ended circuits, for a circuit topology generation, it is sufficient to generate single-ended circuits.

- circuits have one single-ended input.

- no passive circuits, i.e. three diode-connected transistors connected together is not considered as a viable circuit.

- circuits have exactly one resistive load.

- the generated circuit topologies will be used to guide circuit designers towards achieving their design specifications by presenting them with circuit topologies whose small-signal models can be shown to meet desired specifications.

- no attempt is made to suggest methods the designers would use to size and bias circuit transistors to achieve the desired performance.

The proposed algorithm uses several techniques in order to reduce the design space and make the search more efficient. One of these methods is symmetric expansion, which produces only symmetric architectures eliminating all others. Another method is to use a multi-level circuit evaluation technique, where topologies are constrained through a set of rules, thus pre-selecting only possible functional topologies to be analyzed and sized by the commercial tool. As the result, this paper demonstrates an extension to the methods in [9]–[11] and describes a method of automatically generating *all functional* elementary circuits topologies, where functional topologies are defined as those that do not have any of the features specified in Table I.

The method of generating the elementary circuits draws heavily on combinatorics theory not only to generate all possible functional circuits, but also to prove that our database contains only *unique* circuits. To demonstrate the viability of the method, generation of 2-transistor circuit topologies is discussed in detail, followed by a procedure of extending the number of transistors in elementary circuit topologies to three. To illustrate the efficacy of the proposed method, 56,862 of 2- or 3-transistor elementary, functional, and unique circuit topologies have been generated. The method of extending 2-

transistor topologies to 3-transistor topologies is a precursor to generation of larger elementary topologies based on smaller elementary topologies. Once generated, all elementary circuit topologies are stored in a database and are available to a circuit designer to select the ones meeting any desired combination of circuit performance specifications, such as gain, bandwidth, noise, input power-matching, and power consumption, to name a few.

The main contributions of this work are:

- Introducing a novel method to generate transistor topologies.
- Generating a memory efficient database of the all 2- and 3-transistor topologies, due to sparse nature of symmetric matrices describing the topologies.
- Proving that the database is complete and does not contain duplicate entries.
- Providing an example of how mathematical techniques can be used with the database to find all possible amplifying topologies.

Furthermore, as an *illustration* of a potential use for these circuit topologies, a mathematical programming technique was used to select all input-power-matched amplifier circuits out of the 56,862 circuit topologies. This selection process resulted in 5,177 unique input-power-matched amplifiers. An example of one of these amplifiers, which was not previously known, was published by our group in [16].

This work starts with an overview of the proposed circuit generation approach in Section II. In Sections III and IV, the proposed methodology for producing all functional and unique 2- and 3- transistor topologies is discussed. An example of usage the database of 3-transistor circuits for selecting power-matched circuits with voltage gain of more than 1 is demonstrated in Section V. Concluding remarks are given in Section VI.

II. MOTIVATION

When designing a large analog circuit, the circuit designer resorts to a set of known circuit topologies to construct the required circuit. An example of one such circuit, a fully differential folded-cascode operational transadmittance amplifier, is shown in Fig. 1. While the circuit looks fairly complicated, it is built of elementary circuits, such as current mirrors (CM), current sources (CS), differential amplifiers (Diff-Pair), common-source amplifiers (CSA), and cascode pairs (CP). To design this circuit for a set of given specifications, the designer would first bias the circuit and then conduct a small-signal hand analysis to check whether the circuit performance is near what is required. If so, a more sophisticated simulation based on BSIM transistor models is used to fine-tune and optimize the circuit using a number of possible optimizations [17]–[25]. If, however, it is determined that, for example, the current mirror *CM1* does not provide sufficiently large output resistance or requires overly large voltage for operation, the designer would normally resort to another current mirror topology, perhaps a low-voltage cascode current mirror [26], or some lesser known current mirror as in [27]. If none of the known current mirror topologies provide sufficient

Table I
DEFINITION OF NON-FUNCTIONAL CIRCUITS AS USED IN THIS WORK.

input and output connected with a short circuit	circuit has no connection to I/O ports
the load is connected only to a transistor gate	two or three gates are connected and not connected to anything else
input is connected to ground or the power supply	the gate terminal of a transistor is connected to its source

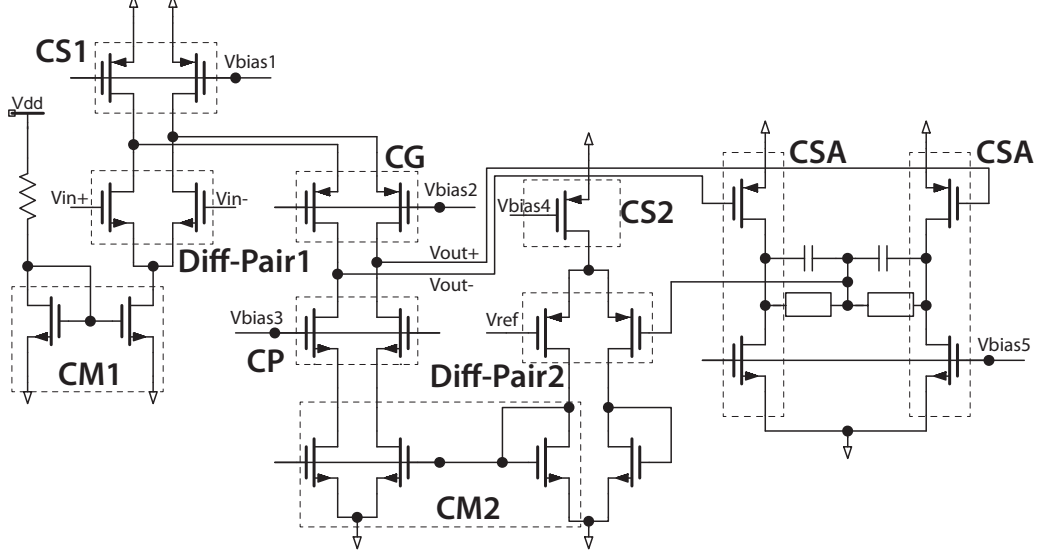


Figure 1. Fully differential folded-cascode operational transadmittance amplifier with a common-mode feedback. The elementary circuits that make up the design are Current Mirror (CM), Current Source (CS), Common Source Amplifier (CSA), Common Gate amplifier (CG), and Cascode Pair (CP). Biasing for current sources CS1 and CS2 are normally generated by other elementary circuits such as current mirrors, which are not shown in this figure to reduce the schematic complexity.

performance, the specifications of the larger circuit would have to be relaxed or a new current mirror topology would have to be created. The same considerations apply to other elementary circuit topologies constituting the larger circuit.

This work is seeking to assist the designer in searching for such new circuit topologies by making a database of all elementary circuit topologies and showing a technique of how a search can be conducted to find those topologies that have either specific usage, such as amplification, current mirror, etc, or specific performance such as input power matching. To do this, a two-step process is envisioned:

Step 1: Construction of all possible elementary circuit topologies

Step 2: Selection of circuit topologies that meet desirable performance.

The focus of this work is on Step 1. Only a limited investigation of the feasibility of Step 2 is performed in this work, as the complete implementation of Step 2 is beyond the scope of this study.

The easiest way to construct all possible circuit topologies is to use an exhaustive search where all possible topologies are first generated, and then, the topologies that do not result in functional circuits are eliminated. However, this can result in very large runtime. For example, a circuit made of 3 transistors and 1 load has 13 nodes: $3 \times 3 = 9$ transistor nodes, and 4 input/output IO nodes including Ground, GND , input voltage, V_{in} , two side of the load, L^+ , and L^- . This means

that there are $2^{\frac{13^2-13}{2}} \approx 3.2 \times 10^{23}$ ways to make connections between the nodes. However, as it will be shown later in this work, only 56,280 of these topologies are functional and unique. This means that a large amount of time has to be spent on eliminating over 10^{23} non-functional circuits.

To show how the proposed *constructive* algorithm produces all functional circuit topologies, the case for the 2-transistor is described in detail in the following. The proposed method is made of two phases as described in the following:

Phase I: Transistor Connection Generation: In this phase, all internal and external connections inside and between transistor blocks that can result in functional circuits are produced. This phase is explained in detail in Section III.

Phase II: IO Connection Generation: In this phase, the connections internal to the input/output (IO) block is first generated. Then, the transistors and the IO blocks are connected to each other to make functional topologies. This phase is explained in detail in Section IV.

III. PHASE I: TRANSISTOR CONNECTION GENERATION

A. Internal transistor, T_i , Configurations

The transistor block (T_i) includes three ports: Gate, G_i , source, S_i , and drain D_i . Theoretically, there are 5 ways to connect these 3 ports: no connections, G and S are connected, G and D are connected, S and D are connected, and all three ports, G , S , and D , are connected. However, to ensure the functionality of the transistor block, the internal connections

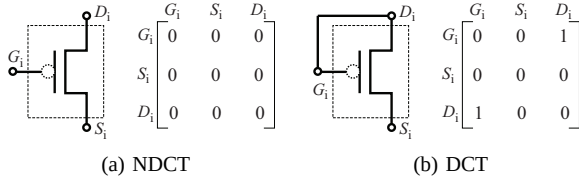


Figure 2. The possible NMOS or PMOS transistor block T_i configurations and their connectivity matrices.

should be constructed based on circuit and transistor operation rules:

- There can be no connections between S and D , as this is a short-circuit.
- There can be no connections between G and S as it turns off transistor's small-signal transconductance.

Therefore, from the possible 5 ways to internally connect a transistor only 2 possible configurations are suitable for functional circuits. These configurations are:

- *Not diode-connected transistor (NDCT)*: No internal connection.
- *Diode-connected transistor (DCT)*: G and D are connected.

