



Modeling in Analog/Mixed-Signal Simulation, Sizing and Layout

Florin Burcea

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:

Prof. Dr.-Ing. Georg Sigl

Prüfende der Dissertation:

1. apl. Prof. Dr.-Ing. habil. Helmut Gräß
2. Assoc. Prof. Dani Tannir, Ph.D.

Die Dissertation wurde am 25.05.2020 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 05.10.2020 angenommen.

Abstract

The complexity of integrated circuits and systems has increased significantly over the last decades. To reflect the behavior of the circuit or systems as accurately as possible through simulation, the complexity of the models of devices, circuits and systems has increased alike. However, the ever higher model complexity has led to a simultaneous increase of the computational cost for simulation. To prevent the simulation cost from becoming prohibitive, efficient models are needed that find a good trade-off between accuracy and complexity. This thesis presents the problem-specific computational modeling with a problem-specific trade-off between accuracy and what behavior has to be modeled on the one side and the computational cost for simulation on the other side for three application areas of electronic circuits and systems, each with specific modeling requirements and each corresponding to a different stage in the design process.

The first contribution is for the simulation of switched-mode DC-DC converters. The transistor-level switching part is replaced by an averaged model over a switching period. Then, a new multi-harmonic model is presented, that reconstructs the transient waveforms using the averaged waveforms, the small-signal information and the Fourier series of the known waveforms at the terminals of the switching cell. The new model is less complex than the full transistor-level representation but with a similar accuracy.

The second contribution is in the simulation and co-optimization of combined systems with MEMS and integrated circuits. For the MEMS part, a reduced-order model is used, less complex than the compact model based on finite element analysis, but with enough accuracy for circuit-level simulation. The reduced-order MEMS model is dynamically and automatically generated during optimization. The MEMS and the integrated circuit are co-optimized, considering simultaneously design and process parameters from both the mechanical and the electrical side.

The third contribution targets the layout of the capacitors of a charge-redistribution ADC. An analytical model is presented that expresses the static integral and differential nonlinearity (INL and DNL) of a charge-redistribution ADC with generalized capacitor ratios as linearized functions of capacitor variations. The model of capacitor variation includes a statistical (caused by local, random mismatch), a systematic (caused by process gradients across the array) and a parasitic (caused especially by routing) component. The linearized model of INL and DNL, combined with the model for capacitor mismatch, conducts a placement and routing methodology with the purpose of minimizing INL and DNL.

Acknowledgements

I would like to begin by expressing my sincere and deep appreciation to my advisors and my collaborators, without whom the results obtained in this work wouldn't have been possible.

First and foremost, I would like to thank Prof. Helmut Gräß for his dedication and patience in supervising this doctoral project, for his continuous availability, for his continuous encouragement and appreciation of my work, for reading and reviewing my papers, for his support and contribution in conceiving and organizing the conference presentations and for teaching me how to emphasize, in papers and in presentations, the contributions of the work.

I would like to thank Prof. Dani Tannir for our excellent collaboration that culminated with the publication "Fast SPICE-Compatible Simulation of Low-Power On-Chip PWM DC-DC Converters with Improved Ripple Accuracy".

I would like to thank Andreas Herrmann for his support in the concept and development of the scripted flow of MEMS-IC co-optimization, for his contribution to the publication "MEMS-IC Robustness Optimization Considering Electrical and Mechanical Design and Process Parameters".

I would like to thank Ye Ding for the excellent work done in his Master thesis, that lead to the publication "PASTEL: Parasitic Matching-Driven Placement and Routing of Capacitor Arrays with Generalized Ratios in Charge-Redistribution SAR-ADCs", and for his contribution to this paper. I would like to thank Dr. Husni Habal for initiating and coordinating this Master thesis within Infineon and for his contribution to the paper.

Finally, I would like to thank Maximilian Neuner for his continuous technical support in managing the infrastructure of the simulation and optimization tools.

Contents

Abstract	iii
Acknowledgements	v
Contents	vii
List of Figures	ix
List of Tables	xi
1 Introduction	1
1.1 Analog vs. Digital Circuit Design	1
1.2 Analog Design Flow	2
1.3 Thesis Content	3
1.4 Modeling for Simulation of Integrated DC-DC converters	4
1.5 Modeling for MEMS-IC Sizing	10
1.6 Modeling for Capacitor Array Layout in Charge-Redistribution ADCs	15
2 Modeling for Simulation of Integrated DC-DC Converters	23
3 Modeling for MEMS-IC Sizing	39
4 Modeling for Capacitor Array Layout in Charge-Redistribution ADCs	69
5 Conclusion & Outlook	87
Bibliography	89

List of Figures

1.1	Analog circuit design flow.	3
1.2	Synchronous Buck converter with peak current control. The circuit part integrated on chip and the PWM switch cell are shown distinctively. On the right side, waveforms at the PWM switch cell terminals.	5
1.3	Output voltage (V_{out}) ripple (total voltage minus average value) for a Buck converter. Reconstructed waveform, from the averaged-model simulation, with 25 harmonics, and the waveform resulted from full transistor-level simulation.	8
1.4	Simulation flow based on PWM switch cell averaged model and Fourier series.	9
1.5	a) Mechanical structure and electrical behavior of a MEMS microphone. b) Readout circuit, in which the MEMS is represented by its electrical behavior.	11
1.6	MEMS-IC optimization flow.	12
1.7	Circuit sizing flow.	14
1.8	MEMS accelerometer: a) 3D model b) basic electrical behavior	15
1.9	A schematic of a charge-redistribution SAR-ADC, with capacitive variations included: top plate-to-substrate parasitic (blue), and individual capacitor variation (red).	16
1.10	Final placement for the 12-bit architecture with generalized ratios characterized by Eq. (1.1), also depicting routing.	21

List of Tables

1.1 Accelerometer performances robustness before and after yield optimization.	15
--	----

1 Introduction

Integrated circuit design is a fundamental part of the electronics industry. Moore's law stated that the number of transistors in an integrated circuit would double every 18 to 24 months [1]. This prediction has become a target for the microelectronics industry, driving it for the past five decades [2]. This growth rate has been made possible not only by the technology feature miniaturization, but also by the electronic design automation (EDA) development.

Recently, technology scaling has slowed down because of approaching physical limits. Also, at technology nodes in the range of nanometers, the process does no longer bring a cost benefit. To improve cost/performance while adding functionality, the industry is looking for new solutions. In the so-called "more than Moore" trend, devices and functionalities of different domains (e.g. MEMS devices, RF devices, power devices), which do not scale according to Moore's law, are integrated on the chip.

To keep up with the increasing complexity and with the strict time-to-market requirements, there is a focus on design productivity improvement, in which circuit design automation is a fundamental factor. In integrated systems, the analog part requires a bigger design effort than the digital part, because it is less supported by automation. The main reason for the lack of automation is the complexity of analog circuit design. Also, the application of optimization methods is computationally prohibitive, unless fast enough simulation models and methods are available. Therefore, the industrial design of analog circuits is still predominantly manual.

1.1 Analog vs. Digital Circuit Design

The design automation of digital circuits has been very successful in the last decades. In digital circuits, the main variable parameter which is not discrete in nature is the time needed for the transition from one logic state to the next, also called delay. If this time can be predicted and if the delay of a large block can be expressed as the sum of the delays of the individual cascaded sub-blocks, the process of building a large digital circuit can be split into two parts:

1. meeting the correct logic functionality of the circuit
2. meeting the overall system timing

The delays are subject to process variability, local and global, such that the delays of each library element have their individual best- and worst-case timing. However, even with the statistical process effects, the total delay is still the sum of the individual delays of the building blocks and therefore mostly a linear relationship. The overall margins

1 Introduction

of a design are determined by the individual delays of the library elements and their margins.

The parametric performance, given by the timings margins, is closely tied to technology. Design and process are linked through the SPICE-based transistor models, in the sense that the SPICE-based simulation model translates technology into circuit parameters. Similarly, the technology spread is translated into timing spread in the digital library elements.

The topology and structure of a digital design is furthermore coded in a technology-independent language. In this way, circuit design and technology development can run in parallel.

In contrast, the design of analog circuits is a multi-dimensional problem, it is closely tied to technology, and there is no simple model to abstract analog circuits from technology.

In analog circuits, there are many variables involved. These include, among others, the transistor geometry (width and length), the threshold voltage, the mobility, resistor and capacitor values, mismatch parameters and noise parameters. Many (or even all) technology parameters and design parameters affect each circuit parameter. In turn, many (or even all) circuit parameters affect each circuit performance. Moreover, these attributes are affected by aging, i.e. they change over time as a result of degradation mechanisms, which may cause the circuit to fail to meet the performance specifications at some point during its intended lifetime [3, 4]. Although the previously mentioned variables and dependencies are found also in the digital design, the analog design has more performance features to be traded off. Therefore, these complex interactions and dependencies are more difficult to understand and to correlate to the yield, in order to optimize the yield.

Overall, analog circuits require significantly more design effort than digital circuits in relation to the size of a circuit. The high complexity of the analog design task is the reason behind the low level of automation. In practice, the design of analog circuits is still predominantly manual.

1.2 Analog Design Flow

The design of an analog circuit begins with the definition of the circuit behavior. The circuit is required to fulfill several specifications, which consist of minimum and/or maximum acceptable values of performances (e.g., the minimum gain and/or bandwidth of an amplifier, the maximum power consumption of the circuit), under some given environmental conditions (e.g., a temperature range and a supply voltage range). Also, the design has to be carried out in a given technology.

Then, a circuit topology is selected. This selection depends on the design inputs (technology, specifications, temperature). For example, an operational amplifier requiring a gain of at least 70dB needs either a cascode structure or at least two gain stages. If the circuit is supplied from a low voltage, the cascode option is not suitable, as this requires a higher supply voltage. Instead, several gain stages are cascaded. The disadvantage is

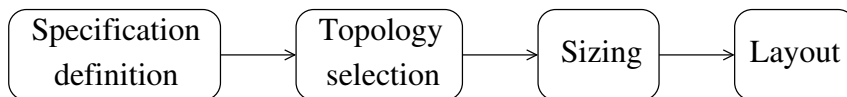


Figure 1.1: Analog circuit design flow.

that this topology requires frequency compensation. Also, several approaches to support the automation of topology generation have been proposed [5, 6, 7, 8].

After selecting the topology, the devices have to be sized. This consists of selecting the appropriate values for the parameters describing the geometry of transistors, resistors and capacitors, e.g. transistor widths and lengths. The parameter values must be appropriate in the sense that the circuit must fulfill the specifications under all operating conditions with maximum probability for the statistical manufacturing process.

The sizing process, whether manual or automated, is strongly dependent on computer simulation. To reflect the circuit behavior with good precision (to accurately calculate the values of the performances), very complex device models are used in the simulation. However, this leads to long simulation time. Moreover, the sizing process needs a large number of simulations. Therefore, efficient models with a good tradeoff between model accuracy/complexity and simulation time are needed.

The design process continues with the layout of the circuit. This step has two components: placement and routing. Simulation is typically required also after the layout (post-layout simulation), to consider the impact of the parasitic elements. This requires complex parasitic extraction tools. The layout process produces the mask data for manufacturing the circuit.

The flow is illustrated in Fig. 1.1.

1.3 Thesis Content

The subject of this thesis is the problem-specific computational modeling, for several stages in the design process, with a problem-specific trade-off between accuracy and what behavior has to be modeled on the one side and the computational cost for simulation on the other side, to make different design tasks more efficient. Three contributions to problem-specific computational modeling in three application areas of electronic circuits and systems, each with specific modeling requirements, are presented in this work.

The first contribution is to the simulation of switched-mode DC-DC converters. In switching circuits, the simulation time is typically very long because of the numerous switching events. Also, small-signal analysis (e.g., for evaluating the stability of control loops) cannot be directly applied. To address these issues, the transistor-level switching part is replaced by a model that averages the behavior of the waveforms over a switching period. As the averaged model alone does not consider the transient behavior within a switching period, this work introduces a new multi-harmonic model that reconstructs the transient waveforms using the averaged waveforms, the small-signal information and the known waveforms at the terminals of the switching cell. The rest of the circuit, apart from the switching cell, is represented at transistor level in the simulation.

1 Introduction

The second contribution is to the simulation and optimization of combined systems with MEMS and integrated circuits. Modeling the mechanical component (the MEMS) by finite-element analysis (FEA) provides good accuracy, but raises a prohibitive computational cost for the electrical co-simulation (in a SPICE-like simulator) of the MEMS and the integrated circuit. The complexity difference from the FEA level to the SPICE (analog) level is comparable with the difference from the analog level to the digital level. At the same time, basic models of the MEMS provide insufficient accuracy. The usage of compact modeling for the MEMS makes MEMS-IC co-simulation both feasible with respect to computational time and accurate enough. This work applies compact modeling in combination with model order reduction, which further speeds up the simulation while little affecting the accuracy. In this case study, the sizing of the combined MEMS-IC is done automatically. For this, methods for robustness analysis and optimization developed for integrated circuits are applied to the extended MEMS-IC case. Furthermore, a new modeling-and-simulation-in-a-loop flow is introduced to generate the reduced-order compact model of the MEMS part for the simulation.

The third contribution refers to successive-approximation charge-redistribution analog-to-digital converters (ADCs). To obtain the static transfer curve of an ADC (output binary code vs. input voltage) by transistor-level simulation, a long transient simulation is necessary, as many points are needed to accurately detect the transition voltages between consecutive output codes. This work presents a simple and efficient analytical model of the static integral and differential nonlinearity (INL and DNL) with respect to the capacitor values (which include variations, global and local, caused by the manufacturing process, and parasitic capacitances caused especially by routing), considering that capacitor non-idealities are the main source of mismatch. This model is used to conduct an optimal placement and routing strategy for the capacitor array that minimizes the static nonlinearity of the ADC.

The three contributions are detailed in the following three sections.

1.4 Modeling for Simulation of Integrated DC-DC converters

Switching circuits are characterized by their transient behavior over a switching period. In contrast to linear circuits, they do not have a DC operating point. Therefore, transient simulation is the only option. The first consequence is that the simulation of switching circuits is generally very time-consuming because of the co-existence of fast switching activity and slow load variation. The second consequence is that small-signal analysis cannot be directly applied (this analysis is essential for, e.g., the stability evaluation of the feedback loop).

Switched-mode DC-DC converters, like Buck and Boost converters, fall into this category. Fig. 1.2 shows a Buck converter with peak current control. A solution to address the two previously mentioned challenges consists of modeling techniques for circuit averaging. These models average the circuit behavior over a switching period and focus on the cycle-to-cycle dynamics, described by a smoothly varying, continuous signal. This implies the disappearance of switching events and hence discontinuities [9].

1.4 Modeling for Simulation of Integrated DC-DC converters

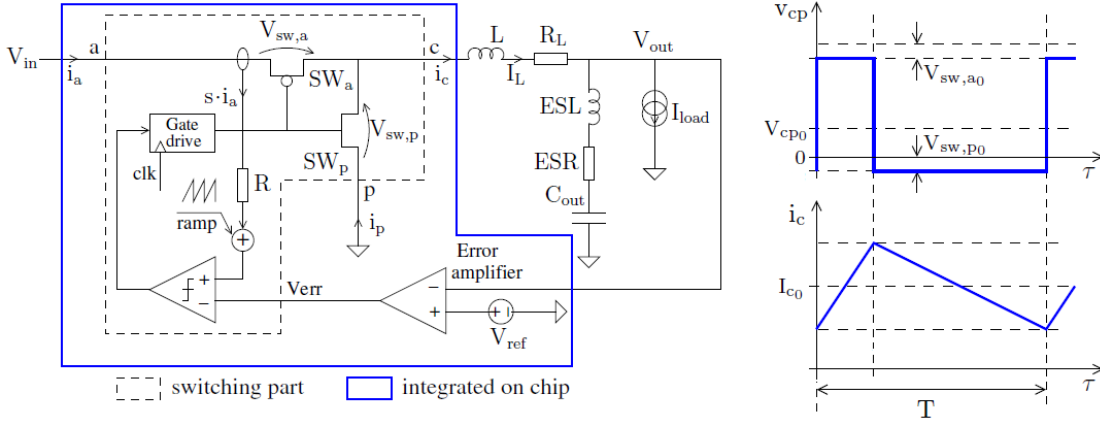


Figure 1.2: Synchronous Buck converter with peak current control. The circuit part integrated on chip and the PWM switch cell are shown distinctively. On the right side, waveforms at the PWM switch cell terminals.

Averaged models have two important advantages. First, simulation results are very fast, as a result of the absence of switching. In this way, long transient effects of low-bandwidth systems can be easily visualized. Second, the small-signal response is obtained immediately. Properties like the loop gain, the phase margin (for stability evaluation of the feedback loop) or the input or output impedance can then be easily assessed. On the contrary, it is very difficult to obtain a small-signal response from a transient simulation.

However, averaged models also suffer from limitations. First, parasitic elements (parasitic inductances or capacitances, leakage currents, forward voltage drops on the switches) are difficult to include. Although advanced models include some parasitic elements, the modeling process is tedious. Second, switching losses are not captured. Third, simulation does not reveal the waveform ripples. Transient models, on the other hand, reflect reality much more precisely, as long as the component models are accurate enough. In this way, ripples, nonlinearity, switching losses, spikes, conduction losses, leakage and other parasitic effects are easily assessed by means of the transistor-level transient simulation.

A very popular circuit averaging technique for DC-DC converters is state-space averaging (SSA). It allows the derivation of small-signal models of DC-DC converters. In [10, 11, 12], large-signal and small-signal SSA models for DC-DC converters operating in continuous conduction mode (CCM) and discontinuous conduction mode (DCM) are presented. However, the models described in [11] and [12] and SSA models in general result from circuit-specific approaches and they are therefore limited with regards to the capability to automate the modeling and simulation process on arbitrary DC-DC converter topologies. Also, the derivation is complicated, involving a large number of state variables and a large number of circuit components.

Alternatively, the pulse-width-modulation (PWM) switch cell model, which is a linearization of switch-cell, can be readily applied to a wide range of DC-DC converter topologies and greatly simplifies the small-signal analysis. This technique does not re-

1 Introduction

quire an averaging or linearization process of the entire converter, but only of the PWM switch cell (enclosed within the dashed box in Fig. 1.2). Additionally, this model fits all topologies by simple model rotations.