These connections have generally been represented using graphs. However, graphs are not easy to store or manipulate. In order to represent these connections efficiently, we propose to use connectivity matrices. A connectivity matrix for a transistor is a 3×3 matrix where the rows and the columns represent the ports of the transistor in the following order: 1) G , 2) S , 3) D . The elements of the matrix are 0 or 1, where a 1 indicates a connection between the nodes of the transistors and 0 signifies no connection. In Fig. 2, the two possible internal configurations for a transistor and their corresponding connectivity matrices are shown. Note in this figure and all other figures in this paper, the transistor symbol with a dashed circle at the gate indicated either an NMOS or a PMOS transistor. Using connectivity matrices instead of graphs enables us to efficiently make and store all circuit topologies using combinatorics principals. In addition, we are able to only construct unique circuits thus reducing the database size by over 32,000 topologies compared to [11].

B. External T_i/T_j Configurations

Once the internal connections of each transistor are made, transistors need to be connected to each other, i.e. external transistor configuration. We propose a constructive algorithm where only functional configurations are produced. This algorithm is based on the non-attacking Rooks theorem and starts with the generation of all possible connections between 2 transistors, i.e. 2-transistor topologies. The 2-transistor topologies are divided into groups: (i) both transistors are NDCT, (ii) one transistor is DCT and one transistor is NDCT. The case of 2DCT is not considered as it results in a circuit consisting of two non-linear resistances. Connection of the two transistor had to ensure that there was no modification to transistor internal topology and that no identical circuits were generated. Once a topology is made, the designer can

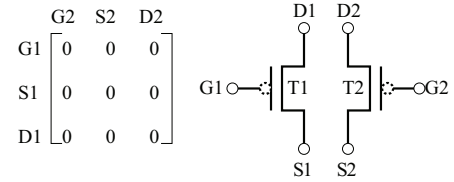


Figure 3. Connectivity matrix and NMOS and/or PMOS transistor topology when both transistors are NDCT and the number of internal connections is zero ($k = 0$).

decide whether an N or P type transistor should be used. The proposed techniques to build these connections and proofs that the resultant circuit topologies are unique are given in the following.

1) *Construction of 2 NDCT Topologies: Lemma 1: Construction of all 2-transistor topologies with 2 NDCTs is equivalent to the constructions of 3×3 matrices in which each row and each column have at most 1 non-zero element.*

Proof: The connectivity matrix describing the connections between two transistors is a 3×3 matrix. If there are more than 2 non-zeros in a row/column of the connectivity matrix, then the corresponding ports of the transistor represented by the non-zero row or column are connected. However, an NDCT has no internal connections. Therefore, at most 1 non-zero element is allowed in any row or column. ■

Based on Lemma 1, to construct all 2-transistor topologies when the transistors are NDCT, we have to produce all matrices, which have at most one non-zero element in each row or column.

Lemma 2: *Constructing the connectivity matrices for 2 NDCTs is equivalent to finding a solution to the non-attacking Rooks problem on a 3×3 board. The number of rooks in the problem varies from 0 to 3.*

Proof: Based on Lemma 1 the connectivity matrix between two transistors, T_i/T_j , can have only one non-zero element in each row or column. This is equivalent to a non-attacking Rooks problem (only 1 Rook in each row or column) of appropriate size [28], [29]. The number of rooks, k , for the NDCT case varies from 0 to 3, where $k = 0$ means no connections between the 2 transistors and $k = 3$ means that the 3 ports of the transistors are connected to each other. ■

In the rest of this section, topologies that can be produced when k varies between 0 to 3 are described. In all following 4 cases, the transistors are NDCT.

2 NDCTs, $k = 0$ (Transistors have no connections): In this case, the two transistors are not connected and all elements of the connectivity matrix are zero. The connectivity matrix and the associated 2-transistor configuration are shown in Fig. 3. For this case, $k = 0$, there is only one possible topology. The resulting 2-transistor topology has 6 free ports that need to be connected individually to the ports of IO blocks. However, as it will be shown in the following section, for the three possible configurations of the IO blocks, there is a maximum of 4 ports leaving at least 2 of the 2-transistor block ports unconnected, thus violating the operational circuit rule of no open circuits. Hence, this configuration is not applicable (NA) for circuits with 2 transistors.

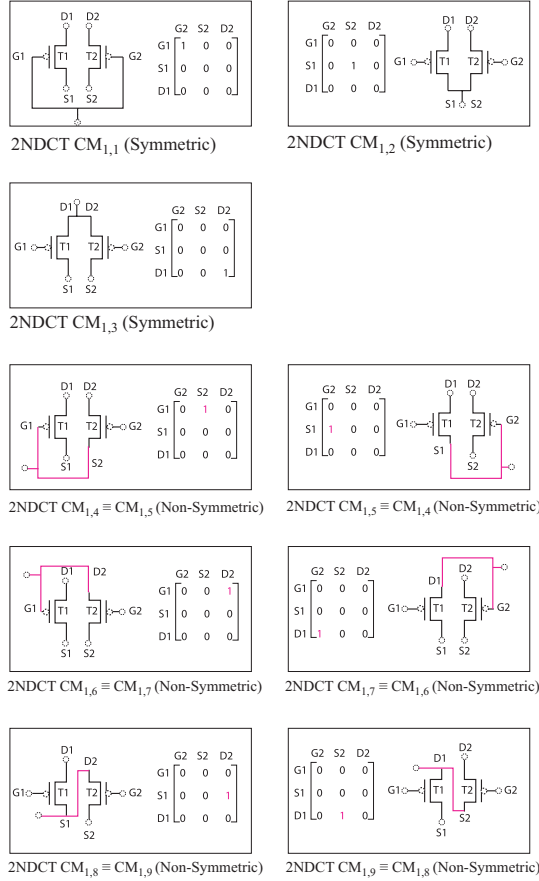


Figure 4. Connectivity matrix and transistor topology when both transistors (PMOS and/or NMOS) are NDCT and the number of internal connections is one ($k = 1$).

2 NDCTs, $k = 1$ (Transistors have 1 connection): In this case, the two transistors are connected with only one connection. Equivalently, in the Rooks problem, there is only one rook on a 3×3 board. All 2-transistor topologies having only one connection between the two NDCTs are shown in Fig. 4. These topologies have 5 ports to be connected to the IO ports. However, IO blocks have a maximum of 4 ports. Hence, to be able to connect the circuits together, the connection between the 2 transistors must be considered as an internal connection, i.e. not connected to an IO block.

2 NDCTs, $k = 2$ (Transistors have 2 connections): This case is equivalent to 2 rooks on a 3×3 board. There are a total of 18 topologies for this case as shown in Fig. 5. The resultant topologies can have 2, 3, or 4 external ports, and depending on the internal connections may also be connected to the IO block. Since, as we will show later in Section IV, the minimum number of the IO block ports is limited to at least 3, the 2-transistor topologies with 2 external ports are not considered.

2 NDCTs, $k = 3$ (Transistors have 3 connections): Finally, the case for the 2 NDCTs connected with 3 connections, is equivalent to the Rooks problem with 3 rooks on a 3×3 board. There are a total of 6 topologies for this case and each topology can only have 3 ports as shown in Fig. 6. So far, we have shown, through enumeration, all possible 2-

NDCT connections. Next, we prove that we have produced all possible configurations for 2-NDCT topologies.

Lemma 3: *The number of all possible 2-NDCT topologies is 34.*

Proof: Based on Lemma 2, building the connectivity matrix between two transistors is equivalent to finding all possible non-attacking Rook polynomial solutions. The number of ways to place 0 to k rooks on an $m \times n$, $n \leq m$, board, where $k = \min\{n, m\} = n$, can be calculated by the Rook polynomial [28], [29]

$$R_{m,n}(x) = n!x^n L_n^{m-n}(-x^{-1}), \quad (1)$$

where x represent the set of instances possible for each k , i.e. x is not a variable but a representation of a set of instances in combinatorics, and

$$L_n^{m-n}(x) = \sum_{i=0}^n (-1)^i \binom{n + (m-n)}{n-i} \left(\frac{x^i}{i!}\right) \quad (2)$$

is the Laguerre polynomial of degree n . In the NDCT case, $m = n = 3$, and the number of rooks can change between 0 to 3, i.e. $k = 0, 1, 2, 3$, and the Rook polynomial equation can be rewritten as

$$R_{3,3}(x) = 6x^3 + 18x^2 + 9x^1 + 1x^0, \quad (3)$$

where x^i represent the set of instances for a problem containing $i \leq k$ rooks. The coefficient of each x^i is equal to the number of ways that the i non-attacking rooks can be placed on the board. Based on (3), there is only 1 topology, which is identified by $1x^0$, for 2 unconnected transistors, i.e. $k = 0$. When the 2 transistors have only 1 connection, $i = 1$, there are 9 topologies, as identified by $9x^1$. For $i = 2$ and $i = 3$, there are 18, $18x^2$, and 6 topologies $6x^3$, respectively. Hence, the total number of topologies in this case is $6 + 18 + 9 + 1 = 34$, which was shown empirically in Figs. 3-6. ■

Uniqueness of circuit topologies: Some of topologies shown in Figs. 4-6 are equivalent circuit topologies. These equivalent topologies have their transistor order switched, which results in 2 identical circuit topologies having 2 different connectivity matrices. Examples of these circuits are shown in matching colours and an arrow between the switched elements of the connectivity matrix. Some examples are: $CM_{1,4}$ and $CM_{1,5}$ in Fig. 4, $CM_{2,7}$ and $CM_{2,8}$ in Fig. 5, and $CM_{3,5}$ and $CM_{3,6}$ in Fig. 6. Because of the equivalent circuits, the 34 topologies in Lemma 3 is only an upper bound on the total number of topologies that can be constructed with 2 NDCTs. The exact number of circuit topologies is calculated using Theorem 1.