The model was first introduced in [13, 14] in its basic form. In [15], the basic PWM switch model was further enhanced by continuously tracking the circuit operating mode in order to allow the simulation of DC-DC converters operating in both CCM and discontinuous conduction mode DCM. In [9], comprehensive and enhanced circuit-averaged models to be adopted in circuit simulators are developed and presented covering a wide range of converter topologies, including voltage mode and current mode controlled converters. In [16] it is shown how to efficiently and accurately account for several non-ideal effects such as subthreshold leakage current and short-channel effects using the averaged model.

The PWM switch cell model is invariant, simple to apply and computationally efficient. However, its major limitation is that it does not consider the transient behavior of the DC-DC converter within one switching period. For example, it provides no information about the ripple of the waveforms. As suggested in [17], the PWM switch cell model has an underlying assumption of a small ripple condition, which prevents it from being applied to converters with large ripples. It has been shown that neglecting the ripples of state variables can lead to large discrepancies in the simulation results of converters operating at low frequencies [18].

To account for the ripples, a lot of work has been done on generalized averaging techniques [19]. In [18], a multi-frequency averaged model is introduced which conducts frequency-selective averaging on switch state-space models of DC-DC converters. However, this method is based on Boost converter types and cannot be easily applied to other converter configurations. In [20], a flexible method of in-place averaging that replaces elements in DC-DC converters with the k^{th} index equivalent averaged elements is presented, thereby allowing the tracking of responses with harmonics up to k^{th} degree. However, the method in [20] assumes ideal switches that have no nonlinear characteristics and no parasitic effects. As a result, such a model becomes neither suitable nor accurate enough for low-power DC-DC converter simulations, where the ideal switch assumptions no longer hold.

In [21, 22] a complete multi-harmonic solution to low-power PWM DC-DC converter simulations was presented. The presented solution combines the accuracy of enhanced state-space averaging [12] with the flexibility of PWM switch cell models and the multi-frequency nature of the generalized models. The multi-frequency model approximates the actual converter responses by the average value with multiple harmonics of ripples of arbitrary degree while accounting for the effects of device nonlinearities. It is also general in the sense that it can be applied to many DC-DC converters including Buck, Boost and Buck-Boost converters without any modifications and in the sense that both continuous conduction mode and discontinuous conduction mode operations are supported. However, as it is not uncommon that the spectrum of the waveforms of DC-DC converters contains high-frequency components and the ripple of these waveforms has a non-conventional aspect, the model can only reflect the waveforms accurately by using a large number of harmonics. In that case, the developed model would become extremely

complex and highly coupled, which makes it computationally very expensive to use. An approach that was proposed to address this problem consists of analytical Fourier series based methods [23, 24, 25, 26, 27, 28], which account for a large number of harmonics to capture the ripple of the output waveforms with a good accuracy. The limitation of these methods is that they have to be derived separately for each converter topology, because they are based on circuit-specific equations that are analytically expanded by means of Fourier series.

The model proposed in [29] is also based on Fourier series and focuses on capturing small-signal properties and transfer functions. In this way, the model is general and not circuit-specific, going a step forward compared to previous approaches. Both in [29] and [30], such small-signal models, being very useful for designing appropriate feedback control, are applied to multi-phase converter design. However, the limitation of these models is that they do not describe the large-signal behavior.

Circuit-specific methods are also limited by the fact that adding new circuit elements requires new modeling effort. An illustrative example is given by the parasitic elements of the output capacitor. In [18], an ideal capacitor is considered, while in [22] the capacitor has a parasitic equivalent series resistance (ESR). No previous work includes the equivalent series inductance (ESL). Ignoring this parasitic element decreases the accuracy of the output voltage waveform. However, adding it to the model would significantly increase the model complexity. In Fig. 1.2, the output capacitor and the inductor are shown together with their parasitic elements.

This work presents a new multi-harmonic model. The model is based on simulation and accurately captures the ripple in the output waveforms in conjunction with circuit averaging techniques. In the proposed model, the coefficients of the Fourier series are calculated from the simulated averaged waveforms, obtained in turn through circuit averaging techniques. The Fourier series coefficients are then used to reconstruct the ripple of the output waveforms. The model is easily applicable to a wide variety of DC-DC converter topologies. The reason for this generality and flexibility is that the model is derived from the common properties of DC-DC converters (instead of being derived from equations specific to one topology): the voltage and current waveforms at the PWM switch cell terminals, as the waveforms shown in Fig. 1.2. In particular, the model can be applied to low-power converters with current-mode control (a Buck converter with such a control loop is shown in Fig. 1.2). Although this topology is popular in integrated circuits, most of the previously discussed models from the literature do not consider it. The presented model efficiently accounts for a large number of harmonics, achieving a very good accuracy with a significant simulation speedup.

The model complexity in the new approach presented in this work is higher than for averaged models and lower than for full transistor-level simulation. The state-of-the-art averaged model is included in the method, serving for the fast computation of the averaged waveforms by transient simulation and for the small-signal analysis. The method addresses the limitation of the averaged model, which does not provide information about the waveform behavior within a switching cycle. With a novel approach for ripple reconstruction, the accuracy of the final waveforms is close to the accuracy of full transistor-level simulation. However, the overall computation time is reduced by

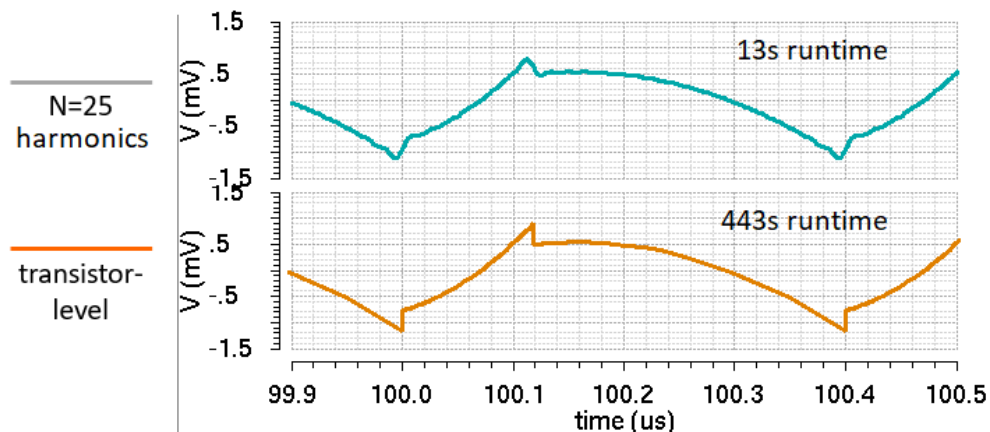


Figure 1.3: Output voltage (V_{out}) ripple (total voltage minus average value) for a Buck converter. Reconstructed waveform, from the averaged-model simulation, with 25 harmonics, and the waveform resulted from full transistor-level simulation.

more than one order of magnitude compared to full transistor-level simulation. Fig. 1.3 compares, for the Buck converter represented in Fig. 1.2, the simulated output voltage waveform for two methods:

- averaged-model simulation and ripple reconstruction with 25 harmonics,
- full transistor-level simulation.

The figure shows that the proposed method provides a waveform that resembles the full transistor-level waveform with a good accuracy and a speedup of about 35x.

Other simulation methods have also proven their efficiency in the transient analysis of switched-mode DC-DC converters. The commercial tool SIMetrix/Simplis [31] is such an example. However, the simulation approach incorporated in this tool is applicable in the system-level analysis of switching converters realized with discrete components, while in the transistor-level simulation of low-power integrated switching converters, which is the target application of the present work, other tools are commonly used. The simulation level targeted here requires simulation tools (e.g., Spectre) optimized to work with complex transistor models (e.g., BSIM). This method is implemented to work in connection with the framework of the commercial tool Cadence Virtuoso. For the Buck converter illustrated in Fig. 1.2, the components enclosed within the solid-line box are integrated on chip.

The flow diagram of the proposed approach is shown in Fig. 1.4. First, a testbench is created. The test circuits proposed in this work for Buck and Boost converters with current mode control (CMC) operating in continuous conduction mode (CCM) allow for the dynamic characterization of the parameters of the PWM switch cell model. Then, the transient analysis is run with the averaged model in order to provide the index-0 waveforms. The coefficients of the Fourier series expansions for the currents and voltages at the PWM switch cell terminals are calculated. These currents and

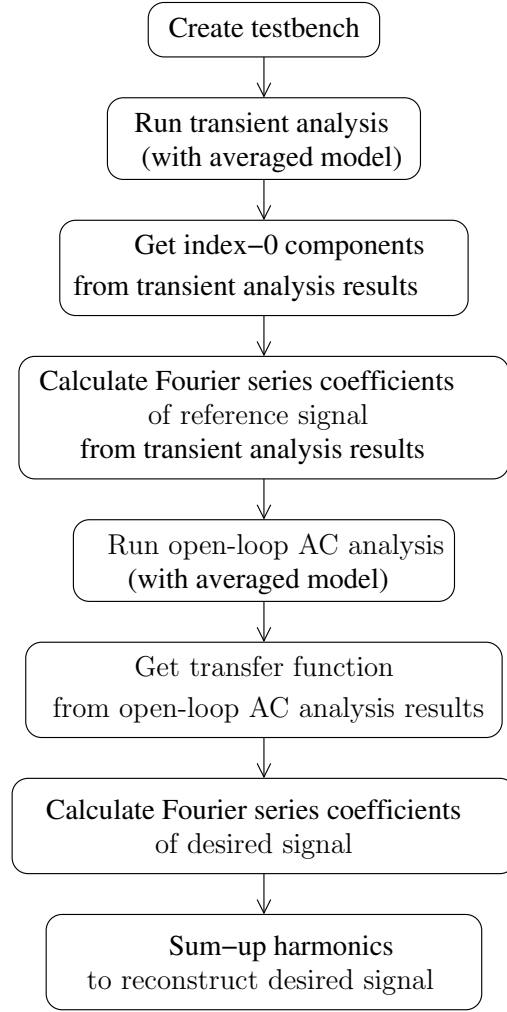


Figure 1.4: Simulation flow based on PWM switch cell averaged model and Fourier series.

voltages serve as reference signals, having well-known expressions. In this work, the expressions of the Fourier series coefficients are calculated. The numerical values of the coefficients are calculated from the averaged waveform values. The steady-state values of all the waveforms in the circuit serve as a DC operating point. In this way, the small-signal analysis can be run with the averaged model, providing the transfer function of every signal in the circuit with respect to the reference signals from the PWM switch cell terminals. Having the Fourier series coefficients of the reference signals and the transfer functions with respect to the reference signals, the Fourier series coefficients of the desired signals are calculated. Finally, the harmonics are summed to reconstruct the desired signals, for the desired number of harmonics.

The distinguishing advantage of the presented method is generality. While most of the other similar approaches from the literature are developed for one specific circuit and

1 Introduction

one specific number of harmonics, this method covers, without any additional modeling effort, a wide variety of DC-DC converter topologies, with flexibility regarding the circuit elements and with any number of harmonics. Specifically, the method offers the following benefits:

- It covers any DC-DC converter topology with current mode control. The applicability range is the same for this method as for the PWM switch cell model. The method can be easily transferred to architectures with other duty cycle control methods (e.g., voltage mode), by simply replacing the PWM switch cell model accordingly.
- It accounts for any parasitic element (e.g., the equivalent series inductance of the output capacitor, which is not included in any other modeling paper).
- It accounts for any transistor-level surrounding circuitry (e.g., error amplifier, current sensing), while all other modeling papers use ideal circuitry.
- By increasing the number of harmonics, increasing levels of precision can be achieved (the choice is determined by the trade-off between accuracy and computational effort/time).

1.5 Modeling for MEMS-IC Sizing

MEMS-IC systems consist of a mechanical part (the MEMS) and an electrical part (the integrated circuit), each belonging to a different energy domain with specific modeling requirements. For integrated circuits, SPICE-based simulation is typically used. The modeling of the MEMS is the issue discussed in this work. Fig. 1.5 shows the mechanical structure and the readout circuit for a MEMS microphone. The mechanical component from [32] is used, while the readout circuit is, as typically, based on an amplifier.

Conventionally, MEMS devices are modeled using finite element analysis (FEA). FEA is the standard method used for the simulation of mechanical components and is very precise. Commercial tools for MEMS design like ANSYS are based on this method. However, the very high modeling complexity of FEA makes the computational cost of co-simulation prohibitive. For this reason, the mechanical part and the electrical part of a MEMS-IC system are commonly designed separately.

However, the IC design depends on the interaction with the MEMS component. At least an approximate electrical behavior of the MEMS is needed in the IC design process, with a low enough complexity such that the computational cost is feasible. In some cases, a basic model is used, made up of one or a few simple circuit elements (e.g., a linearly variable capacitance for a microphone [33], as illustrated in Fig. 1.5, or two linearly variable capacitances for an accelerometer [34, 35]) and capturing the principle electrical behavior of the MEMS. In other cases, analytical modeling is used. For example, the behavior of an accelerometer is described by second-order differential equations [36, 37]. For co-simulation purposes, the analytical model is typically implemented in a

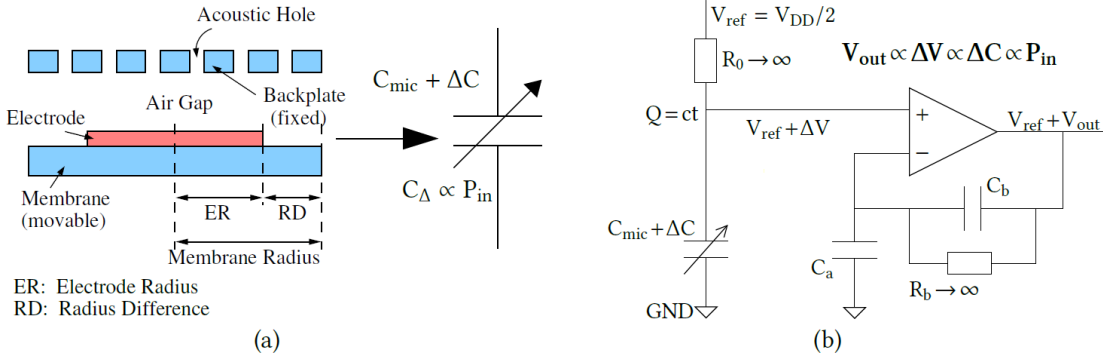


Figure 1.5: a) Mechanical structure and electrical behavior of a MEMS microphone.
b) Readout circuit, in which the MEMS is represented by its electrical behavior.

hardware description language (HDL) like Verilog-A. These two approaches have several advantages:

- They are very suitable for understanding the principle behavior of the MEMS and can be a good starting point in a design.
- They provide fast electrical simulation capabilities.
- They are easy to integrate into an electrical simulator.

These two approaches have two main disadvantages:

- They provide only limited accuracy, both physical (by not capturing all features of the electrical behavior, e.g. the high-frequency resonance) and numerical.
- The method is not general, it is not automated, since each type of MEMS device has its own equivalent circuit element(s)/its own differential equations.

On the contrary, FEA modeling is both automated and very precise. Precision comes though with the disadvantage of a prohibitive co-simulation computational cost. However, the precision provided by FEA is not necessarily needed at circuit level. A solution in this sense is the compact modeling method used in the commercial tool Coventor MEMS+ [38, 39, 32]. The compact model is still based on FEA, but computationally much cheaper. In MEMS+, devices are assembled out of the elements of a library of parametric building blocks. Compared to conventional FEA tools like ANSYS, MEMS+ assembles the models of the individual components into a single equation system. The resulting compact model has a much smaller number of degrees of freedom. For example, the MEMS+ compact model for the microphone from Fig. 1.5 has 180 degrees of freedom, while conventional FEA models typically have around 100k [40].

This complexity of the compact model is significantly lower than of the conventional FEA. However, the MEMS-IC co-simulation still remains challenging. To address the need for fast electrical simulation while preserving the critical nonlinear dynamic behavior of the MEMS part, the tool MEMS+ provides a reduced-order version (a low-order

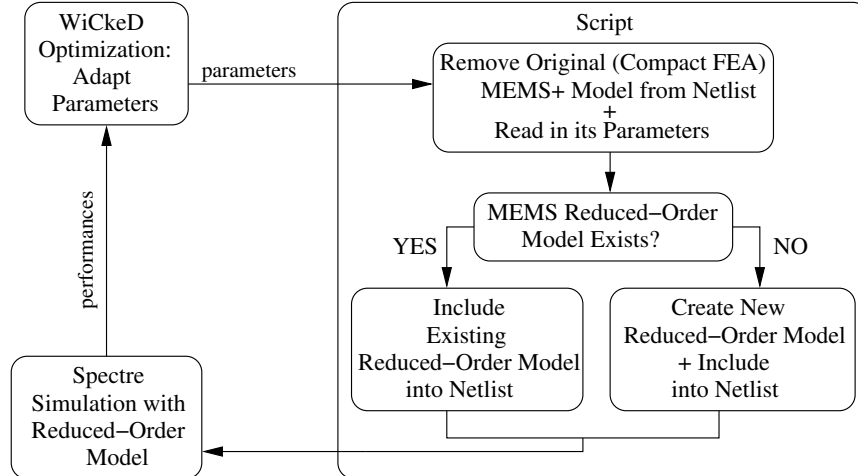


Figure 1.6: MEMS-IC optimization flow.

polynomial version) of the compact model [38, 39, 32]. In MEMS+ the model order reduction is based on a mechanical eigen-decomposition of the system, for which selected modes of interest can be preserved. With model order reduction, the mechanical degrees of freedom of the compact MEMS+ model are replaced by reduced degrees of freedom corresponding to preserved modes. For example, model order reduction decreases the number of degrees of freedom from 180 to 3 for the MEMS microphone from Fig. 1.5. The modes represent the intrinsic modal frequencies of the mechanical component. The simulation cost of the MEMS is now reduced to a level comparable with the simulation cost of the integrated circuit. For example, the runtime of a transient analysis for the MEMS-IC from Fig. 1.5 is about 20 times smaller than with the compact MEMS+ model.

The reduced-order model is exported from MEMS+ and described in the hardware description language Verilog-A. In this way, it is also easy to integrate into the circuit simulation flow. The limitation of this model, as provided by the tool, is that it is not parameterized. The exported model corresponds to a unique set of parameters (e.g., geometry parameters, temperature). Changing the parameters (e.g., for the MEMS microphone from Fig. 1.5, changing the electrode or the membrane radius, or the reference voltage, or the global temperature) requires exporting a new reduced-order model. Consequently, this model is not straightforward to integrate into an automatic sizing flow.