Theorem 1: *The number of unique topologies for two NDCTs is 23.*

Proof: In the proposed topology generation technique switching the order of 2 transistors is equivalent to switching the row and column of their connectivity matrix. Therefore, 2 circuits are identical if their connectivity matrices are transpose of each other. Since in the Rook polynomial the symmetric matrices are accounted for only once, to find identical matrices, symmetric and non-symmetric matrices should be separated to eliminate the non-symmetric matrices whose transpose matrices have already been built.

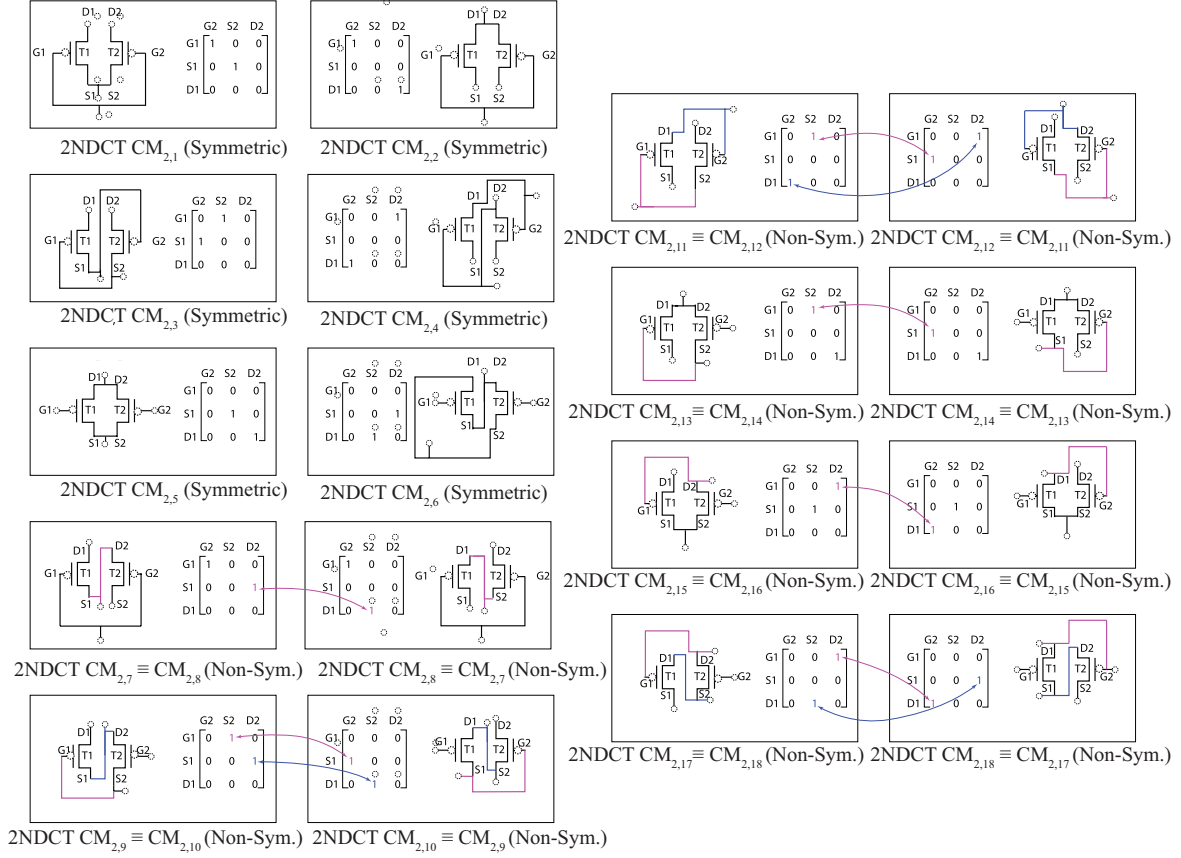


Figure 5. Connectivity matrix and transistor topology when both transistors (PMOS and/or NMOS) are NDCT and the number of internal connections is two ($k = 2$).

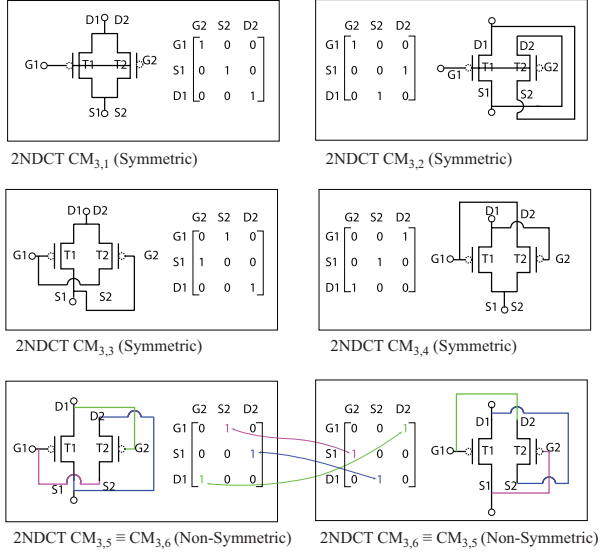


Figure 6. Connectivity matrix and transistor topology when both transistors (PMOS and/or NMOS) are NDCT and the number of internal connections is three ($k = 3$).

The number of symmetric matrices for 2 NDCTs can be calculated based on the number of rooks, k , in the connectivity matrix. When $k = 1$, the only symmetric matrices are when the rook is placed on the diagonal of the connectivity matrix,

i.e. 3 choices. This means that out of the remaining $9 - 3 = 6$ topologies, half of them are transpose of the others. Therefore, the number of unique topologies between 2 transistors with only 1 connection is the sum of the symmetric cases and a half of the non-symmetric cases: $3 + \frac{9-3}{2} = 3 + 3 = 6$.

For $k = 2$, symmetric matrices result when the rooks are either both on the diagonal, i.e. 2 out of 3 choices, or in the opposite locations above and below the diagonal, i.e. 1 out of 3 choices. Therefore, the number of symmetric matrices is: $3 + 3 = 6$. The total number of unique topologies is: $6 + \frac{18-6}{2} = 12$.

When $k = 3$, symmetric matrices result when either all 3 rooks are on the diagonal, i.e. 1 choice, or 1 rook is on the diagonal and 2 rooks are in the opposite off-diagonal locations. In addition, since from Lemma 1 there can be only 1 rook in each row/column, for each rook on the diagonal, there is only 1 non-diagonal symmetric arrangement possible. The number of symmetric matrices for $k = 3$ is $1 + 3 = 4$. The total number of unique topologies is: $4 + \frac{6-4}{2} = 4 + 1 = 5$.

The case of $k = 0$ is not considered as for this case there is 6 transistor ports to be connected to 4 IO ports. This results in two unconnected transistor ports. The unconnected ports make this topology non-functional. Therefore, the total number of topologies is: $6 + 12 + 5 = 23$. ■

2) Construction of 1 DCT and 1 NDCT topologies: These circuit topologies are constructed by following the same

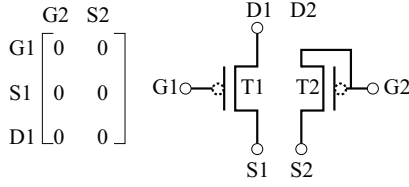


Figure 7. Connectivity matrix and transistor topology with 1 NDCT, 1 DCT, and the number of internal connections is zero ($k = 0$).

procedure and by using Rooks theorem to ensure that all unique topologies are made. However, in this case, the internal connection of the DCT has to remain intact. The following two Lemmas are proven and used to automate the connectivity matrix generation.

Lemma 4: Construction of all 2 transistor topologies with 1 NDCT and 1 DCT is equivalent to constructing 3×2 matrices in which each row or column has at most 1 non-zero element.

Proof: In a DCT, the gate and drain are connected; therefore, the column representing the drain is exactly the same as the column representing the gate and can be eliminated during the topology generation. Hence, the connections between the two transistors can be represented by a 3×2 matrix. ■

Lemma 5: Construction of connectivity matrices for a NDCT and a DCT is equivalent to finding solutions to the non-attacking Rooks problem on a 3×2 board where the number of rooks varies between 0 and 2.

Proof: Based on Lemma 1, the connectivity matrix between the two transistors can have only 1 non-zero element in each row or column. This is equivalent to a non-attacking Rooks problem (only 1 rook in each row or column) of an appropriate problem size. The number of rooks, k , varies between 0 and 2. $k = 0$ means no connections between transistors and $k = 2$ means that the 2 ports of the DCT are connected to 2 ports of the NDCT. For a DCT and NDCT, the number of connections cannot be 3, $k \neq 3$, as the drain and the gate of DCT are connected, so in effect the DCT has only two ports. ■

In the rest of this section, each one of the cases when the connection between the two transistors changes from 0 to 2 is further explained.

1 NDCT and 1 DCT, $k = 0$ (Transistors have no connections): In this case, the two transistors are completely separated and all the elements of the 3×2 connectivity matrix are zero. This means that for the case when $k = 0$ there is only one possible topology and the resulting 2-transistor topology have 5 free ports, as shown in Fig. 7, that need to be connected to the IO block which makes this topology not eligible if the connection between D2 and G2 is considered as an external port. If the D2 and G2 are considered as internal connections, then this circuit will have 4 external ports.

1 NDCT and 1 DCT, $k = 1$ (Transistors have 1 connection): In this case, there is only one connection between the two transistors. Equivalently, in the Rooks problem, there is only one rook on a 3×2 board. The resulting 2-transistor topology has 3 ports if the connection between the 2 transistors is considered as the internal connection. Otherwise, the topology contains 4 ports. The 2 transistor topologies are shown in

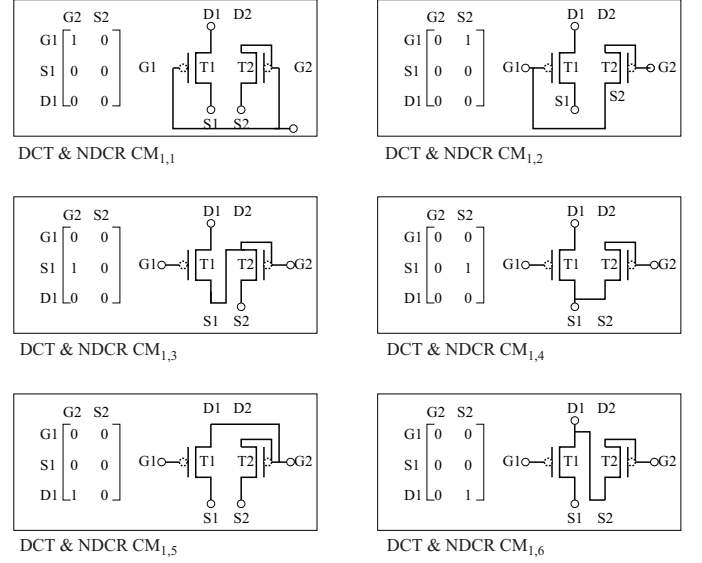


Figure 8. Connectivity matrix and transistor (PMOS and/or NMOS) topology with 1 NDCT and 1 DCT and one internal connection ($k = 1$).

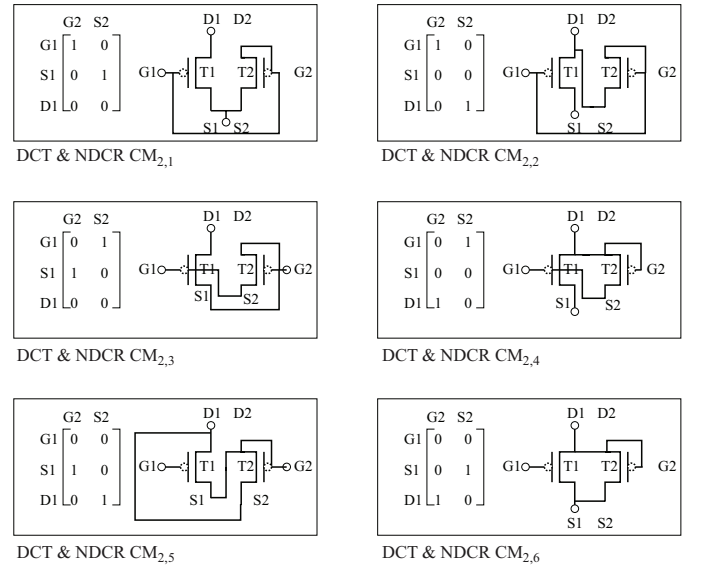


Figure 9. Connectivity matrix and transistor (PMOS and/or NMOS) topology with 1 NDCT and 1 DCT and two, $k = 2$, internal connections.

Fig. 8.

1 NDCT and 1 DCT, $k = 2$ (Transistors have 2 connections): This case is equivalent to 2 rooks on a 3×2 board. All 2 transistor topologies are shown in Fig. 9. When the 2 transistors are connected to each other with 2 connections, the resultant topologies can have 1, 2, or 3 ports, depending on whether the internal connections are also connected to the IO block or not. However, as will be shown later in Section IV, the topologies with only 1 or 2 ports cannot be connected to the IO blocks and are not considered.

1 NDCT and 1 DCT, $k = 3$ (Transistors have 3 connections): For the 1 NDCT and 1 DCT case, there cannot be 3 internal connections, as it violates the connection between gate and drain for the DCT transistor.

Table II
STATISTICS FOR GENERATED 2-TRANSISTORS TOPOLOGIES

	2 NDCTs			1 NDCT & 1 DCT		
	Total circuits	unique circuits	Possible # ports	Total circuits	unique circuits	possible # ports
$k = 0$	1	1	6	1	1	4,5
$k = 1$	9	6	4, 5	6	6	2, 3, 4
$k = 2$	18	12	2, 3, 4	6	6	1,2,3
$k = 3$	6	5	1,2,3	NA	NA	NA
Total	34	24	-	13	13	-

Uniqueness of circuit topologies: Theorem 2: The number of unique topologies to connect a NDCT to a DCT is 13.

Proof: Based on Lemma 5, building the connectivity matrix between two transistors is equivalent to finding all possible non-attacking Rooks solutions. The number of ways to place 0 to k rooks on an $m \times n$, $n \leq m$, board, where $k = \min\{n, m\} = n$, can be calculated by the Rook polynomial stated in (1), where $m = 3$, $n = 2$. Hence, the Rook polynomial is

$$R_{3,2}(x) = 6x^2 + 6x^1 + 1x^0. \quad (4)$$

Based on (4), there is 1 topology for 0 connection, and 6 topologies for 1 and for 2 connections between transistors. The total number is $6 + 6 + 1 = 13$. In the 1 DCT and 1 NDCT case, there are no symmetric matrices as all connectivity matrices are non-square 3×2 matrices. Therefore, all generated topologies are unique, and the number of unique topologies is 13. ■

3) *Summary of 2-Transistor Topology generation:* A summary of 2-transistor topology generation for the 2 NDCTs, and 1 NDCT & 1 DCT cases is given in Table II. In this table, the first column indicates the number of connections between the transistors, k , columns 2 and 5 represent the total number of topologies that can be generated, columns 3 and 6 show the number of the unique topologies, and columns 4 and 7 list the number of output ports. In this table, the case for 1 NDCT and 1 DCT and $k = 3$ is stated as not applicable (NA) based on the discussions above.

4) *Construction of 3-transistor topologies:* To show how the proposed methodology can be modified to produce topologies with more transistors, the methodology for producing functional and unique 3-transistor topology is discussed next. To construct only unique and functional 3-transistor topologies, each of the 36 already made 2-transistor topologies (23 containing 2 NDCTs and 13 containing 1 NDCT and 1 DCT) is considered as a unique building block. Then, the third transistor is added to the building block by producing the connectivity matrix T_1/T_3 . The connectivity matrix T_2/T_3 is built based on T_1/T_2 and T_1/T_3 . There is no need to make separate T_2/T_3 matrices as these matrices are duplicates of T_1/T_3 with only transistors T_1 and T_2 orders switched.

In order to ensure that the resultant 3-transistor topologies are functional, the same rules as in Section III-B are followed. In addition, to ensure that only functional topologies are constructed and stored, the number of possible external ports is considered. If a certain way of connecting the third transistor results in the number of external ports becoming

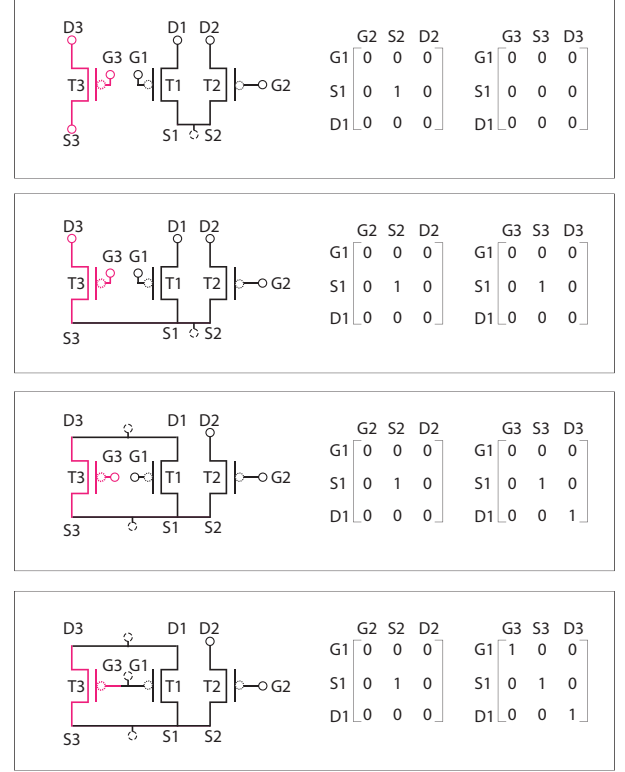
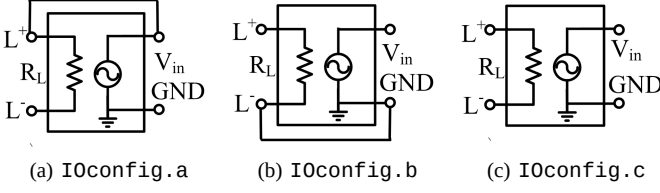


Figure 10. An example of how a third transistor (PMOS or NMOS) is added to a T_1/T_2 combination. All the ports that have to be connected to the input/output unit are shown with solid lines. The ports that can be considered as internal circuit connections are shown with dotted lines. The top two sub-figures show zero and one connections between the T_1/T_2 and T_3 , which result in more than 4 input/output ports and hence are not acceptable.

more than 4 (maximum number of input/output ports that are available) the topology is not considered. For example, if two NDCT transistors are connected by only one connection, i.e. $k = 1$, the resultant topology has at a minimum 4 ports. To connect the third transistor to the T_1/T_2 combination, at least 2 connections are required, otherwise the resultant topology has 5 external ports, which cannot be connected to the 4 ports of the input/output unit. Fig. 10 shows an example of how a 2-transistor topology with 1 connection, $CM_{1,3}$, can be connected to the third transistor. In the top two sub-figures, zero and one connections are used to connect the third transistor to $CM_{1,3}$. The resulting topologies have more than 4 ports to be connected to the input/output unit and hence these topologies are not considered when making 3-transistor topologies. On the other hand, the bottom two sub-figures show the case when 2 or 3 connections are used and hence their maximum number of input/output ports is 4 or less. In Fig. 10, all ports that need to be connected to the input/output unit are shown with solid lines and the ports that can be considered as internal connections are shown with dotted lines. It should be mentioned that all possible topologies are produced, even the ones where the basic 2 transistor topologies were not eligible to be connected to the input/output unit.