This limitation is addressed in the current work, where a scripting flow for MEMS-IC co-optimization was developed, based on the following tools:

- Coventor MEMS+ for MEMS compact and reduced-order modeling,
- Cadence Spectre for simulation,
- MunEDA WiCkeD for optimization.

The principle of this modeling-and-simulation-in-a-loop flow is depicted in Fig. 1.6. In every step of the optimization process, the electrical and mechanical parameters are

changed according to the optimization algorithm. This requires updating the reduced-order MEMS model for every new set of parameters. The task is accomplished by means of a script which is executed before calling Spectre for circuit simulation. This script removes the path to the compact FEA MEMS+ model of the original netlist and reads in all mechanical parameters. Next, it checks if a reduced-order model already exists with these parameters. In case it does not exist, a reduced model is created and included in the new netlist, otherwise the existing reduced model is included in the netlist. Spectre then runs with the reduced-order model instead of the compact FEA model. This approach saves CPU time by using a reduced-order model for the MEMS part.

In the combined MEMS-IC optimization, design parameters from both the electrical and the mechanical domain are adjusted simultaneously. Previous work has only considered nominal optimization [34, 35, 41]. This work presents in addition nominal and yield optimization of the MEMS-IC in consideration of operating and process parameters, including mechanical process parameters, which strongly influence the robustness. A successful optimization requires the simultaneous inclusion of design parameters and process tolerances from both energy domains.

The general sizing flow is described in the following.

The goal of analog circuit sizing is to find a set of design parameters for which the given performance specifications are fulfilled for the whole range of the operating parameters and with high probability (the yield) in consideration of process variations, described by the statistical parameters. The flow is illustrated in Fig. 1.7.

In the first stage, an initial sizing of the devices is needed. Automated approaches have been proposed for this [42, 43, 44, 45]. The resulting circuit will be called the initial design.

After the initial sizing, the circuit is optimized in three steps, using a CAD sizing tool:

1. Feasibility optimization
2. Nominal optimization
3. Yield optimization

First, the feasibility optimization ensures that the design constraints are fulfilled. Although the target of a design is to fulfill the performance specifications, their exclusive consideration in circuit sizing does not necessarily lead to technically meaningful results. The feasibility optimization starts from the initial design and searches for a design point, namely for a set of design parameters for which all constraints are fulfilled. At this stage, only nominal process conditions are taken into account. The optimization is done at worst-case operating conditions, to ensure the fulfillment of constraints for the entire operating range.

The feasible design fulfills all constraints, but can violate one or more of the performance specifications. The circuit is optimized to also fulfill the specifications (in addition to the constraints) in the nominal optimization step. The nominal optimization starts from a feasible design and searches for the set of design parameters for which all performance specifications and constraints are fulfilled. Like in the feasibility, nominal process conditions and worst-case operating conditions are considered.

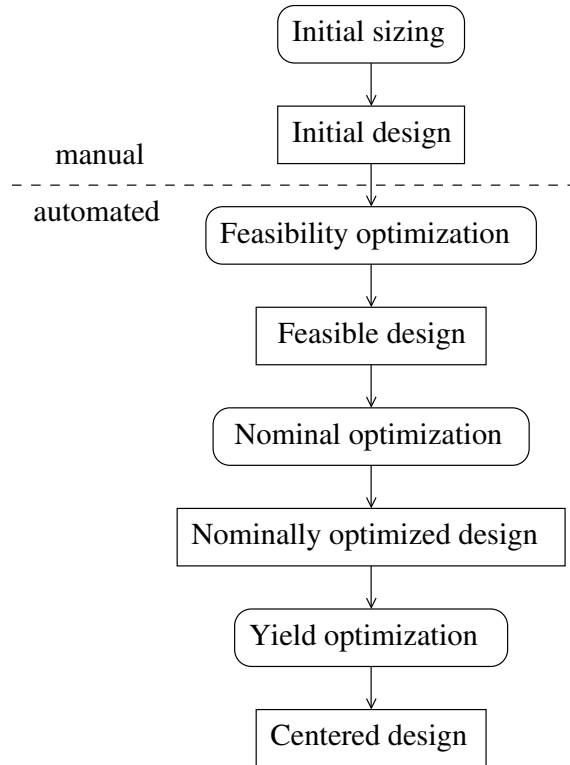


Figure 1.7: Circuit sizing flow.

A nominally optimized design fulfills the specifications for the entire operating range at nominal process conditions, but does not necessarily fulfill all specifications under consideration of process variations. The yield of the circuit is defined as the probability that the circuit fulfills all specifications under all operating conditions. The yield represents the percentage of circuits satisfying all specifications. The yield optimization starts from the nominally optimized design and searches for the set of design parameters for which the maximum yield is obtained. This final step is also called design centering.

The MEMS accelerometer (depicted in Fig. 1.8 with its 3D view and its basic electrical behavior) with readout circuit presented in this work is an illustrative example of MEMS-IC co-optimization. It is a complete example, in the sense that it includes all types of parameters: design and process parameters from both the mechanical and the electrical side and process parameters. The yield optimization succeeds when considering all parameter types.

In the nominally optimized design, all performances fulfill the specification at nominal process conditions. However, at worst-case process conditions, corresponding to a 3-sigma tolerance, 2 out of 4 performances violate the specification. One of these 2 performances is the resonance frequency, which depends only on the mechanical (design and process) parameters. The low robustness of this performance can only be detected when mechanical process parameters are included in the analysis.

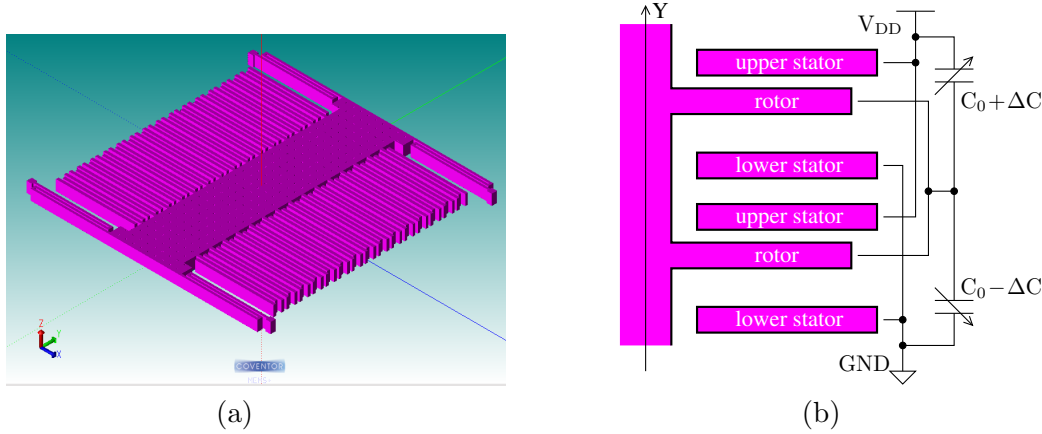


Figure 1.8: MEMS accelerometer: a) 3D model b) basic electrical behavior

Table 1.1: Accelerometer performances robustness before and after yield optimization.

Performance	Robustness/yield (worst-case operation)			
	Before yield optimization	After yield optimization		
		electrical+mechanical	only electrical	only mechanical
Resonance frequency	1.0 σ /84%	5.0 σ /100%	1.0 σ /84%	3.2 σ /84%
Gain	5.6 σ /100%	5.7 σ /100%	9.5 σ /100%	1.5 σ /93%
Total harmonic distortion	3.0 σ /99.7%	4.3 σ /100%	2.5 σ /99.4%	5.3 σ /100%
Opamp Phase Margin	0.6 σ /73%	6.4 σ /100%	7.0 σ /100%	1.0 σ /84%

The yield optimization adjusts the design parameters such that in the optimized design all performances fulfill the specification at nominal and worst-case process conditions. The optimization succeeds only when including simultaneously both the mechanical and the electrical design parameters. The optimization fails when the design parameters are considered only from one domain. In those cases, 2 out of the 4 performances still violate the specification after optimization. The results are presented in Table 1.1.

1.6 Modeling for Capacitor Array Layout in Charge-Redistribution ADCs

The charge-redistribution successive-approximation ADC is a common ADC topology. It is very advantageous from the point of view of power consumption, given the absence of DC currents in the capacitors. This ADC topology consists of a comparator, a capacitor array (which forms a DAC), and a successive-approximation register with the associated logic, as illustrated in Fig. 1.9. This work targets the design of the capacitor array: more

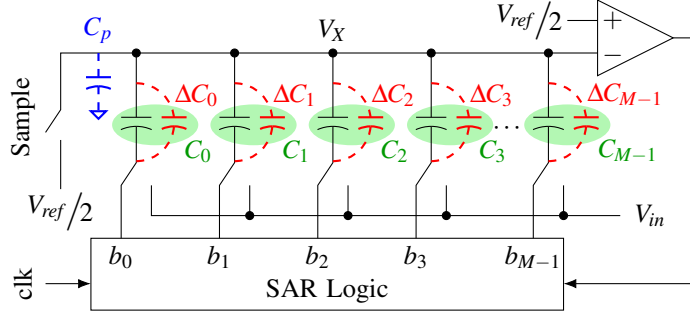


Figure 1.9: A schematic of a charge-redistribution SAR-ADC, with capacitive variations included: top plate-to-substrate parasitic (blue), and individual capacitor variation (red).

specifically, the optimal layout strategy for the capacitor array to minimize the static nonlinearity of the ADC.

Conventionally, the capacitors in the array have binary-weighted values. In this case, M , the number of capacitors, equals N , the number of bits. The number of cycles needed for a complete analog-to-digital conversion is also N . If the unit capacitor has the capacitance C_u , then the array capacitors have the following nominal values:

$$[C_{M-1_0}, C_{M-2_0}, \dots, C_{3_0}, C_{2_0}, C_{1_0}, C_{0_0}] = C_u \cdot [2^{N-1}, 2^{N-2}, \dots, 2^2, 2, 1, 1]. \quad (1.1)$$

The total number of unit capacitors in the array is 2^N . The bits $b_{M-1} = b_N$ to b_1 are iteratively adjusted during the conversion, while b_0 remains fixed to 0. This means that the bottom plate of C_0 is grounded in every iteration. This capacitor is introduced to have a resolution of $V_{ref}/2^N$ instead of $V_{ref}/(2^N - 1)$.

A fast ADC (an ADC operating at a high frequency) needs a comparator with a fast and precise response. However, such a comparator consumes a significant amount of power, because fast switching requires large currents. A solution to this problem is to use a slower comparator, which consumes less power, at the cost of reduced precision. The erroneous decisions of the comparator are compensated by using a successive-approximation algorithm with redundancy instead of the conventional successive-approximation algorithm with binary search.

The conversion algorithm with redundancy has the ability to recover, in the remaining iterations, after an erroneous comparator decision. Consequently, incomplete settling of the comparator output within a cycle is purposely permitted, such that the ADC can operate at a higher frequency than the binary-weighted version. In this way, the overall conversion time is lower for an ADC with redundancy.

The disadvantage of redundancy is the increased number of cycles for a conversion. The number of capacitors, M , is larger than the ADC bit-size, N . The nominal capacitor ratios are no longer binary-weighted; they are called generalized ratios. The example used in this work is a 12-bit ADC ($N = 12$) with 13 capacitors ($M = 13$) having the following nominal values:

$$[C_{M-1_0}, \dots, C_{0_0}] = C_u \cdot [1868, 1012, 552, 303, 168, 90, 48, 26, 14, 8, 4, 2, 1]. \quad (1.2)$$

1.6 Modeling for Capacitor Array Layout in Charge-Redistribution ADCs

However, the number of unit capacitors is 2^N , the same as in the binary-weighted case for the same bit-size N , in order to maintain the same resolution $V_{ref}/(2^N - 1)$. At the end of the conversion, the bit b_0 can be either 0 or 1. It does no longer have a constant value of 1, therefore the bottom plate of C_0 is no longer connected to ground during the entire conversion process.

In practice, ADCs are not ideal. An important source of non-ideality is the variation of capacitances from the nominal values. In Fig. 1.9, the capacitances are illustrated together with their variations, ΔC_0 to ΔC_{M-1} . Especially important is the variation of the capacitive ratios in the capacitor array, commonly called mismatch [46], rather than the variation of the individual capacitances. This work investigates the impact of capacitor mismatch on the static performance metrics of the ADC. The inaccuracy of capacitor ratios does not cause offset error [47] or gain error [48]. However, it causes nonlinearity, which is described by two performance metrics: integral and differential nonlinearity, abbreviated as INL and DNL, respectively.

To reduce these errors, layout techniques for capacitor matching have been proposed [49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59]. The contribution of this work is a model for INL and DNL in dependence of the capacitor mismatch, to be used for a layout strategy (consisting of placement and routing) that minimizes the impact of capacitor mismatch on the static linearity of the ADC. To analyze this impact, two essential modeling elements are required:

1. a model of capacitor mismatch,
2. a model of INL and DNL in dependence of the capacitive variations.

In the charge-redistribution ADC, the capacitor ratios are mainly affected by the following sources of variation:

1. Local, random fluctuations in the manufacturing process.

This source of variation has a statistical behavior and is described by two elements:

- the variation of a unit capacitor, which depends on the unit capacitor area; more specifically, the relative standard deviation of a unit capacitor is inversely proportional to the square root of the device area, according to Pelgrom's law [60], such that it decreases with increasing area.
- the correlation coefficient between two unit capacitors, which depends on the distance between two unit capacitors; this coefficient is assumed to have an exponential behavior in which it decreases with increasing distance.

Unlike in previous layout strategies [57, 58, 59], which considered only the correlation between devices in the statistical model, the model from this work combines it with the area dependence of the standard deviation. Device matching is improved by increasing the device area and/or the correlation between devices. Through area, there is a trade-off between variability and cost. The correlation between devices is increased through dispersion. For capacitors (in general, for any integrated devices) realized out of multiple unit devices, the unit devices belonging

1 Introduction

to separate capacitors should be as interleaved as possible in the layout. However, increasing the level of dispersion also increases the routing complexity. This trade-off plays an important role for the layout strategy.

2. Global gradients in the manufacturing process. This source of variation affects all devices systematically for given magnitude and orientation of the gradient. This work uses a model which assumes linear gradients over the entire capacitor array. Other models also consider second-order gradients [61].
3. Interconnection parasitic capacitances. From the parasitic capacitances, only the top-plate-to-bottom-plate capacitances (depicted by the red-colored parallel capacitors in Fig. 1.9) affect the static linearity. They are connected in parallel with the desired capacitances, contributing additively to their values. Because the top plates of all array capacitors are shortcircuited, the routing problem consists of connecting the unit capacitors' bottom plates to the corresponding exit pin. The routing complexity increases with increasing dispersion. The common placement and routing strategies attempt to match the aforementioned parasitic capacitances in order to reduce (ideally cancel) their impact on the accuracy of capacitor ratios.

Previous work on capacitor placement attempted to optimize the accuracy of capacitor ratios by maximizing the overall correlation (the sum of the correlation coefficients of all pairs of unit capacitors), assuming an underlying exponential spatial correlation model. Some examples are simulated annealing approaches [58, 59], matrix adjustment [56] and heuristics based optimization [57]. The objective of these methods was the ratio accuracy in general. However, when targeting particular objective functions (e.g., the static linearity) for specific circuit topologies (e.g., charge-redistribution SAR-ADC), the results obtained with these methods could be suboptimal.

The performance metrics for the linearity of an ADC are INL and DNL. In [49], we derived a simple analytical model for a charge-redistribution ADC with binary-weighted capacitor ratios, describing the static INL and DNL as linear expressions of capacitor variations. This linear relationship holds as long as the capacitive variations are small. In this work, the model is enhanced for charge-redistribution ADCs with generalized ratios.

For a general ADC, INL and DNL are defined for every binary code from 1 to $2^N - 1$. Therefore, there is a linearized expression with respect to capacitive variations for each of the $2^N - 1$ components of the INL and DNL, respectively. For a charge-redistribution ADC with binary-weighted capacitance ratios, it is shown in [49] that every INL component has a unique expression. However, it is shown as well that out of the $2^N - 1$ DNL components only N have unique expressions:

$$\overline{DNL} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & -1 & 0 \\ & & & & \vdots & & & \\ 0 & 1 & -1 & \cdots & -1 & -1 & -1 & 0 \\ 1 & -1 & -1 & \cdots & -1 & -1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta C_0 \\ \Delta C_1 \\ \vdots \\ \Delta C_{N-1} \\ \Delta C_N \end{bmatrix} \cdot \frac{1}{C_u}. \quad (1.3)$$

1.6 Modeling for Capacitor Array Layout in Charge-Redistribution ADCs

This reduced set of expressions showing a regular pattern provides a good insight into the capacitor placement strategy. Also, the expressions hold irrespective of the source of capacitor variation. When developing the placement, INL is not explicitly used. However, because INL and DNL share the same null space, minimizing DNL significantly reduces INL.

In the case of the charge-redistribution ADC with generalized ratios, the number of DNL components with unique expressions is larger than the bit-size N . For the 12-bit ADC with $M = 13$ capacitors from this work, with the capacitance ratios given in Eq. 1.2, a reduced set of 19 DNL expressions was found:

$$\overline{DNL} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 1 & -1 & -1 & 0 & -1 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 0 & -1 & -1 & 0 & 0 & -1 & -1 & 0 \\ 0 & 1 & -1 & -1 & 0 & -1 & -1 & 0 & -1 & 0 & -1 & 0 & 0 \\ 1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 0 & -1 & -1 & 0 & -1 & -1 & 0 & -1 & 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} \Delta C_0 \\ \Delta C_1 \\ \vdots \\ \Delta C_{M-2} \\ \Delta C_{M-1} \end{bmatrix} \cdot \frac{1}{C_u}.$$

(1.4)

Unlike for the binary-weighted ratios, the DNL expressions do no longer show a regular pattern and have to be derived individually for each set of capacitance ratios. Nevertheless, this reduced set of expressions is used as well to conduct the placement and routing strategy.

Combining the linearized model of INL and DNL with the exponential spatial correlation model produced, in [49], an optimal placement methodology for minimizing the impact of statistical (random fluctuations) and systematic (process gradients) variations on the static nonlinearity (as measured by INL and DNL) of a charge-redistribution ADC with binary-weighted capacitor ratios. The resulting placement style, named “chessboard”, complied with general layout guidelines for this problem [46]: coincidence, symmetry, dispersion, and compactness, with dispersion being here the key criterion.

However, the chessboard placement style suffers from an important disadvantage: it does not provide a solution for routing. Interconnection parasitics between the top and bottom plates additively contribute to the overall capacitances. Therefore, matching

1 Introduction

these parasitic capacitances is a prerequisite for matching the total capacitances. Moreover, the very high dispersion from the chessboard placement raises a very complex problem of routing.

The layout methodologies from [62, 52, 53] were developed with a strong interdependence between placement and routing, resulting in “globally distributed, locally connected” capacitor arrays. In [55], the coupling capacitance between routing tracks was considered with a dependency on the distance between tracks. In [54], integer linear programming was applied to solve the routing problem. However, the latter two approaches targeted the optimization of capacitor ratios in general and not specifically the minimization of INL and DNL. In all these publications [51, 62, 53, 54, 55], INL and DNL were evaluated using their complete, original expressions instead of the linearized model introduced in [49]. The complete definitions have the disadvantage that they do not provide insight into the layout strategy.

The modeling contribution of this work is a linearized analytical model of INL and DNL with respect to capacitor variations for any SAR-ADC with generalized capacitor ratios. In this model, the three sources of nonlinearity, statistical, systematic and parasitic are independent and additive. The model is then applied in developing a layout automation algorithm towards minimization of the static INL and DNL by considering the capacitor ratio inaccuracy caused, on the one hand, by the placement-related mismatch, and on the other hand by the parasitic capacitances of the interconnection wires of the bottom plates.

The resulting placement and routing for the 12-bit ADC with the capacitor ratios given in Eq. (1.1) are depicted in Fig. 1.10. The following can be observed, in connection to the underlying model:

- The capacitors with a large number of units are very dispersed over the array, similarly to the chessboard style. These capacitors, having large values, have a significant contribution to INL and DNL. For them, it is therefore critical to reduce the impact of local, random mismatch on INL and DNL.
- The capacitors with a small number of units are not dispersed, but instead grouped together, in the middle of the array, in opposition to the chessboard style. Having smaller values, these capacitors contribute less to INL and DNL through their local, random mismatch. However, parasitic capacitances should also be kept small, in order to be proportional to the desired capacitances. Therefore, these capacitance require short wires, which makes routing critical.

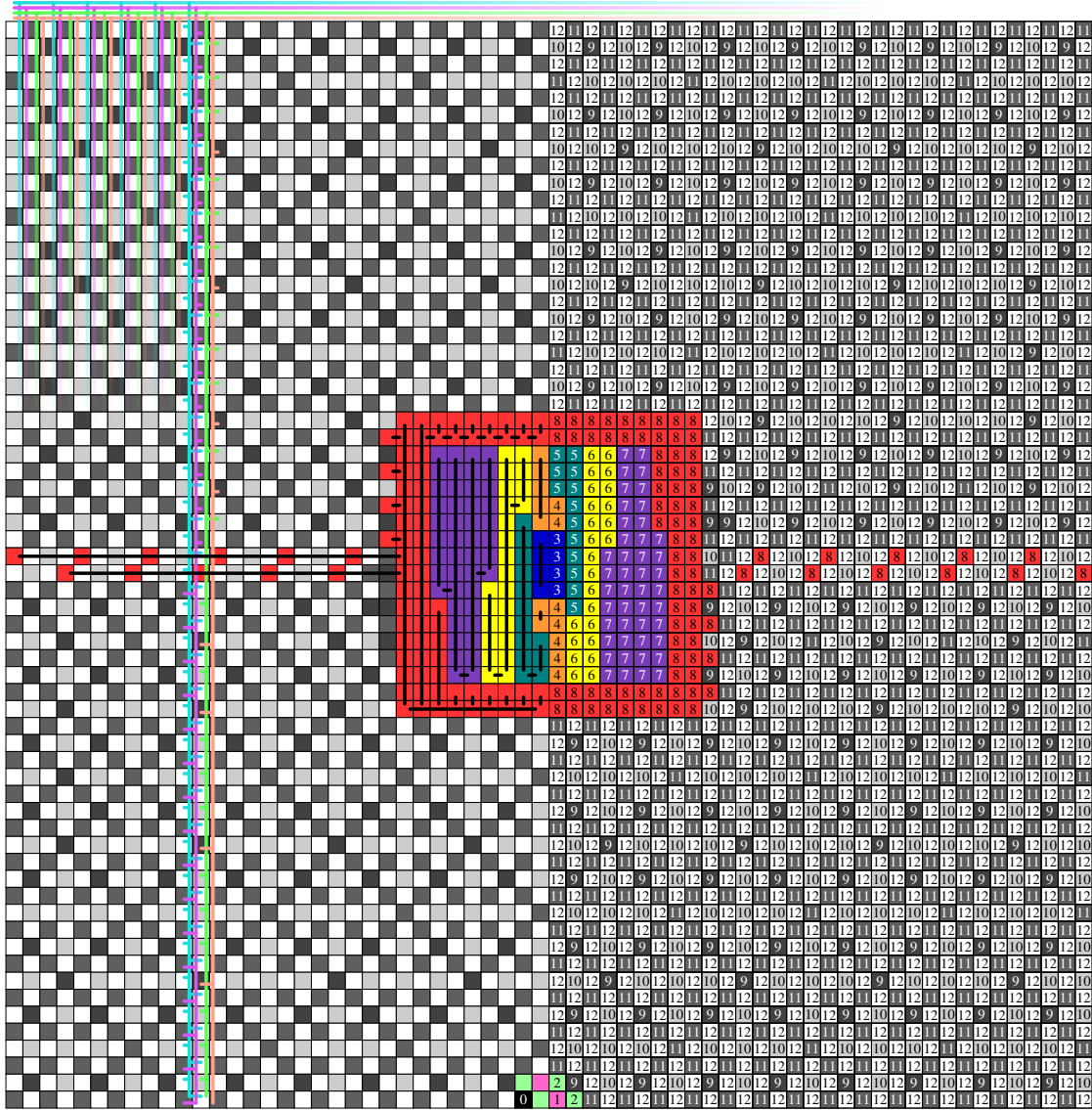


Figure 1.10: Final placement for the 12-bit architecture with generalized ratios characterized by Eq. (1.1), also depicting routing.

2 Modeling for Simulation of Integrated DC-DC Converters

This chapter presents a multi-harmonic modeling and simulation technique for low-power on-chip PWM DC-DC converters. The technique is based on the large-signal averaged modeling of the PWM switch cell and the Fourier series expansion of the voltages and currents at the switch cell terminals. The distinguishing feature of this technique is generality. With no additional modeling effort, it is applicable to a wide range of DC-DC converters and it supports any parasitic elements, any surrounding circuitry and any number of harmonics.

Circuits for dynamic characterization of the PWM switch cell parameters are presented and brought together with the PWM switch cell averaged model and the surrounding circuitry into the same testbench.

The proposed method is illustrated for a Buck and for a Boost converter, both implemented at transistor level. It achieves a speedup of more than one order of magnitude over the full transistor-level simulation, with a loss of accuracy below 3%. The method correctly predicts subharmonic instability.

This chapter consists of the following publication:

- F. Burcea, D. Tannir and H. E. Graeb, "Fast SPICE-Compatible Simulation of Low-Power On-Chip PWM DC-DC Converters With Improved Ripple Accuracy," in *IEEE Transactions on Power Electronics*, vol. 35, no. 8, pp. 8173-8185, Aug. 2020.

The paper was written under the leading pen of the candidate. The candidate conceptualized the paper and was responsible for the paper organization and for writing and editing the manuscript. The candidate brought the major technical contributions, which are:

- implementation of the large-signal averaged model,
- concept, derivation and implementation of the Fourier-series-based model,
- concept and implementation of the simulation testbenches,
- implementation of the two circuit examples,
- obtaining of the experimental results.

Fast SPICE-Compatible Simulation of Low-Power On-Chip PWM DC-DC Converters with Improved Ripple Accuracy

Florin Burcea, Dani Tannir, *Senior Member, IEEE*, and Helmut Graeb, *Fellow, IEEE*

Abstract—The circuit averaging technique has long been used as the basis for modeling the behavioral effects of switched-mode pulse-width-modulated (PWM) DC-DC power converter circuits due to its simplicity and efficiency in simulation. However, circuit-averaged models struggle to capture the effects of higher order harmonics on the output waveforms. Alternatively, multi-harmonic models that capture high frequency characteristics of output waveforms are typically very complex and computationally expensive. A general, efficient and accurate multi-harmonic modeling and simulation technique for low-power on-chip PWM DC-DC converters is presented in this paper. The technique is based on the large-signal averaged model of the PWM switch cell and on the Fourier series expansion of the typical converter waveforms. Its applicability range includes current mode controlled DC-DC converters. The method is exemplified on a Buck and on a Boost converter and achieves a speedup of one order of magnitude with an accuracy loss below 3% over the transistor-level simulation. The method accounts for non-ideal circuitry and supports any number of harmonics.

Index Terms—deep-submicron CMOS, low-power on-chip DC-DC converter, current-mode control, large-signal averaged model, multi-harmonic modeling, Fourier series.

I. INTRODUCTION

The simulation of DC-DC converters is generally very time-consuming due to the co-existence of fast switching activities and slow load variations. Circuit averaging techniques average the switching behavior and focus on the cycle-to-cycle dynamics. The drawback of circuit-averaging techniques, however, is the lack of information on the dynamic behavior within a period and in their limited ability to accurately capture waveform ripples. State-space averaging has been a popular simulation technique of PWM DC-DC converters [1]–[3]. However, its main challenge lies in the circuit-specific approach, thereby limiting the ability to automate the modeling and simulation process on arbitrary DC-DC converter topologies. Alternatively, the PWM switch cell model, which was first introduced in [4], [5] and which is a linearization of the three-terminal switch cell, can be readily applied to a wide

Florin Burcea and Helmut Graeb are with the Chair of Electronic Design Automation, Technical University of Munich, Munich 80333, Germany (e-mail: florin.burcea@tum.de; helmut.graeb@tum.de).

Dani Tannir is with the Electrical and Computer Engineering Department, Lebanese American University, P.O. Box 36, Byblos, Lebanon (e-mail: dani.tannir@lau.edu.lb).

©2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, including reprinting/republishing this material for advertising or promotional purposes, collecting new collected works for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. Authors' version. Definitive version published in IEEE Trans. on Power Electronics, <https://ieeexplore.ieee.org/document/8937511>.

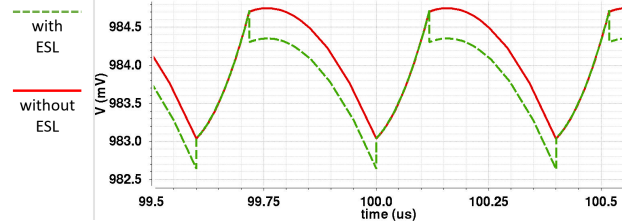


Fig. 1. Output voltage waveform of a Buck converter with current mode control with (dashed line) and without (solid line) equivalent series inductance (ESL) of the output capacitor.

range of DC-DC converters. Enhancements to this model were developed in [6]–[8]. While the PWM switch cell model is simple to apply and very efficient during simulation, its major limitation is that it neglects the dynamic behavior of DC-DC converters within one cycle and provides no information about the waveform ripples [9], [10].

To account for the ripples, a lot of work has been done on generalized averaging techniques [10]–[12]. In [13], [14] a complete multi-harmonic solution to low-power PWM DC-DC converter simulations was presented. The multi-frequency model is general in the sense that it can be applied to many DC-DC converters including Buck, Boost and Buck-Boost converters without any modifications. However, it is important to note that the output signals of DC-DC converters contain several high frequency characteristics and non-conventional looking ripple waveforms, such as the green curve shown in Fig. 1. Such waveform characteristics can be very difficult to capture without accounting for a high number of harmonics in the model. In that case, the developed model would become extremely complex. For this reason, analytical Fourier series based methods have been developed to accurately capture the output ripple waveforms using a large number of harmonics [15]–[20]. Since these methods are analytical in nature, they are based on circuit-specific equations that are expanded using Fourier Series, which means that these models have to be derived for each converter topology. In [21], a general Fourier series based model is presented that is not circuit-specific. The model focuses on capturing targeted small-signal properties and transfer functions and it is therefore useful primarily as a tool for designing appropriate feedback control.

Simulation methods which are particularly efficient in the transient analysis of switching converters have been developed and incorporated into the commercial tool SIMetrix/Simplis [22]. However, this simulation approach is applicable in the system-level analysis of switching convert-

ers realized with discrete components, not in the transistor-level simulation of low-power integrated switching converters, which is the target application area of this paper's contribution.

In this paper, we address the limitations of Fourier series-based modeling techniques by presenting a simulation-based multi-harmonic model that accurately captures the non-conventional characteristics of output waveforms in steady-state in conjunction with circuit averaging techniques. This method significantly reduces the simulation time even while accounting for a large number of harmonics. In our proposed model, the circuit averaging technique is first used to determine the dc operating point of the converter, from which the Fourier series coefficients are then extracted to accurately reconstruct the output ripple waveforms. The proposed model is general in the sense that it is not derived from topology specific circuit equations, but from the common properties of DC-DC converters: the voltage and current waveforms at the PWM switch cell terminals. Therefore, the model can be adapted and applied to arbitrary converter circuit topologies. We will show that the model can be applied to low-power converters with current-mode control, which not many models account for in the literature. Of the existing work addressing converter topologies with current-mode control, the models in [21], [23] are the most related to this work. They present small-signal models and apply to multi-phase converter designs, while this paper presents general large-signal models and targets low-power integrated circuit applications. Finally, we will show that our model brings one order of magnitude reduction in simulation time while maintaining remarkable accuracy even for a large number of harmonics.

The presented approach is an abstraction level in-between averaged models and full transistor-level simulation. The method incorporates the state-of-the-art averaged model that allows for fast simulation of the transient response and the capability of performing small-signal analysis. While the averaged model does not provide information about the waveform ripple, this method presents a novel approach for ripple reconstruction, with an accuracy close to transistor-level simulation, but with a speed improvement of over one order of magnitude.

In more detail, the following new contributions are presented:

- Test circuits for the dynamic characterization of the parameters of the PWM switch cell model are presented for Buck and Boost converters with current mode control (CMC) operating in continuous conduction mode (CCM).
- The Fourier series expansions for the currents and voltages at the PWM switch cell terminals are calculated.
- The Fourier series coefficients and the small signal transfer function, obtained through simulation, are used to reconstruct the ripple of any waveform in the circuit, for any number of harmonics.

The key characteristic of the method presented in this paper, which distinguishes it from any previous similar work, is generality. Specifically, this method provides the following features with no additional modeling effort:

- It covers any DC-DC converter topology with current mode control. The applicability range is the same for this method as for the PWM switch cell model. The method can be easily transferred to architectures with other duty

cycle control methods (e.g., voltage mode), by simply replacing the PWM switch cell model accordingly.

- It accounts for any parasitic element (e.g., the equivalent series inductance of the output capacitor, which is not included in any other modeling paper).
- It accounts for any transistor-level surrounding circuitry (e.g., error amplifier, current sensing), while all other modeling papers use ideal circuitry.
- By increasing the number of harmonics, increasing levels of precision can be achieved (the choice is determined by the trade-off between accuracy and computational effort/time).

The method is particularly useful in the transistor-level simulation of integrated switching converters. At this simulation level, simulators (e.g., Spectre) are needed that can efficiently handle complex transistor models (e.g., BSIM). This method is implemented within the framework of the commercial tool Cadence Virtuoso.

The remainder of the paper is organized as follows. Section II gives an overview of the characteristics and operation of current mode controlled Buck and Boost converters. Section III presents the PWM switch cell large-signal averaged model. Section IV describes the circuits for dynamic characterization of the PWM switch cell model parameters during simulation. Section V presents the Fourier series expansion for the voltages and currents at the PWM switch cell nodes and then the new simulation flow. The simulation flow is then illustrated by the experimental results in Section VI.

II. DC-DC CONVERTERS WITH CURRENT-MODE CONTROL

A. Buck Converter

Fig. 2 shows the topology of a synchronous Buck converter with peak current control. The inductor L is shown together with its parasitic resistance R_L and the output capacitor C_{out} together with its parasitic elements: the equivalent series resistance (ESR) and inductance (ESL).

The blue box encircles the circuit elements integrated on a single chip. The elements implemented as discrete components are the following:

- The inductor L , as integrated inductors have a poor performance,
- The output capacitor C_{out} , as capacitances in the range of μF (typical value for the output capacitor of a DC-DC converter) are not feasible for integration.

The dashed box encloses the switching elements of the converter. The meaning of the terminals a, p and c will be explained in Section III. The core of this subcircuit is composed of two switches: the high-side switch SW_a , implemented as a PMOS transistor, and the low-side switch SW_p , implemented as an NMOS transistor. The other elements of the subcircuit are part of the control loops.

The topology has two control loops. The first loop regulates the output voltage V_{out} by comparing it with the reference voltage V_{ref} by means of an error amplifier. The second loop regulates the duty cycle of the gate driving signal, which is controlled by the comparator. The error voltage V_{err} fixes the peak value up to which the inductor current must increase before switching. Additionally, a compensation ramp is subtracted from V_{err} for stability purposes [7].

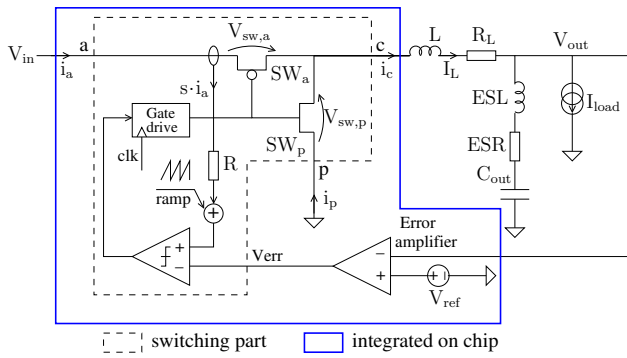


Fig. 2. Synchronous Buck converter with peak current control. The circuit part integrated on chip is shown distinctly.