To produce 4- or 5-transistor topologies, the same procedure can be applied. The construction of 4-transistor topologies is

Figure 11. The possible internal *IO* block configurations.

	GND	V _{in}	L ⁺	L ⁻
GND	0	0	0	0
V _{in}	0	0	0	1
L ⁺	0	0	0	0
L ⁻	0	1	0	0

	GND	V _{in}	L ⁺	L ⁻
GND	0	0	1	0
V _{in}	0	0	0	0
L ⁺	1	0	0	0
L ⁻	0	0	0	0

	GND	V _{in}	L ⁺	L ⁻
GND	0	0	0	0
V _{in}	0	0	0	0
L ⁺	0	0	0	0
L ⁻	0	0	0	0

(a) IOconfig.a (b) IOconfig.b (c) IOconfig.c

Figure 12. Connectivity matrices for internal *IO* block configurations.

basically reduced to adding one transistor a 3-transistor block or connecting two 2-transistor blocks such that the number of output terminals are at a maximum 4. The construction of 5-transistor blocks can also be reduced to adding one transistor to a 4-transistor block, or adding a 2-transistor block to a 3-transistor block while considering the number of input/output terminals. The proposed methodology is very efficient because duplicate or non-functional blocks are not constructed from the beginning, hence eliminating the need for subsequent pruning.

IV. PHASE II: IO CONNECTION GENERATION

A. Internal *IO/IO* block Configurations

The input-output (*IO*) block includes 4 ports: Ground (*GND*), input voltage (*V_{in}*), and two sides of the load (*L⁺* and *L⁻*). To ensure the functionality of *IO* blocks, the following rules need to be followed:

- *GND* cannot be connected to *V_{in}*.
- Both sides of the load (*L⁺* and *L⁻*) cannot be connected to *GND* or *V_{in}* simultaneously.
- Either of the load ports cannot be connected to *GND* and *V_{in}* simultaneously.

These rules restrict all possible internal connections for the *IO* block to only 3 configurations:

- **IOconfig.a:** *L⁺* is connected to *V_{in}*.
- **IOconfig.b:** *L⁻* is connected to *GND*.
- **IOconfig.c:** No internal *IO* connections.

The three *IO* block configurations are shown in Fig. 11.

It should be mentioned that *L⁺* and *L⁻* are interchangeable, and the two configurations of *L⁻* connected to *V_{in}* and *L⁺* connected to *GND* are not considered. It is also important to note that the output can be taken between any two nodes in a circuit topology, as long as one of them is neither *GND* nor *V_{in}*. In this case the circuit topology would be considered as having a differential and/or a common-mode output. To represent these configurations, the connectivity matrix for each configuration is derived and shown in Fig. 12.

(a) Connectivity matrix

	GND	V _{in}	L ⁺	L ⁻	G1	S1	D1	G2	S2	D2	G3	S3	D3
GND	0	0	1	0	0	1	0	0	0	1	0	0	0
V _{in}	0	0	0	0	1	0	0	0	0	0	0	0	0
L ⁺	0	0	0	0	0	1	0	0	1	0	0	0	1
L ⁻	0	0	0	0	0	0	0	0	0	0	0	0	0
G1					0	0	0	0	0	0	0	0	0
S1					0	0	1	0	0	0	0	1	0
D1					0	1	0	0	0	0	0	0	0
G2						0	0	0	0	0	0	0	0
S2						0	0	1	0	0	0	1	0
D2							0	1	0	0	0	0	0
G3								0	0	0	0	0	0
S3								0	0	0	0	0	0
D3									0	0	0	0	0

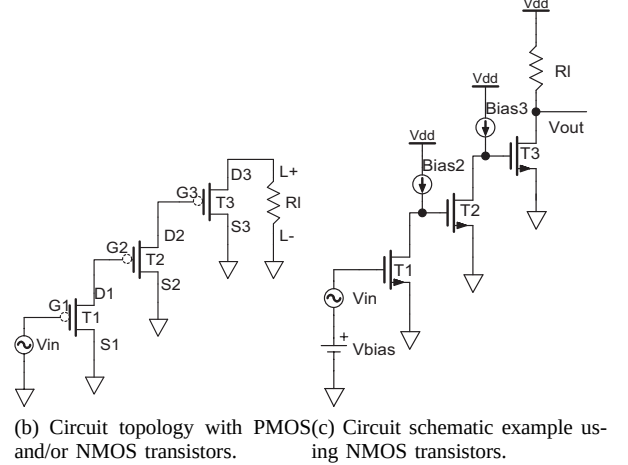


Figure 13. An example of an analog circuit and its connectivity matrix.

B. External *IO/T* Connection Generation

Once the *IO* blocks and 2- or 3-transistor topologies are made, they are connected to each other to form complete circuit topologies. To generate these connections automatically, all connectivity matrices produced so far are arranged in a larger connectivity matrix, *CM*, that describes the connectivity between all elements of the circuit topologies. The rows and columns of *CM* are ordered as follows: *GND*, *V_{in}*, *L⁻*, and *L⁺* of the *IO* block, followed by the ports of Transistor blocks, *G_i*, *S_i*, *D_i*, where *i* ∈ {1, 2} for 2-transistor circuits and *i* ∈ {1, 2, 3} for 3-transistor matrices. These diagonal *CM*s show the internal connections of the corresponding **IOconfig.a** to **IOconfig.c** of the *IO* blocks and the connections of the transistor blocks. The connections between transistors, *T_i/T_j*, are represented by 3 × 3 matrices, which are located right above the diagonal transistor *T_i* matrices. Matrices *T_i/T_j* and *T_j/T_i* are transpose of each other and all other diagonal matrices constituting *CM*s are symmetric. This means that *CM*s are symmetric matrices, and the lower diagonal elements of *CM*s need not be stored. An example of a diagonal *CM* of a 3-transistor circuit is illustrated in Fig. 13(a). Since in the figure, the *IO* block is **IOconfig.b** is used, only the matrix element corresponding to *GND* connection to *L⁻* is set to 1. All three transistors in the circuit are NDCT, hence their three bottom diagonal matrices in Fig. 13(a) only contain 0. In Figs. 13(b) and (c), the circuit topology and its biased schematic are shown, respectively.

To complete CM , the connection between the ports of the IO and T_i are the only remaining connections to be made. These connections are represented by IO/T_i matrix, located in rows 1 to 4, and columns 5 and onward of the CM matrix. The number of rows in the IO/T_i is equal to 4 as we only have 4 ports for the IO block. The number of the columns of this matrix is equal to the number of transistor ports in the circuit: 6 for the 2-transistor circuits and 9 for the 3-transistor circuit.

To generate the IO/T matrix, each IO configuration is considered separately. For each configuration, different combinations of connections between the ports of the IO and T blocks are produced based on the following rules:

- no short or open circuits,
- no alteration to the internal block topologies.

The maximum number of circuits that can be produced depends on number of ports in the IO configuration and the number of ports in the transistor configuration. For example, **IOconfig.a** has 3 ports. If the transistor block of 2 NDCTs has only one internal connection ($k = 1$) then the transistor block has 5 ports if all connections are considered as external ports and 4 ports if the one connection is considered as an internal transistor-block connection (see Fig. 4). In this case, it is not possible to connect this topology to the IO configurations **IOconfig.a** and **IOconfig.b**.

In the rest of this section, the number of possible circuits for each IO configuration and 2-transistor blocks is calculated.

1) *Circuits for IOconfig.a and IOconfig.b*: The **IOconfig.a** and **IOconfig.b** have 3 ports, therefore, only transistor topologies that have 3 ports can be connected to these configurations. For each 3-port transistor topology, there will be $3!=6$ ways to be connected to each **IOconfig.a** or **IOconfig.b**. In the following we consider the case for 2 NDCTs and 1 DCT & 1 NDCT cases, which are summarized in Table II.

2 NDCTs:

- $k = 0$: The number of ports for the T block is always 6. Hence, there are no 3-port transistor topologies that can be connected to **IOconfig.a** and **IOconfig.b** configurations.
- $k = 1$: There is 1 connection between the two transistors, hence the number of ports for the transistor block can be 5 or 4. These topologies cannot be connected to **IOconfig.a** and **IOconfig.b** configurations, which have 3 ports.
- $k = 2$: When the two transistors are connected internally by two connections, the 12 unique topologies shown in Fig. 5 can have 3 ports as long as one of the connections is considered as internal, i.e. not connected to the IO block. Then, there are $3!$ ways to connect the three ports of the IO block to the 3 remaining ports of each of the T blocks. In total we have $2 \times 3!$ ways to connect the IO block to each T block. As we have 12 unique T blocks, in total there are $12 \times 2 \times 3! = 144$ circuit topologies.
- $k = 3$: There are 5 transistor topologies that have 3 ports. There exist $3!$ ways to connect the three IO block ports to the three ports of the 5 unique transistor topologies.

Table III
THE NUMBER OF FUNCTIONAL AND UNIQUE CIRCUITS FOR 2-TRANSISTOR TOPOLOGIES. FOR THESE IO CONFIGURATIONS, THE NUMBER OF CIRCUIT TOPOLOGIES IS EQUAL TO THE TOTAL NUMBER OF TRANSISTOR TOPOLOGIES WITH 3 PORTS TIMES $6=3!$.

	k	2 NDCTs		1 NDCT & 1 DCT	
		3-port T topologies	tot Cir.	3-port T topologies	tot Cir.
IOconfig.a or IOconfig.b	k=0	0	0	0	0
	k=1	0	0	9	54
	k=2	24	144	6	36
	k=3	5	30	-	-
Total	-	-	174	-	90

Hence, in total there are $3! \times 5 = 30$ unique circuit topologies.

1 NDCT & 1 DCT:

- $k = 0$: In this case the number of ports for the transistor block is 5. Hence, this transistor topology cannot be connected to the IO configurations.
- $k = 1$: In this case the number of ports can be 3 if one of the connections between the transistor topologies shown in Fig. 8 is considered as internal and not connected to the IO . In the case of $CM_{1,1}$, $CM_{1,3}$ and $CM_{1,5}$ once one of the connections is considered as internal, the other one is automatically also considered as internal. For the other three cases, $CM_{1,2}$, $CM_{1,4}$ and $CM_{1,6}$, there is a choice for which connection can be considered as internal, so each topology can count as 2 unique topologies. Therefore, we have in total 9 unique 2-transistor topologies containing 1 NDCT & 1 DCT and one internal connection. In total for the 9 unique topologies and each of **IOconfig.a** or **IOconfig.b** configurations, there are $3 \times 3! + 3 \times 2 \times 3! = 9 \times 3! = 54$ circuit topologies that can be made.
- $k = 2$: In this case, there are 6 topologies as shown in Fig. 9, and each one has exactly 3 ports. Therefore, there are $6 = 3!$ unique circuit topologies.

A summary of these calculations are given in Table III.

2) *Circuits for IOconfig.c*: The **IOconfig.c** has 4 ports, therefore, only transistor topologies that have 4 ports can be connected to this IO configuration. There are $4! = 24$ ways to connect the 4 ports of the IO block to 4 ports of a transistor block. However, as the two sides of the load L^+ and L^- are symmetric and can be interchanged, the total number of unique ways to connect the IO block to each transistor topology are $\frac{4!}{2} = 12$. In the following we show the calculations of the number of circuits for 2 NDCTs and 1 DCT & 1 NDCT cases, which were summarized in Table II.

2 NDCTs:

- $k = 0$: The number of ports for the T block is 5. Hence, this transistor topology cannot be connected to the IO configuration.
- $k = 1$: The number of ports can be 5 or 4. Overall, for the 6 topologies, shown in Fig. 4, there are $6 \times \frac{4!}{2} = 72$ unique circuit topologies.
- $k = 2$: Each of the 12 transistor topologies shown in Fig. 5 has 4 ports. Hence, there are $12 \times \frac{4!}{2} = 144$ unique circuit topologies.

Table IV

THE NUMBER OF UNIQUE CIRCUIT TOPOLOGIES FOR 2-TRANSISTOR TOPOLOGIES AND `IOconfig.c`. FOR `IOconfig.c`, THE NUMBER OF CIRCUITS IS EQUAL TO THE NUMBER OF THE TRANSISTOR TOPOLOGIES WITH 4 PORTS TIMES $\frac{4!}{2} = 12$.

	k	2 NDCTs		1 NDCT & 1 DCT	
		4-port T topologies	tot Cir.	4-port T topologies	tot Cir.
IOconfig.c	k=0	0	0	1	12
	k=1	6	72	6	72
	k=2	12	144	0	0
	k=3	0	0	-	-
Total	-	-	216	-	84

- $k = 3$: The transistor block has a maximum of 3 ports, and hence there is no way to connect it to the `IOconfig.c` and make circuits that do not contain an open circuit.

1 NCDT and 1 DCT:

- $k = 0$: The number of ports for the T block can be 4 if one gate-drain connection, such as G_2 to D_2 connection in Fig. 7, is considered as an internal connection. Hence, there are $1 \times \frac{4!}{2} = 12$ circuit topologies.
- $k = 1$: The number of ports can be equal to 4 for each of the 6 available topologies, as shown in Fig. 8. In total, there are $6 \times \frac{4!}{2} = 72$ circuit topologies.
- $k = 2$: The maximum number of ports is equal to 3, hence no circuit can be made.

A summary of the number of 4-port topologies and circuits for `IOconfig.c` are given in Table IV.

C. Results

From Tables III and IV, it can be said that the number of unique circuit topologies that can be built with two transistors is $2 \times (174 + 90) + 216 + 84 = 828$. However, some of these circuits are not functional as defined in Table I.

In Table V, columns 2 and 3, the maximum number of circuit topologies possible with 2 transistors (max) and the number of functional circuits (func), when both transistors are NDCTs are given, respectively. Similarly, columns 5 and 6 show the number of topologies for 1 NDCT and 1 DCT configurations.

Once all transistor topologies are built, they are connected to the three possible *IO* configurations. The number of circuits with 3 transistors are shown in columns 10 to 15 of Table V. For the 3-transistor circuit topologies, three transistor configurations are considered: (i) all transistors are NDCTs (3 NDCTs), (ii) two transistors are NDCTs and one is a DCT (2 NDCTs & 1 DCT), (iii) one transistor is an NDCT and two are DCTs (1 NDCT & 2 DCTs). The cases where all three transistors are DCTs is not considered as the resultant circuit topologies will be equivalent to a combination of three non-linear resistors. This procedure can be extended further to 4- and 5-transistor topologies by adding more transistors one at a time. However, it was not attempted in this work.

In total, 56,280 functional circuit topologies with 3 transistors and 582 functional circuit topologies with 2 transistors are produced. The number of 3-transistor circuits is significantly

lower (36% lower) than 88,347 circuits generated in [11], due to avoiding the construction of identical and non-functional circuits that is made possible by using combinatorics and mathematical techniques in designing the circuit topologies.

V. Potential uses of the circuit topologies

Having created all unique elementary 2- and 3-transistor circuit topologies, we next briefly examine the possibility of performing Step 2, identified in Section II, to select elementary circuits based on their desired functions. This step would allow an analog designer with an aid of a proper circuit topology selection tool to scan through the database of the circuit topologies and to select the ones meeting certain performance requirements. Note that the design of a flexible topology selection tool is beyond the scope of this work and this section is just a demonstration of the concept.

Out of many possible elementary circuit functionalities, such as gain, bandwidth, noise, power consumption, linearity, etc, two functionalities, i.e. input power match and gain, were chosen to demonstrate the feasibility of Step 2. Of course, these goals are not the only design goals possible, but rather chosen to illustrate the envisioned use of the database of elementary circuit topologies in assisting an analog designer.

In this example, we use mathematical programming techniques used for non-linear programs such as [30]–[34] discussed in the Appendix to find the best transistor transconductances and output conductances in all generated circuit topologies. Note that symbolic analysis could also be used for this purpose [12]–[15]. Since the runtime was not critical for the demonstration of the potential use of the generated topologies, the mathematical problem is solved using MATLAB optimization lab toolbox. MATLAB optimization toolbox uses techniques such as interior point methods [35] to solve constrained optimization problems. This toolbox is suited for problems with small number of variables, less than 10 in our problems, and can produce optimal solutions in reasonable runtime¹. Note that each mathematical programming problem needs to be solved only once as the results can be stored in the database for future use.

For this demonstration, the maximum number of circuit topologies, which can be designed to be input-power-matched amplifiers, has been determined as shown in columns labeled “amp” in Table V. A total of 74 two-transistor circuit topologies and 5,103 three-transistor circuit topologies are shown the potential to be designed into amplifiers with the stated specifications.

Based on the mathematical programming problem described in (VI.1) to (VI.6) of Appendix, the majority of these topologies are able to achieve the intrinsic transistor gain of $g_m g_{ds}^{-1} = 10$. However some of these circuit topologies are capable of achieving gains of over 140. In Fig. 14, a histogram for the maximum gains, which are potentially achievable by the 3-transistor circuit topologies, is given.

The optimization has identified 32 input power-matched amplifiers with gain higher than 140 as shown in Fig. 14.

¹This process can be easily parallelized as each problem is independent from the others thus reducing runtime significantly.

Table V
THE NUMBER OF MAXIMUM, FUNCTIONAL, AND AMPLIFIER CIRCUIT TOPOLOGIES.

	2-transistor circuit topologies								3-transistor circuit topologies									
	2 NDCTs				1 NDCT & 1 DCT				total		3 NDCTs		2 NDCTs & 1 DCT		1 NDCT & 2DCTs		total	
	max	func	amp		max	func	amp		func	amp	func	amp	func	amp	func	amp	func	amp
<i>IOconfig.a</i>	174	142	16		90	74	15		216	31	6338	606	6340	803	2140	274	14818	1683
<i>IOconfig.b</i>	174	132	16		90	74	15		206	31	6338	614	6340	788	1920	274	14598	1676
<i>IOconfig.c</i>	216	112	8		84	48	4		160	12	16440	822	8354	658	2070	264	26864	1744
Total	564	386	40		264	196	34		582	74	29116	2042	21034	2249	6130	812	56280	5103

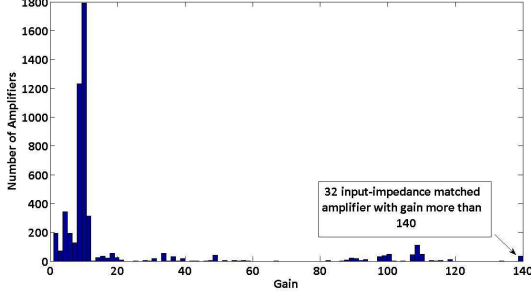


Figure 14. Histogram of maximum gains of all 3-transistor circuit topologies.