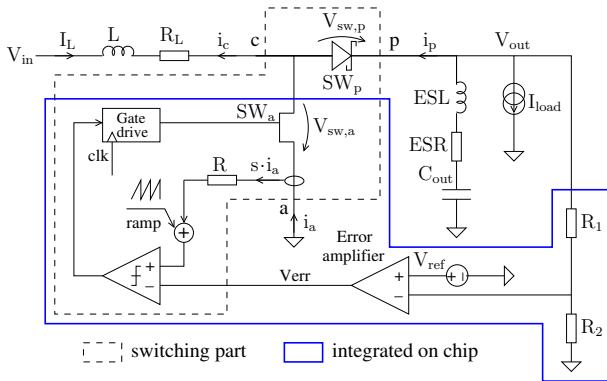


Fig. 3. Asynchronous Boost converter with peak current control. The circuit part integrated on chip is shown distinctly.

B. Boost Converter

Fig. 3 shows the topology of an asynchronous Boost converter with peak current control. There are several similarities to the Buck topology from the previous section:

- The inductor L and the output capacitor C_{out} are shown together with their parasitic elements.
- The dashed box encloses the switching part of the circuit. The core of this subcircuit consists of two switching elements: the high-side switch SW_p , implemented as a Schottky diode, and the low-side switch SW_a , implemented as an NMOS transistor. The meaning of the terminals a , p and c will be explained in Section III.
- The blue box encircles the circuit elements integrated on chip. In addition to the inductor L and the output capacitor C_{out} , the Schottky diode is also implemented as a discrete element.
- There are two control loops: one for regulating the output voltage V_{out} and the other for regulating the duty cycle. The control loops work in a similar way to the Buck example, as described in Section II-A.

Because the output voltage of the Boost is larger than the input, it is divided on a resistive divider in order to be compared with the reference voltage V_{ref} .

III. THE PWM SWITCH CELL LARGE-SIGNAL AVERAGED MODEL

Large-signal averaged models have been developed to reduce the computational cost of simulating switching circuits.

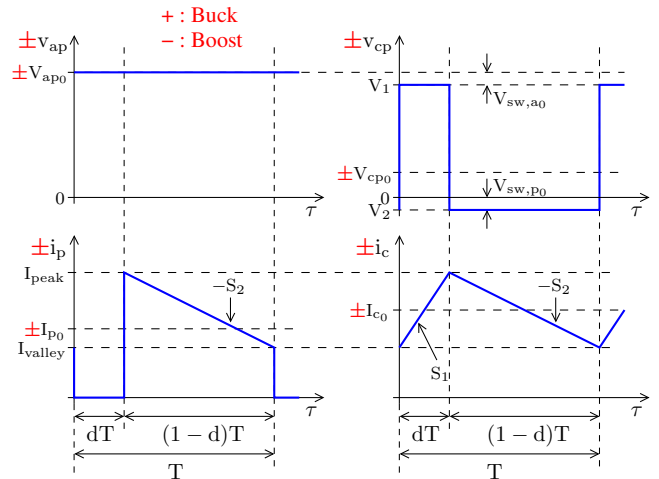


Fig. 4. PWM switch cell instantaneous terminal voltages and currents for a Buck (+) and a Boost (-) converter operating in CCM.

The simulation based on such models produces the waveforms averaged over a switching period and is much faster than the simulation based on the switched model (since the small time step in the transient analysis is no longer needed). Also, by linearizing the large-signal averaged model, the small-signal model is obtained. The drawback of using the averaged model is the accuracy loss, as the simulated waveforms over a period contain the average value, without the ripple.

The PWM switch model replaces the switching network of the converter (marked with the dashed box in Fig. 2 and 3) with a three-terminal model, having the nodes a (active), p (passive) and c (common, the junction between the two switches) [7]. This approach has the great advantage of generality, as it fits all DC-DC converter topologies just through model rotations.

A. Voltage and Current Waveforms

Fig. 4 shows the PWM switch cell instantaneous terminal voltages and currents and their average values for a Buck and for a Boost converter operating in CCM, where d represents the duty cycle and T the switching period. In the first phase, dT , the switch SW_a is on and the diode SW_p is off. In the second phase, $(1-d)T$, SW_a is off and SW_p is on. This holds for both converters. The waveforms are the same for the Buck and for the Boost converter, except for the opposite sign. For example, v_{cp} for the Buck overlaps $-v_{cp}$ for the Boost. Also, in the Boost converter i_c has an opposite direction with the inductor current, such that the bottom-right waveform of Fig. 4 represents $-i_c$. The average values of these waveforms are denoted with the index “0” and will be called from now on “index-0 components”. The terminal voltages and currents and the inductor current slopes for the Buck and for the Boost converter are given in Tables I and II.

B. PWM Switch Cell Equivalent Circuit

The large-signal averaged model developed in [4] expresses a relationship between the index-0 components for an ideal DC-DC converter. Adding the voltage drops on the switching elements gives the final large-signal averaged model [7], depicted in Fig. 5. In this model, V_{err} represents the output of

TABLE I
PWM SWITCH CELL TERMINAL VOLTAGES AND CURRENTS FOR
BUCK AND BOOST CONVERTERS

	Buck	Boost
V_{ap0}	V_{in}	$-V_{out}$
V_{cp0}	V_{out}	$V_{in} - V_{out}$
I_{c0}	I_{load}	$-I_{load}/d$
I_{p0}	$I_{load}(1-d)$	$-I_{load}$

TABLE II
PWM SWITCH CELL VOLTAGE LEVELS AND CURRENT SLOPES
AND LEVELS

	w/o losses	w/ losses
V_1	V_{ap0}	$V_{ap0} - V_{sw,a0}$
V_2	0	$-V_{sw,p0}$
S_1	V_{ac0}/L	$(V_{ac0} - V_{sw,a0})/L$
S_2	V_{cp0}/L	$(V_{cp0} + V_{sw,p0})/L$
I_{peak}	$I_{c0} + S_1 dT/2$	
I_{valley}	$I_{c0} - S_1 dT/2$	

the error amplifier, $V_{sw,a0}$ and $V_{sw,p0}$ are the voltage drops on SW_a and SW_p , respectively, during their on-state, R_i is the equivalent sensing resistor for the inductor current and S_a is the slope of the compensation ramp [7]. The current I_μ is a notation used for compactness purposes [7]. It represents that part of the current in node c dependent on the duty cycle d. The remaining part of I_c is dependent on V_{err} , the output of the error amplifier.

The capacitor C_s is introduced to model subharmonic oscillations [7]. The value of this capacitor is chosen such that it resonates with the inductor L at the frequency of the subharmonic oscillations, which is $F_{sw}/2$, half of the switching frequency. This creates a peak in the frequency response at $F_{sw}/2$, predicting the subharmonic instability.

The major advantage of the PWM switch cell model is its generality and flexibility: it holds both for the Buck and for the Boost converter (and also for other DC-DC converters [7]). The equivalent circuit of the PWM switch cell large signal averaged model depends on the control method. Each control technique (e.g., voltage mode, current mode) has its own equivalent circuit for the PWM switch cell. The equivalent circuit of the PWM switch cell large signal averaged model shown in Fig. 5 corresponds to that of current mode control. The test circuits and the simulation method presented in this paper are applicable for any control technique, as long as the PWM switch cell equivalent circuit specific to each technique is modeled accordingly.

IV. DYNAMIC CHARACTERIZATION OF THE LARGE-SIGNAL AVERAGED MODEL PARAMETERS

A. Parameter Dynamic Characterization

The PWM switch cell model for current mode control contains several parameters: L, T, $V_{sw,a0}$, $V_{sw,p0}$, R_i and S_a . The inductance L and the switching period T are design parameters, which need to be provided before the start of the simulation. The rest of the parameters are extracted dynamically, during the simulation of the circuit with the averaged PWM switch cell model and will each be explained next.

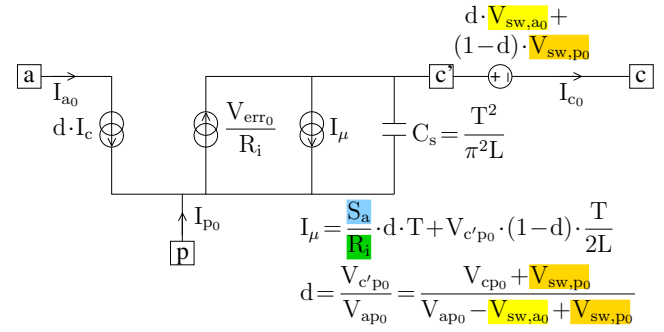


Fig. 5. Large-signal averaged model with voltage drops on switching elements for a DC-DC converter with peak current control.

The voltage drops on the switches, $V_{sw,a0}$ and $V_{sw,p0}$, depend on the current flowing through the switches during their on-state. In each phase, the current through the on-switch (SW_a in the first phase and SW_p in the second phase) is equal to the inductor current. The inductor current equals i_c for the Buck converter and $-i_c$ for the Boost converter and is represented in the bottom-right plot of Fig. 4. Therefore, by injecting the averaged value over a period of this current ($I_{L0} = |I_{c0}|$) into a copy of the switch, one can get the averaged voltage drop over the switch, as illustrated in Fig. 6 (a) for the synchronous Buck converter and in Fig. 7 (a) for the asynchronous Boost converter.

The equivalent sense resistance R_i is also extracted during simulation, by replicating the sensing circuit and the active switch SW_a (in both examples from this paper, the Buck converter from Fig. 2 and the Boost converter from Fig. 3, the current through the active switch is sensed), with the gate driven in the on-state, and injecting the index-0 inductor current into it, as explained in Fig. 6 (b) for the Buck converter and in Fig. 7 (b) for the Boost converter. Then, having the voltage drop V_R and the current to be sensed I_{L0} , the equivalent resistance R_i is obtained.

The last parameter is the slope of the compensation ramp, S_a . Unlike the other parameters, which depend on linear circuits, S_a is based on a switching circuit, depicted in Fig. 8 (a). At a time t after the switch turned off, C_a is charged at the voltage $V_a = I_a \cdot t / C_a$. The linear circuit in Fig. 8 (b) reproduces the state of the ramp generator at the time t . With two instances of this circuit, the state at two different time points within a period is reproduced: t_1 and t_2 , when C_a is charged at V_{a1} and V_{a2} , respectively. The time interval $\Delta t = t_2 - t_1$ is linked to the voltage difference $\Delta V_a = V_{a2} - V_{a1}$ by $\Delta t = C_a \cdot \Delta V_a / I_a$. Then, by making use of the linear increase in V_R and measuring this voltage for the two states, the compensation ramp is obtained: $S_a = (V_{R2} - V_{R1}) / \Delta t$.

B. Simulation Testbench

The circuit blocks described in the previous section are assembled to form the simulation testbench for the large-signal averaged model. Fig. 6 shows the testbench for the Buck converter and Fig. 7 for the Boost converter. The testbench contains the following parts:

- The DC-DC converter, shown in Fig. 6 (c) and Fig. 7 (c), respectively, in which the switching part (the dashed box

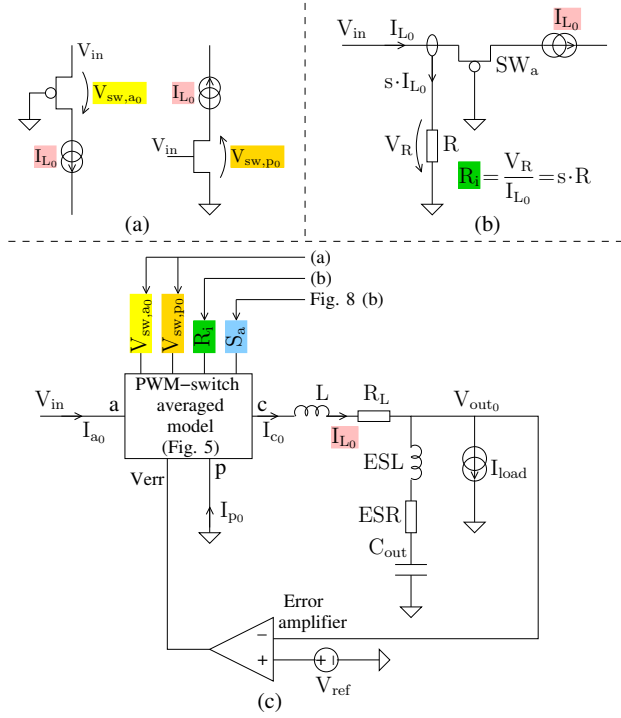


Fig. 6. Simulation setup for the synchronous Buck converter with the PWM switch cell large-signal averaged model. (a) Main circuit. (b) Circuit for dynamic evaluation of index-0 voltage drop on the switches. (c) Circuit for dynamic evaluation of equivalent sense resistance.

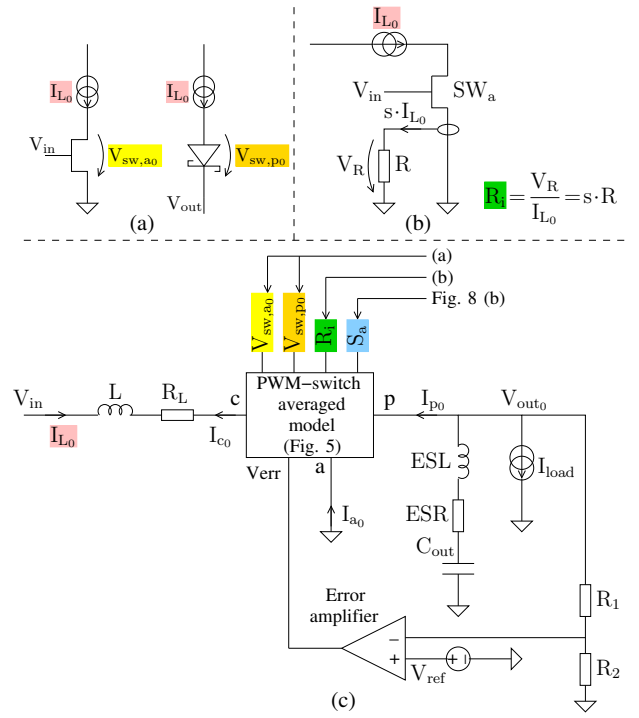


Fig. 7. Simulation setup for the asynchronous Boost converter with the PWM switch cell large-signal averaged model. (a) Main circuit. (b) Circuit for dynamic evaluation of index-0 voltage drop on the switches. (c) Circuit for dynamic evaluation of equivalent sense resistance.

in Fig. 2 and Fig. 3, respectively) is replaced by the PWM-switch averaged model from Fig. 5;

- The circuit for calculating the averaged voltage drops $V_{sw,a0}$ and $V_{sw,p0}$ on the switches SW_a and SW_p , shown in Fig. 6 (a) and Fig. 7 (a), respectively;
- The circuit for calculating the equivalent resistance R_i for current sensing, shown in Fig. 6 (b) and Fig. 7 (b), respectively;
- The circuit for calculating the slope of the compensation ramp, made up of two instances of the circuit from Fig. 8 (b).

The circuit blocks from the averaged-signal simulation testbench form a feedback loop. For the Buck converter, for example, the inductor current I_{L_0} from the main circuit from Fig. 6 (c) is replicated and injected into the circuits from Fig. 6 (a) and Fig. 6 (b), to calculate the parameters $V_{sw,a0}$, $V_{sw,p0}$ and R_i . The values of these parameters are then feedback to the main circuit from Fig. 6 (c), namely to the PWM-switch averaged model. The ramp generation circuit from Fig. 8 (b) is also part of the testbench, but it provides the compensation ramp S_a to the model in an open loop. The same explanations hold for the Boost converters (Fig. 7).

V. FOURIER-SERIES REPRESENTATION OF PWM SWITCH CELL SIGNALS

This section provides the Fourier series expansions for the signals from Fig. 4, the voltage v_{cp} and the currents i_c and i_p of a DC-DC converter operating in CCM. Then, it will be shown how the Fourier series of these three signals and the small-signal analysis of the circuit can be used to reconstruct other signals in the circuit.

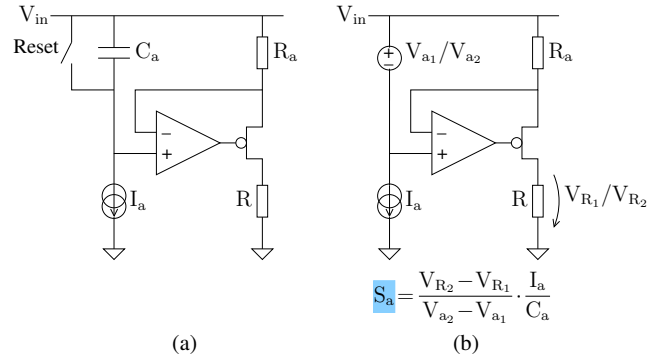


Fig. 8. Dynamic calculation of the slope of the compensation ramp. (a) Switching circuit. (b) Averaged model, which requires two instances of this circuit.

A periodic signal with period T and angular frequency $\omega = 2\pi/T$ can be represented by its complex Fourier series:

$$x(\tau) = \sum_{n=-\infty}^{\infty} X_n \exp(jn\omega\tau), \quad (1)$$

where the coefficients X_n are calculated as:

$$X_n = \frac{1}{T} \int_t^{t+T} x(t) \exp(-jn\omega\tau) dt. \quad (2)$$

Then, the signal $x(t)$ can be reconstructed from its first N harmonics:

$$x(\tau) = X_0 + 2 \sum_{n=1}^N |X_n| \cos(n\omega\tau + \arg(X_n)). \quad (3)$$

The precision in reconstructing $x(t)$ increases with the number of harmonics N .

Applying (2) to these waveforms from Fig. 4, the coefficients of the Fourier series expansion are obtained:

$$\begin{aligned} V_{cp_n} &= |V_{cp_n}| \exp(j \arg(V_{cp_n})) \\ I_{c_n} &= |I_{c_n}| \exp(j \arg(I_{c_n})) \\ I_{p_n} &= |I_{p_n}| \exp(j \arg(I_{p_n})), \end{aligned} \quad (4)$$

where

$$\begin{aligned} |V_{cp_n}| &= (V_1 - V_2) \frac{\sin(n\pi d)}{n\pi} \\ &= (V_{ap_0} - V_{sw,a_0} + V_{sw,p_0}) \frac{\sin(n\pi d)}{n\pi} \end{aligned} \quad (5)$$

$$\arg(V_{cp_n}) = -\pi n d$$

and

$$\begin{aligned} |I_{c_n}| &= \frac{T(S_1 + S_2)}{2} \frac{\sin(n\pi d)}{(n\pi)^2} \\ &= \frac{T(V_{ac_0} - V_{sw,a_0} + V_{cp_0} + V_{sw,p_0})}{2L} \frac{\sin(n\pi d)}{(n\pi)^2} \end{aligned} \quad (6)$$

$$\arg(I_{c_n}) = -\pi(nd + 0.5)$$

For the current i_p , the complex coefficients of the Fourier series expansion are calculated as:

$$I_{p_n} = \frac{2 \exp(-jn\pi d) \sin(n\pi d) \left(\frac{S_2 T}{2n\pi} - jI_{peak} \right) + S_2 T(1-d)}{2jn\pi}, \quad (7)$$

where S_2 and I_{peak} are given in Table II. Because of the shape of this waveform, the expression is more complex than in the case of v_{cp} and i_c . Separating I_{p_n} into the magnitude and phase does not lead to simple compact expression. I_{p_n} is hence left in the complex form.