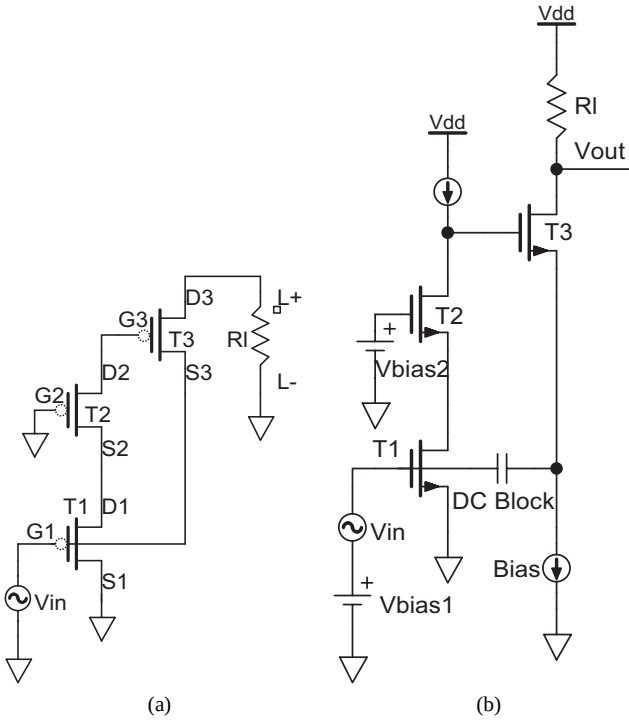


Figure 15. Example of the highest-gain three-transistor circuit: (a) Circuit topology with PMOS and/or NMOS transistors. (b) Example of a biased schematic with NMOS transistors.

The circuit topology that achieves the highest gain is already known and its circuit diagram based on NMOS transistors is shown in Fig. 15.

In Figs. 16(a) and (b), the circuit topology and its schematic of a previously unknown circuit are shown. The circuit in Fig. 16(b) was developed in CMOS technology using BSIM4 transistor models and was shown to have the characteristics that were calculated by proposed methodology [16]. The

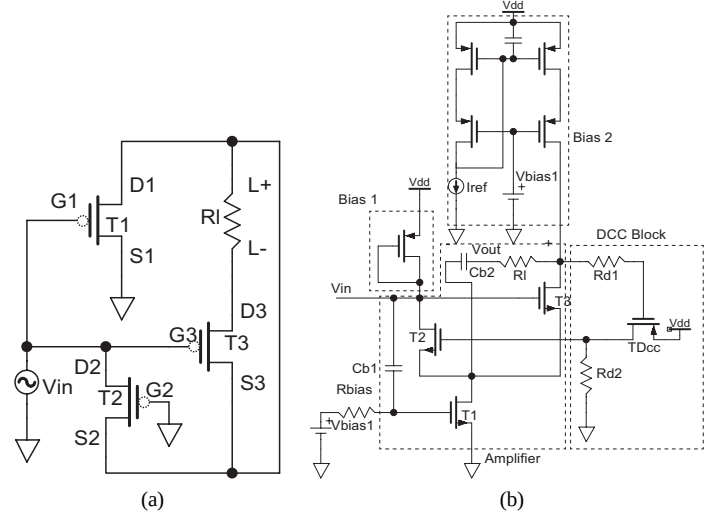


Figure 16. Example of a previously unknown three-transistor circuit identified with a search for amplifiers. (a) Circuit topology based on its CM. (b) Biased schematic. The amplifier consists of transistors T1, T2, and T3 and the load RL. Elementary circuits “Bias 1” and “Bias 2” were added to provide biasing for the amplifier. Also, “DCC Block” is a DC common-mode feedback circuit, which consists of common-source amplifier, required for biasing T2 [16]. Cb1 and Cb2 are DC blocking capacitors.

Table VI
SMALL-SIGNAL TRANSISTOR PARAMETERS FOR AMPLIFIER IN FIGURE 15(B) TARGETING 0.13- μ M CMOS.

Specification	Optimal Value	Circuit parameters	Optimal Value
A_f	90.1 (39 dB)	g_{m1} (A/V)	0.055*
A_{rev}	0	g_{m2} (A/V)	0.001
$R_{in}(\Omega)$	100	g_{m3} (A/V)	0.055*
$R_o(\Omega)$	181	$R_L(\Omega)$	10k

* Max. g_m 's were limited to 0.055 A/V to limit power consumption.

gains and other small-signal characteristics of this previously unknown circuit as optimized and implemented by analog designers in [16] are given in Table VI. To design the circuit, the small-signal parameter obtained with the MATLAB optimization were used as the guide for corresponding small-signal parameters required for each transistor. Transistor biasing and sizes were selected to achieve these small-signal parameters, which were confirmed with BSIM4 simulations. In this particular circuit implementation NMOS transistors were selected for $T_{1,2,3}$. The circuit biasing was accomplished with three elementary circuit topologies, which are identified in Fig. 16(b) as “Bias 1,” “Bias 2,” and “DCC Block” and which are discussed in details in [16]. The final circuit in Fig.

16(b) looks very different from its elementary-circuit topology in Fig. 16(a) and includes DC blocks Cb1 and Cb2, current sources, and the DCC network needed to prevent transistors in the Bias 2 network from entering triode. Clearly designing and simulating this circuit requires some conventional design work, but the *creative task* of identifying the new circuit topology and recommending its small-signal component parameters was automated.

VI. CONCLUSION

In this paper, an automated methodology to generate functional and unique circuit topologies is discussed and 582 and 56,280 possible 2- and 3-transistor topologies are generated. The results of the proposed methodology is a database that includes all functional and unique topologies. Once the circuit topologies were generated, they can be used to select elementary circuits for building larger analog circuits.

As an example of such a procedure, all generated circuits were searched by using a mathematical programming procedure to select only those which act as input-power-matched amplifiers. This search resulted in 74 two-transistor and 5,103 three-transistor amplifier circuits.

It is envisioned that having all elementary circuit topologies available to analog designers, they will be able to perform appropriate selection procedures to select circuit topologies capable of meeting certain design requirements needed for their work. The future work can be searching the topology database for other criteria such as gain bandwidth, noise figure, power consumption and other types of circuits such as passive circuits, current mirrors, and mixers.

APPENDIX

EXAMPLE OF CIRCUIT GAIN-MAXIMIZATION PROBLEM

The main purpose of this Appendix is to demonstrate a possible approach to using the database of elementary circuits by optimizing all 3-transistor circuits for maximum possible small-signal DC voltage gain with simultaneous input power match. In addition, to ensure that the resultant topologies are realistic, we ensured that they meet the following requirements:

- Forward voltage Gain ($A_f \geq 1$).
- Reverse voltage Gain ($A_{rev} < 1/A_f$ for stable operation).
- Input Impedance (Z_{in}) to have positive real part for stable operation and input reflection coefficient to be less than -10 dB, to minimize power loss due to reflections.
- Output Impedance (Z_o) to have positive real part for stable operation.

The forward gain of a topology, A_f , is a function of transconductances of the transistors that make the topology, g_m , and the output load, R_L . To find the maximum gain, we can simply take the derivative of $A_f(g_{m_i}, R_L)$ and set it equal to zero. However, there are several constraints that limit the variables, g_{m_i} and R_L and their relations. These constraints need to be added to the problem to obtain realistic results. Each one of these constraints are described briefly below:

- The output conduction of transistor i , g_{ds_i} , is set to a typical ratio of $g_m/g_{ds} = 10$.

- The reverse gain of the transistor, A_{rev} , should always be less than the inverse of the forward gain.
- There are limits on the values of the transconductances of the transistors, g_{m_i} which are set to $0.001 \leq g_{m_i} \leq 0.1$ based on our experience with designing analog circuits. It should be mentioned that g_m of 0.1 A/V is possible but requires a relatively large transistor consuming significant amount of power and $g_m = 0.001$ A/V is possible with small, low-power transistors.
- R_{in} or the real part of the input impedance, Z_{in} has to be between 25 and 100Ω , and the load resistance is set between 1 and $10,000 \Omega$.

The mathematical model representing the above objective and constraints can be written as follows:

$$\max_{g_{m_i}, R_L} A_f(g_{m_i}, R_L) \quad (VI.1)$$

$$\text{s.t.} \quad g_{ds_i} = 0.1 \times g_{m_i} \quad (VI.2)$$

$$A_{rev} < \frac{1}{A_f} \quad (VI.3)$$

$$0.001 \leq g_{m_i} \leq 0.1 \text{ A/V} \quad i = 1, 2, 3 \quad (VI.4)$$

$$25 \leq R_{in} \leq 100 \Omega \quad (VI.5)$$

$$1 \leq R_L \leq 10,000 \Omega. \quad (VI.6)$$

This mathematical formulation in the format of (VI.1) to (VI.6) is a non-convex, non-linear mathematical programming problem. To solve these problems, techniques ranging from neural networks [36], [37] to non-convex optimization techniques [38] exist and can be employed.

REFERENCES

- [1] S. Danesh, "Technologies that could change the world-you decide," *IEEE Solid States Circuits Magazine*, vol. 4, pp. 88–90, 2012.
- [2] C.-P. Chang, J.-H. Chen, and Y.-H. Wang, "A fully integrated 5 GHz low-voltage LNA using forward body bias technology," *IEEE Microwave and Wireless Components Letters*, vol. 19, no. 3, pp. 176–178, 2009.
- [3] H.-H. Hsieh, J.-H. Wang, and L.-H. Lu, "Gain-enhancement techniques for CMOS folded cascode LNAs at low-voltage operations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 8, pp. 1807–1816, 2008.
- [4] A. Motamed, C. Hwang, and M. Ismail, "A low-voltage low-power wide-range CMOS variable gain amplifier," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, no. 7, pp. 800–811, 1998.
- [5] M. S. Steyaert, B. De Muer, P. Leroux, M. Borremans, and K. Mertens, "Low-voltage low-power CMOS-RF transceiver design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 1, pp. 281–287, 2002.
- [6] A. Doboli and R. Vemuri, "Exploration-based high-level synthesis of linear analog systems operating at low/medium frequencies," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* *IEEE Transactions on*, vol. 22, no. 11, pp. 1556–1568, Nov 2003.
- [7] M. Meissner and L. Hedrich, "FEATS: Framework for explorative analog topology synthesis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 2, pp. 213–226, Feb 2015.
- [8] E. Klumperink, "Transconductance based cmos circuits: circuit generation, classification and analysis," Ph.D. dissertation, University of Twente, 1997.
- [9] F. Brucoleri, E. Klumperink, and B. Nauta, "Generating all 2-transistor circuits leads to new wide-band CMOS LNAs," in *Proceedings of the 26rd European Solid-State Circuits Conference, 2000. ESSCIRC '00.*, Sept 2000, pp. 316–319.
- [10] E. Klumperink, F. Brucoleri, and B. Nauta, "Finding all elementary circuits exploiting transconductance," *IEEE Transactions on Circuits and Systems II*, vol. 48, no. 11, pp. 1039–1053, November 2001.
- [11] J. Schmitz, "Systematic design approach of MOSFET based low noise amplifiers for wideband RF receivers," in *2nd European Microwave Integrated Circuits Conference*, October 2007, pp. 1253–1256.

- [12] A. S. Elwakil and M. A. Murtada, "All possible canonical second-order three-impedance class-A and class-B oscillators," *Electronics Letters*, vol. 46, no. 11, pp. 748–749, May 2010.
- [13] A. S. Elwakil, "Design of non-balanced cross-coupled oscillators with no matching requirements," *IET Circuits, Devices & Systems*, vol. 4, pp. 365–373, Sept 2010.
- [14] A. S. Elwakil and B. J. Maundy, "Single transistor active filters: What is possible and what is not," *IEEE Transactions on Circuits and Systems I*, vol. 61, no. 9, pp. 2517–2524, Sept 2014.
- [15] A. S. Elwakil, B. Maundy, and M. A. Murtada, "Mapping of circuit variables into two-port network variables in basic amplifier structures: identifying new topologies," *International Journal of Circuit Theory and Applications*, vol. 42, no. 11, pp. 1203–1208, 2014.
- [16] D. Shahhosseini, M. Taghavi, L. Behjat, and L. Belostotski, "An analog-design assistant tool and an example of its application," in *IEEE Midwest Symposium on Circuits and Systems*, August 2013, pp. 233–236.
- [17] G. Alpaydin, S. Balkir, and G. Dundar, "An evolutionary approach to automatic synthesis of high-performance analog integrated circuits," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 3, pp. 240–252, June 2003.
- [18] D. Binkley, C. Hopper, S. D. Tucker, B. C. Moss, J. M. Rochelle, and D. Foty, "A CAD methodology for optimizing transistor current and sizing in analog CMOS design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 2, pp. 225–237, 2003.
- [19] S. S. Sapatnekar, V. B. Rao, P. M. Vaidya, and S.-M. Kang, "An exact solution to the transistor sizing problem for CMOS circuits using convex optimization," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 12, no. 11, pp. 1621–1634, 1993.
- [20] P. Pant, V. K. De, and A. Chatterjee, "Simultaneous power supply, threshold voltage, and transistor size optimization for low-power operation of CMOS circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 6, no. 4, pp. 538–545, 1998.
- [21] T. Sripramong and C. Toumazou, "The invention of CMOS amplifiers using genetic programming and current-flow analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 21, no. 11, pp. 1237–1252, November 2002.
- [22] T. Dastidar, P. Chakrabarti, and P. Ray, "A synthesis system for analog circuits based on evolutionary search and topological reuse," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 2, pp. 211–224, April 2005.
- [23] T. Massier, H. Graeb, and U. Schlichtmann, "The sizing rules method for CMOS and bipolar analog integrated circuit synthesis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 12, pp. 2209–2222, December 2008.
- [24] T. McConaghy, P. Palmers, M. Steyaert, and G. Gielen, "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*, vol. 28, no. 9, pp. 1281–1294, September 2009.
- [25] R. Vural and T. Yildirim, "Analog circuit sizing via swarm intelligence," *AEU-International Journal of Electronics and Communications*, vol. 66, no. 9, pp. 732–740, 2012.
- [26] B. Razavi, *Design of Analog CMOS Integrated Circuits*. McGraw-Hill, 2000.
- [27] H. Sato and S. Takagi, "Adaptively-biased MOSFET for low voltage CMOS analog circuits," *Analog Integrated Circuits and Signal Processing (Springer)*, vol. 72, no. 3, pp. 557–563, March 2012.
- [28] E. Y. Gik, *Chess and Mathematics (Shakhmaty i matematika)*. Nauka Publishers, Moscow (In Russian), 1983, no. ISBN 3-87144-987-3.
- [29] J. Riordan, *An Introduction to Combinatorial Analysis*. Princeton University Press, 1980, no. ISBN 978-0-691-02365-6.
- [30] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004, no. ISBN 978-0-521-83378-3.
- [31] D. P. Bertsekas, *Nonlinear programming*, second ed. Athena Scientific, 1999, no. ISBN 1-886529-00-0.
- [32] D. Shanno and R. Vanderbei, "Interior point methods for non convex nonlinear programming: orderings and higher order methods," *Mathematical Programming*, vol. 87, no. 23, pp. 303–316, 2000.
- [33] M. Avriel, *Nonlinear Programming: Analysis and Methods (Dover Books on Computer Science)*. Prentice-Hall, 2003, no. ISBN 0-486-43227-0.
- [34] [Online]. Available: <http://neos-guide.org/Optimization-Guide>
- [35] S. Wright, *Primal Dual Interior Point Methods*. SIAM, 1997.
- [36] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. Academic Press, 2009.
- [37] M. Laguna and R. Marti, *Scatter Search: Methodology and Implementations in C*. Kluwer Academic Publishers, Boston, 2003.
- [38] R. J. Vanderbei, *Linear Programming: Foundations and Extensions (International Series in Operations Research & Management Science)*. Springer, July 2013, no. ISBN 978-1-4614-7629-0.



Delaram Shahhosseini received the B.Sc. degree from University of Tehran, Tehran, Iran, in 2010, and the M.Sc. degree from University of Calgary, AB, Canada, in 2012, both in electrical engineering. During her M.Sc. studies, she worked on computer-aided design tools for analog circuit design. Since 2013, she has been working at Advanced Micro Devices Inc. (AMD) where she was involved in design verification at IP level and development of firmware and multimedia drivers.



Eugene Zailer (S'10) is a PhD student at the University of Calgary. He received his Bachelor of Electrical and Computer Engineering from Lakehead University in Thunder Bay, where his final year project received First Place recognition. Since 2014, Mr. Zailer has been working towards his PhD degree. His current research includes the design of a fully integrated radio-telescope receiver-chain (System on Chip) including system components such as Low-Noise Amplifiers, Voltage-Controlled Oscillators, Mixers, Variable-Gain Amplifiers, and Analog-to-Digital Converters. Mr. Zailer is an IEEE Young Professionals member and IEEE Solid-State Circuits Society member.



Laleh Behjat is a Professor in the department of Electrical and Computer Engineering, Schulich School of Engineering, University of Calgary. She joined the University of Calgary in 2002. Dr. Behjat's research focus is on developing EDA techniques for physical design and application of large-scale optimization in EDA. Her research team has won several awards including 1st and 2nd places in ISPD 2014 and ISPD 2015 High Performance Routability Driven Placement Contests and 3rd place in DAC Design Perspective Challenge in 2015. She is an Associate Editor of the IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems and Optimization in Engineering from Springer. Dr. Behjat has been developing new and innovative methods to teach Computer Science and EDA to students. She acted as an academic advisor for of Google Technical Development Guide and has won several awards for her efforts in education including 2017 Killam Graduate Student Supervision and Mentorship Award. Her team, Schulich Engineering Outreach Team, was also the recipient of the ASTech Leadership Excellence in Science and Technology Public Awareness Award in 2017. Her other interests include raising awareness about issues related to diversity and inclusion and promoting diversity in engineering. She received the Women in Engineering and Geoscience Award from APEGA in 2015 in recognition of her work in this area.



Leonid Belostotski (S'97-M'01-SM'05) received the BSc and MSc degrees in electrical engineering from the University of Alberta, Edmonton, AB, Canada, in 1997, 2000, respectively, and the PhD degree from the University of Calgary, Calgary, AB, Canada, in 2007. He was an RF Engineer with Murandi Communications Ltd., Calgary, from 2001 to 2004. He is currently a Professor with the University of Calgary and the Canada Research Chair in High-Sensitivity Radiometers and Receivers. His current research interests include RF and mixed-signal ICs,

high sensitivity receiver systems, antenna arrays, and terahertz systems. Dr. Belostotski was a recipient of the IEEE Microwave Theory and Techniques-11 Contest on Creativity and Originality in Microwave Measurements in 2008 and the Outstanding Student Designer Award from Analog Devices, Inc., in 2007. He is an Associate Editor of IEEE Transactions on Instrumentation and Measurement and the IEEE Southern Alberta Solid-State Circuits and Circuits and Systems Chapter's Chair.