The index-0 components V_{ap_0} , V_{ac_0} , V_{cp_0} , V_{sw,a_0} and V_{sw,p_0} are obtained from the simulation of the large-signal averaged model, using the simulation testbench described in Fig. 6 for the Buck converter and in Fig. 7 for the Boost converter. The duty cycle d is calculated from the index-0 components, as given in Fig. 5:

$$d = \frac{V_{cp_0} + V_{sw,p_0}}{V_{ap_0} - V_{sw,a_0} + V_{sw,p_0}} \quad (8)$$

Having the index-0 components and the duty cycle, obtained from the averaged large-signal simulation, the Fourier series coefficients of v_{cp} , i_c and i_p can be computed. Then, with the small-signal analysis, the transfer function of any voltage or current in the circuit relative to v_{cp} , i_c or i_p at any integer multiple of the switching frequency is calculated. The small-signal transfer function at a specified frequency is a complex number with magnitude and phase. This transfer function is used to reconstruct the steady-state waveform of that signal. The following example will illustrate the method.

Consider the transfer function of v_{out} vs. v_{cp} at frequency f , denoted as $H(v_{out}, v_{cp}, f)$, having a magnitude $|H(v_{out}, v_{cp}, f)|$ and a phase $\arg(H(v_{out}, v_{cp}, f))$:

$$H(v_{out}, v_{cp}, f) = |H(v_{out}, v_{cp}, f)| \exp(j \arg(H(v_{out}, v_{cp}, f))). \quad (9)$$

The output voltage v_{out} has a small ripple around the average value V_{out_0} . Therefore, the small-signal assumption

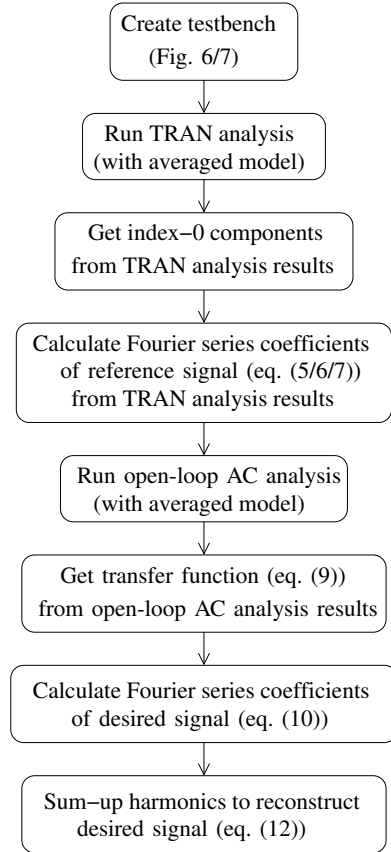


Fig. 9. Simulation flow based on PWM switch cell averaged model and Fourier series.

can be applied. Using the transfer function $H(v_{out}, v_{cp}, f)$ and the coefficients of the Fourier series expansion of v_{cp} , the harmonics (at multiples of the switching frequency $1/T$) of v_{out} are calculated:

$$V_{out_n} = V_{cp_n} \cdot H\left(v_{out}, v_{cp}, \frac{n}{T}\right) = |V_{out_n}| \cdot \exp(j \arg(V_{out_n})), \quad (10)$$

where

$$\begin{aligned} |V_{out_n}| &= |V_{cp_n}| \cdot \left| H\left(v_{out}, v_{cp}, \frac{n}{T}\right) \right| \\ \arg(V_{out_n}) &= \arg(V_{cp_n}) + \arg(V_{out_n}), \end{aligned} \quad (11)$$

with $|V_{cp_n}|$ and $\arg(V_{cp_n})$ given in (5).

Afterwards, using (3), the signal $v_{out}(\tau)$ in steady-state can be reconstructed from its Fourier series expansion, with the desired number of harmonics N :

$$v_{out}(\tau) = V_{out_0} + 2 \sum_{n=1}^N |V_{out_n}| \cos\left(2\pi \frac{n}{T} \tau + \arg(V_{out_n})\right). \quad (12)$$

The same procedure is applied for any other signal in the circuit, relative to, v_{cp} , i_c or i_p . As it will be shown in the experimental results, the signal on which to refer is chosen depending on the converter type. For the Buck converter, v_{cp} or i_c can be used with similar precision, while for the Boost converter, using i_p provides the best precision. The flowchart shown in Fig. 9 summarizes the proposed modeling approach presented in this section.

VI. SIMULATION RESULTS

In this section, the proposed multi-harmonic averaged model is tested on a Buck and on a Boost converter, to demonstrate its accuracy and speedup. The proposed model is compared with the simulation results of the original transistor-level converter circuits, which are implemented in a standard CMOS 180nm 5V technology. The PWM switch cell large-signal averaged model is implemented in VerilogA. All circuits are simulated using the commercial tool Cadence Spectre on a Linux workstation with a 3.4GHz Intel i5 processor and 8GB of RAM. In the proposed model, the circuits are simulated in the same conditions and then the computation of the waveform ripples is done with a Cadence SKILL script.

Switching circuits cannot be simulated directly (at transistor level) in the frequency domain. This kind of analysis becomes possible by means of the large-signal averaged model of the PWM switch cell. Previous work proposed a linearized (small-signal) version of this model [4], [5], [7]. However, in modern simulators like Cadence Spectre, nonlinear devices (e.g., the MOS transistor, the PWM switch cell) are automatically linearized by the simulator for the small-signal analysis. Therefore, it is not necessary to manually implement a linearized version of the PWM switch cell model, but the small-signal analysis (e.g. open-loop AC analysis) can be applied directly to this model.

A. Buck Converter

The first example circuit is a synchronous Buck converter operating in CCM, with current mode control of the duty cycle, as the one shown in Fig. 2. The converter receives an input voltage of $V_{in} = 4V$ (which is also the supply voltage of the entire circuit) and produces a desired output voltage $V_{out} = 1V$ at a load current $I_{load} = 2.5A$. The switching frequency is $F_{sw} = 2.5MHz$, equivalent to a switching period $T = 400ns$. The component values are as follows:

- Inductance $L = 1\mu H$, with parasitic equivalent series resistance $R_L = 50m\Omega$,
- Output capacitance $C_{out} = 20\mu F$, with parasitic equivalent series resistance $ESR = 10m\Omega$ and equivalent series inductance $ESL = 100pH$.

Previous work dealing with multi-harmonic Buck converter modeling typically did not consider the equivalent series inductance of the output capacitor, as this would significantly increase the complexity of the equations for the methods based on state-space averaging. However, for a converter operating at a high switching frequency, this parasitic element has a noticeable effect on the output voltage ripple, as shown in Fig. 1 in Section I, and therefore should not be ignored.

The Buck converter is implemented at transistor level. As it is a switching circuit, transient analysis is necessary to simulate its behavior. The analysis requires a small time step. The simulation is run over 1ms, meaning 2500 periods, as the switching frequency is 2.5MHz. For this simulation time, 1.3M points are simulated and the total CPU time is 443s in the transistor-level transient simulation.

The conventional averaged model provides much faster simulation capabilities. For the same simulation time of 1ms, only 669 points are simulated, meaning 5000 times less than with the transistor model, and the transient analysis lasts only

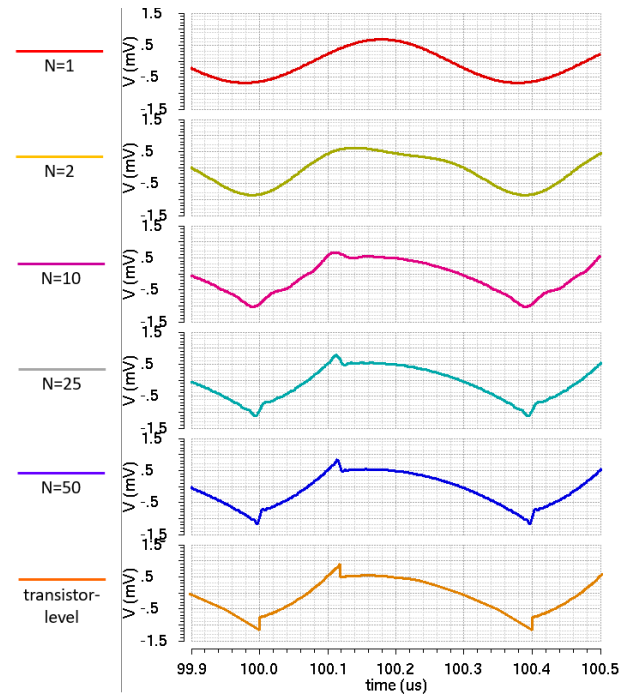


Fig. 10. Output voltage (V_{out}) ripple (total voltage minus average value) for the Buck converter. Reconstructed waveforms, from the averaged-model simulation, with a gradually increasing number of harmonics: 1, 2, 10, 25, and 50, respectively. At the bottom, the waveform resulted from transistor-level simulation.

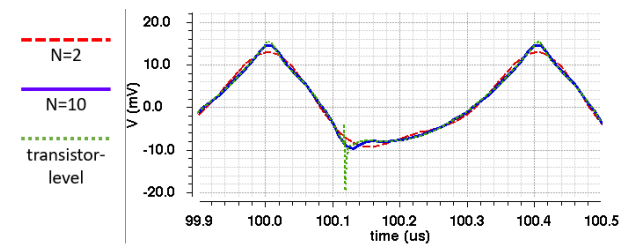


Fig. 11. Error voltage (V_{err}) ripple (total voltage minus average value) for the Buck converter. Reconstructed waveforms, from the averaged-model simulation, with 2 and 10 harmonics, respectively, and the waveform resulted from transistor-level simulation.

1s in this case. The disadvantage is that it produces only the waveform values averaged over a period.

On the other hand, by using the proposed method described in Section V, a good approximation of the transistor-level waveforms in steady state can be obtained in a shorter time than with the transistor-level simulation. Note that, in addition to the transient analysis, a small-signal analysis (open-loop AC analysis) is necessary, to obtain the small-signal transfer functions. Its CPU runtime is, however, negligible.

Figs. 10 and 11 visually compare the waveforms obtained from the original transistor-level circuit with the waveforms reconstructed using the proposed model with an increasing number of harmonics. Fig. 10 shows the output voltage V_{out} . This is the most important waveform in the circuit to be reconstructed, because common performance figures, like the output peak-to-peak ripple, are related to it. The output voltage is reconstructed with a gradually increasing number of harmonics: 1, 2, 10, 25 and 50, respectively. When considering

only the 1st order component, the reconstructed ripple is a sine wave. With components up to the 10th order, the waveform begins to look similar to the transistor-level waveform. With components up to the 50th order, the reconstructed waveform is almost identical to the transistor-level one.

The ripple waveform can be reconstructed for other signals in the circuit as well. Fig. 11 shows a comparison between the reconstructed ripple waveform (using 2 and 10 harmonics) and the transistor-level ripple waveform for the error voltage (V_{err} , the output of the error amplifier). For the ripple waveform generated with 2 harmonics, the difference to the transistor-level model is visible, but with 10 harmonics the waveforms almost overlap. The same method can be applied for the inductor current I_L and the capacitor current I_{cap} (both have a triangular shape).

The ripple waveforms for the signals of a Buck converter can be reconstructed by taking either the inductor current i_c or the voltage v_{cp} as the reference signal. The difference is insignificant, as seen in Table III, which quantitatively evaluates the accuracy of the waveforms generated using the proposed approach. The table shows the RMS error of the reconstructed waveform vs. the transistor-level waveform over one period in steady-state. The RMS error, which is a standard error metric, is calculated as:

$$Error_{RMS} = \frac{1}{T} \sqrt{\int_t^{t+T} \left(\frac{x_{out}(\tau) - x_{out,N}(\tau)}{X_{out,pp}} \right)^2 d\tau}, \quad (13)$$

where x_{out} is the ripple of transistor-level waveform, $x_{out,N}$ is the ripple reconstructed with N harmonics and $X_{out,pp}$ is the peak-to-peak magnitude of the transistor-level ripple. When computing the RMS error, only the ripples were considered, centered around 0. Note that the averaged values are subtracted due to the small error introduced by the averaged model compared to the transistor-level model. The differences in the averaged values are in the order of mV. However, this is a limitation of the averaged model, not of the method based on Fourier series for reconstructing the ripple. In order to evaluate the accuracy of this method, Table III only compares the ripples. For V_{out} , the RMS error is very close to 2% for 10 harmonics and already at 1% for 50 harmonics. For V_{err} , the RMS error for 10 harmonics is very close to 2%. For a larger number of harmonics, there is almost no improvement in the accuracy of the reconstructed ripple of V_{err} . The reason for this is the negative spike of the transistor-level waveforms, as seen in Fig. 11. This spike does not appear in the reconstructed waveforms and so causes a systematic error. In order for it to be accounted for, the number of harmonics needs to be significantly larger than 50, which is impractical from the CPU runtime point of view.

The accuracy of the reconstructed waveforms is also a result of including nonideal elements. For example, the voltage drops on the switches during their on state, which have typical values of 50mV to 100mV, are considered, as described in the simulation setup from Fig. 6 and in the expressions for ripple calculation derived in Section V. Table IV shows the RMS error for the reconstructed waveform (with and without considering voltage drops on the switches) vs. the accurate transistor-level waveform (with non-ideal switches). With no voltage drops on the switches, the error is larger. For V_{out} , the error drops below 1% with non-ideal switches, but remains

TABLE III
RMS ERROR OVER ONE PERIOD OF RECONSTRUCTED BUCK WAVEFORM RIPPLES VS. COMPLETE WAVEFORM RIPPLE

Number of Harmonics N	V_{out}		V_{err}	
	vs. i_c	vs. v_{cp}	vs. i_c	vs. v_{cp}
1	9.6%	9.6%	8.3%	8.3%
2	5.0%	5.0%	4.1%	4.1%
10	2.1%	2.1%	2.1%	2.1%
25	1.3%	1.3%	2.0%	2.0%
50	1.0%	1.0%	2.0%	2.0%

TABLE IV
RMS ERROR OVER ONE PERIOD OF RECONSTRUCTED BUCK WAVEFORM RIPPLES VS. COMPLETE WAVEFORM RIPPLE WITH IDEAL AND NON-IDEAL SWITCHES

Number of Harmonics N	V_{out}		I_L	
	non-ideal switches	ideal switches	non-ideal switches	ideal switches
1	9.6%	10.1%	8.7%	8.9%
2	5.0%	5.6%	2.5%	3.2%
10	2.1%	3.0%	0.4%	2.4%
25	1.3%	3.0%	0.1%	2.4%
50	1.0%	3.1%	0.1%	2.4%

above 3% with ideal switches even for a large number of harmonics. For I_L , the error reaches 0.1% with switch voltage drops and remains above 2% with ideal switches.

The above described RMS error is an important indicator of the method's accuracy. It shows the accuracy of the reconstructed waveform compared to the transistor-level waveform. Properties like the peak-to-peak ripple magnitude and the harmonic distortion can be extracted. The former is a common performance figure for DC-DC converters. Table V shows the accuracy of the peak-to-peak ripple magnitude of the reconstructed waveform relative to the transistor-level waveform, calculated as:

$$Error_{ripple} = \left| \frac{X_{out,pp} - X_{out,N,pp}}{X_{out,pp}} \right|, \quad (14)$$

where $X_{out,pp}$ and $X_{out,N,pp}$ are the peak-to-peak ripple magnitudes for the transistor-level waveform and the waveform reconstructed with N harmonics, respectively. For V_{out} , the error is below 10% at 25 harmonics and 5% at 50 harmonics. For I_L , the error is below 2% at 25 harmonics and below 1% at 50 harmonics. As for the RMS error, the accuracy is improved (especially for I_L) by considering the switch voltage drops. The proposed method also has the capability to account for arbitrary parasitic elements with no additional modeling effort. Fig. 1 shows how considering the parasitic inductance of C_{out} provides a more accurate peak-to-peak ripple.

Table VI presents the runtime comparison between the transistor-level model simulation and the averaged model simulation with ripple reconstruction based on Fourier series. The transient analysis runtime is negligible for the latter (1s) compared to the former (443s). The open-loop AC analysis runtime is negligible as well. With the averaged model, additional time is needed to calculate the ripple, time which increases with the desired accuracy, namely with the number of harmonics. The choice of the number of harmonics is determined by the trade-off between accuracy and computational effort.

In this Buck converter example, the ripple waveforms are generated with a time resolution of 1ns, meaning 1/400 of

TABLE V
PERCENTAGE ERROR OF PEAK-TO-PEAK RIPPLE MAGNITUDE OF RECONSTRUCTED WAVEFORM VS. COMPLETE WAVEFORM WITH IDEAL AND NON-IDEAL SWITCHES

Number of Harmonics N	V_{out}		I_L	
	non-ideal switches	ideal switches	non-ideal switches	ideal switches
1	35%	38%	22%	26%
2	30%	32%	11%	14%
10	19%	21%	4.0%	7.2%
25	9%	12%	1.5%	5.1%
50	5%	7%	0.9%	4.5%

TABLE VI
RUNTIME COMPARISON - BUCK CONVERTER

Model	Number of harmonics	Simulation time	Ripple calculation	Total time	Speedup
transistor-level	N/A	443s (TRAN)	N/A	443s	1x
averaged	N=1	1s (TRAN+AC)	1s	2s	220x
	N=2		1s	2s	220x
	N=10		2s	3s	150x
	N=25		12s	13s	35x
	N=50		45s	46s	10x

a period. By reconstructing the output voltage V_{out} with 25 harmonics, a 35x speedup is achieved, while having an accuracy loss of only 1.3%. With 50 harmonics, the speedup is still 10x and the accuracy loss is only 1%.

The results from Tables III and VI, discussed in the previous paragraphs, are graphically illustrated in Fig. 12, showing for the Buck converter the trade-off between accuracy (measured by the RMS error) and computational effort (measured by the CPU runtime), both in relation to the number of harmonics.

Practical integrated circuits need to account for robustness in the design and verification process. Monte-Carlo analysis is a common method, applied for various circuits, including DC-DC converters [24], [25]. Alternatively, deterministic approaches can be used, such as the worst-case distance approach. Both methods require a large number of simulations (typically, hundreds to thousands). Circuit optimization is another situation requiring a large number of simulations [26]. These methods would benefit significantly from the speedup of the individual simulations of DC-DC converters.

Consider the Monte-Carlo analysis of the Buck converter example from this section. The output voltage peak-to-peak ripple is a common performance figure. The averaged model produces no ripple information, while transistor-level simulation is computationally expensive. The method presented in this paper provides a faster way to obtain the ripple information. In this example, the number of harmonics to reconstruct the ripple waveform for the output voltage can be chosen as 25, as this provides a good trade-off between accuracy and speed, according to Tables III and VI and Fig. 12. For a Monte-Carlo analysis with 100 simulations, the total runtime decreases from approximately 12h30min in the transistor-level approach to approximately 20min with the proposed method.

B. Boost Converter

The second example is an asynchronous Boost converter, shown in Fig. 3, operating in CCM and with current mode control of the duty cycle. The converter produces an output

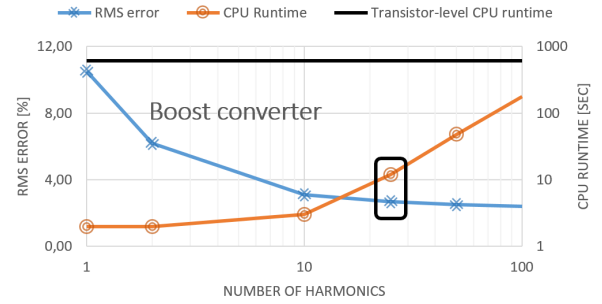


Fig. 12. Trade-off between accuracy (measured by the RMS error) and computational effort (measured by the CPU runtime), both in relation to the number of harmonics, for the Buck converter. A typical trade-off between accuracy and CPU effort is marked at 25 harmonics.

TABLE VII
RMS ERROR OVER ONE PERIOD OF RECONSTRUCTED BOOST WAVEFORM RIPPLES VS. COMPLETE WAVEFORM RIPPLE

Number of Harmonics N	V_{out}		V_{err}	
	vs. i_c	vs. i_p	vs. i_c	vs. i_p
1	10.5%	10.5%	9.7%	9.7%
2	6.2%	6.2%	3.9%	3.9%
10	3.3%	3.1%	1.1%	0.7%
25	2.9%	2.7%	1.1%	0.6%
50	2.7%	2.5%	1.1%	0.6%

voltage $V_{out} = 19V$ at a load current $I_{load} = 400mA$ from an input voltage $V_{in} = 4V$ (which is also the supply voltage of the entire circuit). The switching frequency is $F_{sw} = 500kHz$, equivalent to a switching period $T = 2\mu s$. The component values are:

- Inductance $L = 10\mu H$, with parasitic equivalent series resistance $R_L = 50m\Omega$,
- Output capacitance $C_{out} = 10\mu F$, with parasitic equivalent series resistance $ESR = 10m\Omega$. The equivalent series inductance ESL is not considered in this case, as it was not provided in the available output capacitor model.

Like the previous Buck converter, the Boost converter is implemented at transistor level, in the same technology.

The method for generating the output waveforms, described in the previous subsection for the Buck converter, is applied in this subsection to the Boost converter as well. Figs. 13 and 14 display the transistor-level and the reconstructed waveform ripples for the output voltage V_{out} and for the error voltage V_{err} , respectively. Then, Table VII shows quantitative results for the waveform reconstruction accuracy. Finally, Table VIII illustrates the runtime performance of the proposed method.

The waveform ripples are reconstructed based on the Fourier series expansion of the PWM switch cell signals, by referring to i_c and i_p , respectively. The RMS errors presented in Table VII for V_{out} and V_{err} and calculated in the same way as for the Buck converter show that referring to i_p provides slightly more accurate results. The reason is that, on the signal path, i_p is closer to the reconstructed signals. In the case of i_c or v_{cp} , additional distortion is introduced by the nonlinear behavior of the PWM switch cell model.

Fig. 13 shows the ripple of the output voltage V_{out} , reconstructed with a gradually increasing number of harmonics: 10, 25 and 50, respectively. With 25 harmonics, the generated waveform begins to look similar to the transistor-level wave-

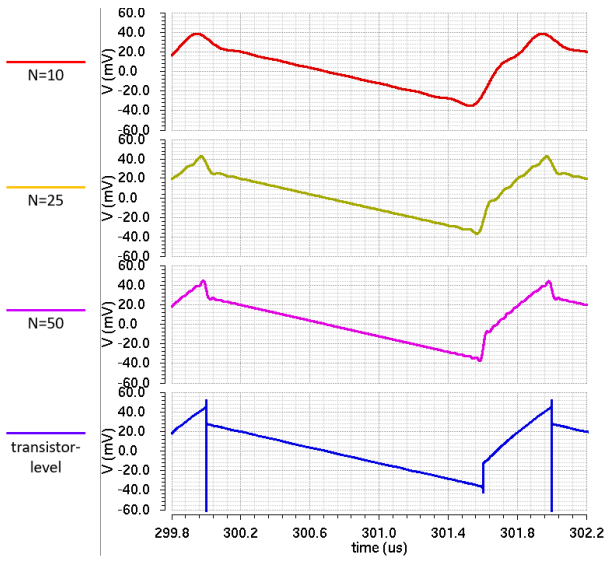


Fig. 13. Output voltage (V_{out}) ripple (total voltage minus average value) for the Boost converter. Reconstructed waveforms, from the averaged-model simulation, for 10, 25 and 50 harmonics, respectively. At the bottom, the waveform resulted from transistor-level simulation.

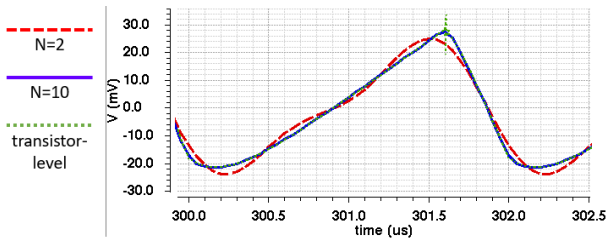


Fig. 14. Error voltage (V_{err}) ripple (total voltage minus average value) for the Boost converter. Reconstructed waveforms, from the averaged-model simulation, with 2 and 10 harmonics, respectively, and the waveform resulted from transistor-level simulation.

form. This can be seen also from the RMS error, which drops below 3% for 25 harmonics, as shown in Table VII. With 50 harmonics, the reconstructed waveform looks very close to the transistor-level one and the error drops to 2.5%. Even with 50 harmonics, the error is above 2%. There is a systematic error caused by the negative spike from the transistor-level waveform, as seen in Fig. 13. To account for such a spike, an impractically large number of harmonics would be needed.

Fig. 14 shows the waveform ripple for the error voltage V_{err} obtained from the transistor-level simulation and the waveform obtained using the proposed model with 2 and 10 harmonics, respectively. With 2 harmonics, there is a visible difference to the transistor-level waveforms, indicated also by the RMS error close to 2%, as shown in Table VII. With 10 harmonics, the reconstructed waveform almost overlaps the transistor-level one. The RMS error is in this case 0.6%. With further increases in the number of harmonics, the accuracy does not improve anymore, as seen in Table VII.

The numerical results of the runtime performance are summarized in Table VIII. For a fair comparison between the transistor-level simulation and the method proposed in this paper and for a fair comparison with the Buck converter from the previous subsection, a transient analysis over 2500 periods

TABLE VIII
RUNTIME COMPARISON - BOOST CONVERTER

Model	Number of harmonics	Simulation time	Ripple calculation	Total time	Speedup
transistor-level	N/A	607s (TRAN)	N/A	607s	1x
averaged	N=1	1s (TRAN+AC)	1s	2s	300x
	N=2		1s	2s	300x
	N=10		2s	3s	200x
	N=25		11s	12s	50x
	N=50		47s	48s	13x

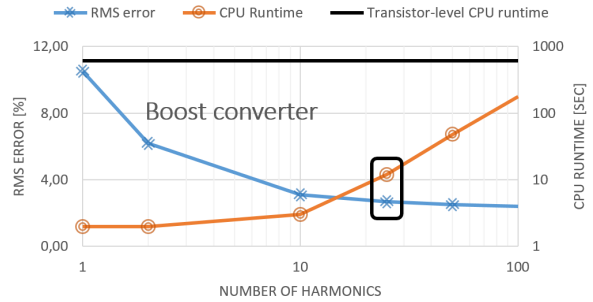


Fig. 15. Trade-off between accuracy (measured by the RMS error) and computational effort (measured by the CPU runtime), both in relation to the number of harmonics, for the Boost converter. A typical trade-off between accuracy and CPU effort is marked at 25 harmonics.

is applied to the Boost converter. As the switching frequency is 500kHz, the simulation time is 5ms. Also, the ripples are reconstructed with a resolution of 1/400 of a period, of 5ns.

In this Boost converter example, the CPU runtime of the transistor-level simulation is 607s and 1.7M points are simulated. On the other hand, the transient analysis with the averaged model has a negligible runtime of 1s. Only 144 points are simulated, meaning more than 10k less than in the transistor-level transient analysis. The open-loop AC analysis runtime, for getting the small signal information, is also negligible. However, time is needed to reconstruct the waveform ripple, time which increases with the number of harmonics. For V_{out} , the loss of accuracy for the reconstructed waveform with 25 harmonics is below 3%, according to Table VII, while achieving a 50x speedup compared to the transistor-level simulation, as shown in Table VIII. With 50 harmonics, the accuracy is improved, as the error decreases to 2.5%, but the speedup also decreases to 13x.

The results from Tables VII and VIII are graphically illustrated in Fig. 15, showing for the Buck converter the trade-off between accuracy (measured by the RMS error) and computational effort (measured by the CPU runtime), both in relation to the number of harmonics.

Monte-Carlo analysis is a situation where the method proposed in this paper brings a significant CPU time reduction. According to the results from Tables VII and VIII and Fig. 15, with 25 harmonics a good trade-off can be achieved when reconstructing the ripple waveform for the output voltage. With this choice, the total runtime for 100 simulations decreases from approximately 18h30min in the transistor-level approach to approximately 20min with the proposed method.

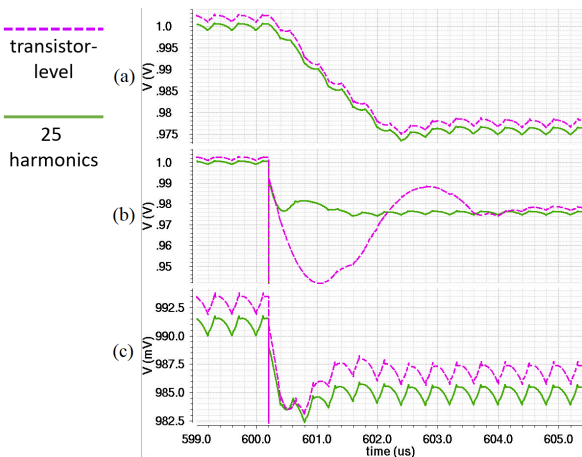


Fig. 16. Transient behavior of the Buck converter for load current variation: comparison between the transistor-level waveform and the waveform reconstructed with 25 harmonics. The load current variation is: (a) slow and large, (b) fast and large, (c) fast and small.

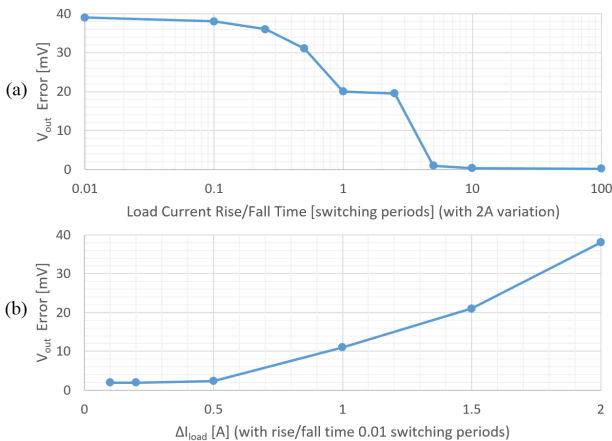


Fig. 17. Accuracy of proposed method vs. transistor-level simulation in dynamic conditions: maximum error magnitude (minus the systematic error from steady state) in simulating the Buck converter V_{out} with respect to: (a) rise/fall time, (b) magnitude of load current step, respectively.

C. Dynamic Behavior

The previous experimental results have demonstrated the method's performance in steady-state conditions. However, the method is also applicable in dynamic conditions, e.g., in the case of a load step transient. The following results will illustrate the method's performance and limitation in such conditions: a good accuracy is obtained for slow and/or small variations of the load current (in other words, in quasi-steady-state conditions).

Fig. 16 shows the output voltage V_{out} of the Buck converter. In three load current variation scenarios, the waveform simulated at transistor-level and the waveform reconstructed with 25 harmonics are compared. The results are illustrated for increasing load current. For decreasing load current, very similar results are obtained.

First, Fig. 16 (a) shows the behaviour for a large and slow load variation. The load increases from 1A to 3A in a time equal to 5 switching periods. The reconstructed waveform reproduces the transistor-level waveform with good accuracy. The plot shows a systematic error of approximately 2mV. This

error is caused by the limitations of the equivalent averaged model of the PWM switch cell [7] and not by the ripple reconstruction. Moreover, this error is more than 10x smaller than the V_{out} variation due to load regulation (approximately 2mV for load currents from 1A to 3A). The variation of V_{out} with the operating conditions (e.g. load current) is a consequence of the limited loop gain (46dB or 200) of this Buck converter implementation. The Boost converter presented in this paper uses a different compensation method for the error amplifier and has a loop gain of 100dB (or 100k). This makes the V_{out} regulation much more precise and is the reason why the averaged model does not suffer from the steady-state error in the Boost converter. Therefore, the steady-state error is highly dependent on the circuit topology.

Second, in the scenario shown in Fig. 16 (b), the load current variation is large and fast. The load increases from 1A to 3A in a time much smaller than the switching period. In this case, the error is significant during the settling, reaching up to 40mV in magnitude. The error is again mainly caused by the limitations of the averaged PWM switch cell model. In general, large-signal averaged models do not work well under dynamic conditions with large and sudden variations, since averaged models use an “averaged and continuous” representation of the original function over several periods [7], [27].

Third, in Fig. 16 (c) there is again a fast change in the load, but this time the current variation is smaller (0.5A instead of 2A). The method shows here a good accuracy, with an error magnitude within 2mV.

Fig. 17 quantitatively describes the meaning of “slow variations” and “small variations” for the Buck converter example. For a fair comparison, the systematic steady-state error (approximately 2mV) is subtracted from the total error. First, Fig. 17 (a) shows that, for the maximum considered current variation of 2A, the proposed method shows a good accuracy for rise/fall times starting from 5 switching periods. Second, Fig. 17 (b) shows that, for fast load variations (the rise/fall time is much smaller than the switching period), a good accuracy is obtained for load steps up to 0.5A.

The theory of the method was developed in steady-state conditions. However, it holds also under dynamic behavior, in quasi-steady-state conditions.

D. Stability

As most practical DC-DC converters include control loops, stability needs to be accounted for in the design process. In current-mode controlled DC-DC converters, subharmonic stability is a major issue. This is the reason why a compensation ramp is often needed in such converter designs [7].

The presented method is able to capture subharmonic oscillations through the averaged model. The state-of-the-art averaged model includes the capacitor C_s , introduced specifically for this purpose, as described in Section III-B. When present, oscillations are revealed in the frequency response, as well as in the transient waveforms. This is illustrated in the following for the Buck converter example.

First, the frequency response is compared for the converter with and without compensation ramp in Fig. 18. When no such ramp is applied, the magnitude shows, at $F_{sw}/2 = 1.25\text{MHz}$, a peak specific to subharmonic oscillations [7], which raises the gain above 0dB at frequencies around $F_{sw}/2$. The unity

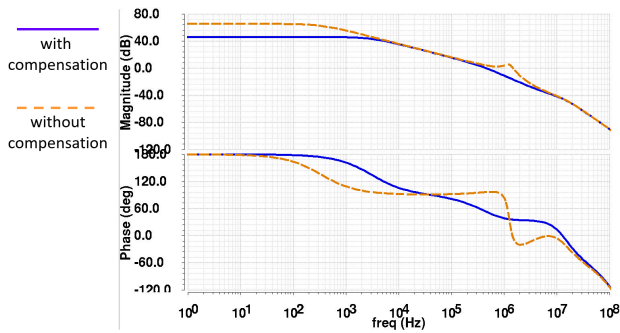


Fig. 18. Frequency response of Buck converter with and without compensation ramp. The circuit with compensation ramp has a loop gain of 46dB.

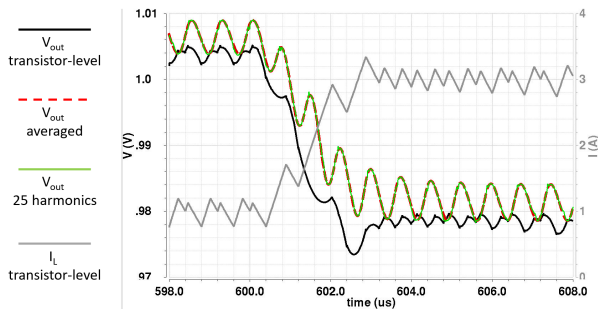


Fig. 19. Transient waveforms of Buck converter with subharmonic oscillations.

gain frequency is about 1.5MHz. The phase margin of -5° reveals instability. Adding the compensation ramp damps the peak and so removes the subharmonic oscillations. The unity-gain frequency decreases to 450kHz and the phase margin of 53° indicates a stable loop. In this way, the averaged model, having the compensation ramp S_a as a parameter (Fig. 5), can be used to design the compensation ramp of the circuit.

Second, the subharmonic oscillations are captured in the transient waveforms (of the converter without compensation ramp), in steady-state as well as in dynamic conditions, as shown in Fig. 19. The output voltage obtained with the averaged model oscillates with the expected frequency, $F_{sw}/2 = 1.25\text{MHz}$, revealing the instability of the circuit, in agreement with the transistor-level waveforms, which show as well subharmonic oscillations. The limitation of the proposed method is the inability to correctly predict the amplitude and the shape of the waveforms. The averaged model produces an oscillating instead of a constant (in steady state) waveform (as in stable conditions). For V_{out} , applying the Fourier-series-based reconstruction produces a waveform nearly overlapping the index-0 waveform resulted from the averaged model, because the amplitude of the subharmonic sine wave is much larger than the harmonic ripple.

VII. CONCLUSION

A multi-harmonic modeling and simulation technique for low-power PWM DC-DC converters was presented. The technique is based on the large-signal averaged modeling of the PWM switch cell and the Fourier series expansion of the voltages and currents at the switch cell terminals. The distinguishing feature of this technique is generality. With no

additional modeling effort, it is applicable to a wide range of DC-DC converters and it supports any parasitic elements, any surrounding circuitry and any number of harmonics. Circuits for dynamic characterization of the PWM switch cell parameters were presented as well and brought together with the PWM switch cell averaged model and the surrounding circuitry into the same testbench. The proposed method was illustrated for a Buck and for a Boost converter and achieved a speedup of one order of magnitude over the transistor-level model, with a loss of accuracy below 3%. The method works with a good accuracy in steady-state, as well as in dynamic conditions with slow and/or small variations. The method correctly predicts subharmonic instability.

REFERENCES

- [1] R. D. Middlebrook and S. Čuk, "A general unified approach to modelling switching-converter power stages," *International Journal of Electronics Theoretical and Experimental*, vol. 42, no. 6, pp. 521–550, 1977.
- [2] A. Davoudi, J. Jatskevich, and T. De Rybel, "Numerical state-space average-value modeling of pwm dc-dc converters operating in dcm and ccm," *IEEE Trans. on Power Electronics*, vol. 21, no. 4, pp. 1003–1012, 2006.
- [3] J. Sun, D. M. Mitchell, M. F. Greuel, P. T. Krein, and R. M. Bass, "Averaged modeling of pwm converters operating in discontinuous conduction mode," *IEEE Trans. on Power Electronics*, vol. 16, no. 4, pp. 482–492, 2001.
- [4] V. Vorperian, "Simplified analysis of PWM converters using model of PWM switch part I: Continuous conduction mode," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 26, no. 3, pp. 490–496, 1990.
- [5] V. Vorperian, "Simplified analysis of pwm converters using model of pwm switch. ii. discontinuous conduction mode," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 26, no. 3, pp. 497–505, 1990.
- [6] Y. Amran, F. Huliehel, and S. Ben-Yaakov, "A unified SPICE compatible average model of PWM converters," *IEEE Trans. on Power Electronics*, pp. 585–694, 1991.
- [7] C. Basso, *Switch-Mode Power Supplies*. McGraw-Hill Education, 2014.
- [8] D. Tannir, Y. Wang, and P. Li, "Accurate modeling of nonideal low-power pwm dc-dc converters operating in ccm and dcm using enhanced circuit averaging techniques," *ACM Trans. on Design Automation of Electronic Systems*, vol. 21, no. 4, pp. 1–15, 2016.
- [9] S. R. Sanders, J. M. Noworolski, X. Z. Liu, and G. C. Verghese, "Generalized averaging method for power conversion circuits," *IEEE Trans. on Power Electronics*, vol. 6, no. 2, pp. 251–259, 1991.
- [10] V. A. Caliskan, O. Verghese, and A. M. Stankovic, "Multifrequency averaging of dc/dc converters," *IEEE Transactions on Power Electronics*, vol. 14, no. 1, pp. 124–133, 1999.
- [11] D. Maksimović, A. M. Stanković, V. J. Thottuvelil, and G. C. Verghese, "Modeling and simulation of power electronic converters," *Proceedings of the IEEE*, vol. 89, no. 6, pp. 898–912, 2001.
- [12] J. M. Noworolski and S. R. Sanders, "Generalized in-plane circuit averaging," in *Applied Power Electronics Conference and Exposition, 1991. APEC'91. Conference Proceedings, 1991., Sixth Annual*. IEEE, 1991, pp. 445–451.
- [13] Y. Wang, D. Gao, D. Tannir, and P. Li, "Multi-harmonic nonlinear modeling of low-power pwm dc-dc converters operating in ccm and dcm," in *Proceedings of the 2016 Design, Automation & Test in Europe Conference & Exhibition*. EDA Consortium, 2016.
- [14] Y. Wang, D. Gao, D. Tannir, N. Dong, P. Fang, W. Dong, and P. Li, "Multi-harmonic small-signal modeling of low-power pwm dc-dc converters," *ACM Trans. on Design Automation of Electronic Systems*, vol. 22, no. 4, pp. 1–16, 2017.
- [15] F. Misoc, M. M. Morcos, and J. Lookadoo, "Fourier-series models of dc-dc converters," in *Proceedings of the North American Power Symposium*, 2006, pp. 193–199.
- [16] R. Trincherio, P. Manfredi, I. S. Stievano, and F. G. Canavero, "Steady-state analysis of switching converters via frequency-domain circuit equivalents," *IEEE Trans. on Circuits and Systems—II*, vol. 63, no. 8, pp. 748–752, 2016.
- [17] E. Setiawan, T. Hirata, and I. Hodaka, "Accurate symbolic steady state modeling of buck converter," *International Journal of Electrical and Computer Engineering*, vol. 7, no. 5, pp. 2374–2381, 2017.
- [18] M. Forouzesh, Y. Siwakoti, S. A. Gorji, F. Blaabjerg, and B. Lehman, "Step-up dc-dc converters: A comprehensive review of voltage-boosting techniques, topologies, and applications," *IEEE Trans. on Power Electronics*, vol. 32, no. 12, pp. 9143–9177, 2017.

- [19] R. Trincherro, I. S. Stievano, and F. G. Canavero, "Steady-state analysis of switching power converters via augmented time-invariant equivalents," *IEEE Trans. on Power Electronics*, vol. 29, no. 11, pp. 5657–5661, 2014.
- [20] J. W. van der Woude, W. L. de Koning, and Y. Fuad, "On the periodic behavior of pwm dc-dc converters," *IEEE Trans. on Power Electronics*, vol. 17, no. 4, pp. 585–595, July 2002.
- [21] Y. Yan, F. C. Lee, and P. Mattavelli, "Unified three-terminal switch model for current mode controls," *IEEE Trans. on Power Electronics*, vol. 27, no. 9, pp. 4060–4070, Sep. 2012.
- [22] *SIMetrix/SIMPLIS – User's Manual*, Simetrix Technologies Ltd., 2015.
- [23] S. Tian, F. C. Lee, J. Li, Q. Li, and P. Liu, "A three-terminal switch model of constant on-time current mode with external ramp compensation," *IEEE Trans. on Power Electronics*, vol. 31, no. 10, pp. 7311–7319, Oct 2016.
- [24] O. A. Beg *et al.*, "Model validation of pwm dc-dc converters," *IEEE Trans. on Industrial Electronics*, vol. 64, no. 9, pp. 7049–7059, 2017.
- [25] A. G. O. Dela-Cruz *et al.*, "Cmos implementation of hysteretic controller for a dc-to-dc buck converter using 0.35um library," in *DLSU Research Congress*. De La Salle University, Manila, Philippines, 2018.
- [26] T. C. Neugebauer and D. J. Perreault, "Computer-aided optimization of dc/dc converters for automotive applications," *IEEE Trans. on Power Electronics*, vol. 18, no. 3, pp. 775–783, May 2003.
- [27] G. Migoni, M. E. Romero, F. Bergero, and E. Kofman, "A mixed modeling approach for efficient simulation of pwm switching mode power supplies," *IEEE Trans. on Power Electronics*, vol. 34, no. 10, pp. 9758–9767, Oct 2019.



Florin Burcea received the B.Sc. degree in electronics and telecommunication engineering from University Politehnica of Bucharest (UPB) in 2013 and the M.Sc. degree in communications engineering from Technical University of Munich (TUM) in 2015. He is currently working towards PhD with the Chair of Electronic Design Automation, TUM, focusing on the design methodology and design automation of analog circuits.

In 2013, he won the first place in the Analog Integrated Circuits section of the Annual National Contest "Tudor Tănăsescu" in Romania. In 2015, 2016 and 2017 he won the first place in the Synopsys Annual International Microelectronics Olympiad of Armenia. Both contests are held in partnership with IEEE. In 2016, he received the Award of the Association for Electrical, Electronic and Information Technologies (VDE) for his Master Thesis.



Dani Tannir (S'02–M'12–SM'19) received the Bachelor of Engineering degree, with distinction, in Electrical Engineering from the American University of Beirut in 2004. He then received the Master and Ph.D. degrees in Electrical Engineering from McGill University in Montreal, Canada in 2006 and in 2010, respectively.

He was named on the Dean's honor list for his Master thesis work and was the recipient of several awards during his Ph.D. studies, including the Alexander Graham Bell Canada Graduate Scholarship from the Natural Sciences and Engineering Research Council of Canada.

In 2011, Prof. Tannir joined the Department of Electrical and Computer Engineering at the Lebanese American University in Byblos, Lebanon where he is currently an Associate Professor. Prof. Tannir also worked as a Fulbright visiting research scholar at Texas A&M University in College Station, Texas in 2014, and as a DAAD visiting research scholar at the Technical University of Munich, Germany in 2018.

His research area of expertise focuses on the development of efficient simulation and modeling techniques for nonlinear analog and radio frequency integrated circuits.



Helmut E. Graeb (M'02–SM'03–F'14) received the Dipl.-Ing., Dr.-Ing., and Habilitation degrees in electrical engineering from the Technical University of Munich (TUM), Munich, Germany, in 1986, 1993 and 2008, respectively.

He was with Siemens Corporation, Munich, from 1986 to 1987, where he was involved in the design of DRAMs. Since 1987, he has been with the Chair of Electronic Design Automation, TUM, where he has been the Head of a research group since 1993.

He has published around 190 papers, six of which were nominated for best papers at the Design Automation Conference (DAC), the International Conference on Computer-Aided Design (ICCAD), and the Design, Automation and Test in Europe (DATE) conference.

His current research interests include design automation for analog and mixed-signal circuits, with particular emphasis on Pareto optimization of analog circuits considering parameter tolerances, analog design for yield and reliability, hierarchical sizing of analog circuits, analog/mixed-signal test design, discrete sizing of analog circuits, structural analysis of analog and digital circuits, and analog layout.

Dr. Graeb was a recipient of the 2008 Prize of the Information Technology Society (ITG), the 2004 Best Teaching Award of the TUM EE Faculty Students Association, and the Third Prize of the 1996 Munich Business Plan Contest. He has served as the Vice-President Publications of the IEEE Council on Electronic Design Automation, an Executive Committee Member of the ICCAD, a member or the Chair of the Analog Program Subcommittees of the ICCAD, DAC, and DATE conferences, an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—PART II: ANALOG AND DIGITAL SIGNAL PROCESSING and the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, and a member of the Technical Advisory Board of MunEDA GmbH Munich, which he co-founded. He is a member of VDE ITG and VDE/VDI-Society Microelectronics, Microsystems, and Precision Engineering.

IEEE COPYRIGHT FORM

To ensure uniformity of treatment among all contributors, other forms may not be substituted for this form, nor may any wording of the form be changed. This form is intended for original material submitted to the IEEE and must accompany any such material in order to be published by the IEEE. Please read the form carefully and keep a copy for your files.

Fast SPICE-Compatible Simulation of Low-Power On-Chip PWM DC-DC Converters with Improved Ripple Accuracy

Burcea, Florin; Tannir, Dani; Graeb, Helmut

IEEE Transactions on Power Electronics

COPYRIGHT TRANSFER

The undersigned hereby assigns to The Institute of Electrical and Electronics Engineers, Incorporated (the "IEEE") all rights under copyright that may exist in and to: (a) the Work, including any revised or expanded derivative works submitted to the IEEE by the undersigned based on the Work; and (b) any associated written or multimedia components or other enhancements accompanying the Work.

GENERAL TERMS

1. The undersigned represents that he/she has the power and authority to make and execute this form.
2. The undersigned agrees to indemnify and hold harmless the IEEE from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.
3. The undersigned agrees that publication with IEEE is subject to the policies and procedures of the [IEEE PSPB Operations Manual](#).
4. In the event the above work is not accepted and published by the IEEE or is withdrawn by the author(s) before acceptance by the IEEE, the foregoing copyright transfer shall be null and void. In this case, IEEE will retain a copy of the manuscript for internal administrative/record-keeping purposes.
5. For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
6. The author hereby warrants that the Work and Presentation (collectively, the "Materials") are original and that he/she is the author of the Materials. To the extent the Materials incorporate text passages, figures, data or other material from the works of others, the author has obtained any necessary permissions. Where necessary, the author has obtained all third party permissions and consents to grant the license above and has provided copies of such permissions and consents to IEEE

BY TYPING IN YOUR FULL NAME BELOW AND CLICKING THE SUBMIT BUTTON, YOU CERTIFY THAT SUCH ACTION CONSTITUTES YOUR ELECTRONIC SIGNATURE TO THIS FORM IN ACCORDANCE WITH UNITED STATES LAW, WHICH AUTHORIZES ELECTRONIC SIGNATURE BY AUTHENTICATED REQUEST FROM A USER OVER THE INTERNET AS A VALID SUBSTITUTE FOR A WRITTEN SIGNATURE.

Florin Burcea

Signature

18-12-2019

Date (dd-mm-yyyy)

Information for Authors

AUTHOR RESPONSIBILITIES

The IEEE distributes its technical publications throughout the world and wants to ensure that the material submitted to its publications is properly available to the readership of those publications. Authors must ensure that their Work meets the requirements as stated in section 8.2.1 of the IEEE PSPB Operations Manual, including provisions covering originality,

authorship, author responsibilities and author misconduct. More information on IEEE's publishing policies may be found at http://www.ieee.org/publications_standards/publications/rights/authorrightsresponsibilities.html Authors are advised especially of IEEE PSPB Operations Manual section 8.2.1.B12: "It is the responsibility of the authors, not the IEEE, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it." Authors are also advised of IEEE PSPB Operations Manual section 8.1.1B: "Statements and opinions given in work published by the IEEE are the expression of the authors."

RETAINED RIGHTS/TERMS AND CONDITIONS

- Authors/employers retain all proprietary rights in any process, procedure, or article of manufacture described in the Work.
- Authors/employers may reproduce or authorize others to reproduce the Work, material extracted verbatim from the Work, or derivative works for the author's personal use or for company use, provided that the source and the IEEE copyright notice are indicated, the copies are not used in any way that implies IEEE endorsement of a product or service of any employer, and the copies themselves are not offered for sale.
- Although authors are permitted to re-use all or portions of the Work in other works, this does not include granting third-party requests for reprinting, republishing, or other types of re-use. The IEEE Intellectual Property Rights office must handle all such third-party requests.
- Authors whose work was performed under a grant from a government funding agency are free to fulfill any deposit mandates from that funding agency.

AUTHOR ONLINE USE

- **Personal Servers.** Authors and/or their employers shall have the right to post the accepted version of IEEE-copyrighted articles on their own personal servers or the servers of their institutions or employers without permission from IEEE, provided that the posted version includes a prominently displayed IEEE copyright notice and, when published, a full citation to the original IEEE publication, including a link to the article abstract in IEEE Xplore. Authors shall not post the final, published versions of their papers.
- **Classroom or Internal Training Use.** An author is expressly permitted to post any portion of the accepted version of his/her own IEEE-copyrighted articles on the author's personal web site or the servers of the author's institution or company in connection with the author's teaching, training, or work responsibilities, provided that the appropriate copyright, credit, and reuse notices appear prominently with the posted material. Examples of permitted uses are lecture materials, course packs, e-reserves, conference presentations, or in-house training courses.
- **Electronic Preprints.** Before submitting an article to an IEEE publication, authors frequently post their manuscripts to their own web site, their employer's site, or to another server that invites constructive comment from colleagues. Upon submission of an article to IEEE, an author is required to transfer copyright in the article to IEEE, and the author must update any previously posted version of the article with a prominently displayed IEEE copyright notice. Upon publication of an article by the IEEE, the author must replace any previously posted electronic versions of the article with either (1) the full citation to the IEEE work with a Digital Object Identifier (DOI) or link to the article abstract in IEEE Xplore, or (2) the accepted version only (not the IEEE-published version), including the IEEE copyright notice and full citation, with a link to the final, published article in IEEE Xplore.

Questions about the submission of the form or manuscript must be sent to the publication's editor.

Please direct all questions about IEEE copyright policy to:

IEEE Intellectual Property Rights Office, copyrights@ieee.org, +1-732-562-3966

3 Modeling for MEMS-IC Sizing

This chapter presents a complete approach to MEMS-IC robustness co-optimization. All parameters (design, process and operating) of the electrical and the mechanical part are included in the optimization. The optimization methodology is illustrated on two demonstrative examples: a MEMS microphone and a MEMS accelerometer, each with an integrated readout circuit. The optimization succeeds only when simultaneously including mechanical and electrical design parameters. The robustness is strongly influenced by the mechanical process parameters.

A simulation-in-a-loop flow is implemented, based on the commercial tools WiCkeD (MunEDA) for optimization, Spectre (Cadence) for continuous electrical simulation and MEMS+ (Coventor) for MEMS simulation. Within this flow, a reduced-order model, exported from MEMS+ and described in VerilogA, is used for the MEMS. Using this model, a CPU time reduction of up to 4 times is achieved. A contribution to the CPU time reduction is also brought by exporting the reduced-order model only when necessary. Any newly exported MEMS model is saved in a database and retrieved from there when needed again, maximizing in this way the efficiency of the optimization flow. A control flow for accessing the database is implemented to support parallel simulation.

Mathematical methods of diagnosis are transferred from the IC level to the extended MEMS-IC level and applied to the MEMS accelerometer. The mixed mechanical and electrical influence on the improvement of the worst-case distances and on the overall yield is illustrated.

This chapter consists of the following publication:

- F. Burcea, A. Herrmann, B. Li, and H. Graeb. 2019. "MEMS-IC Robustness Optimization Considering Electrical and Mechanical Design and Process Parameters". ACM Trans. Des. Autom. Electron. Syst. 24, 4, Article 43 (June 2019), 24 pages.

The paper was written under the leading pen of the candidate. The candidate conceptualized the paper and was responsible for the literature review. The candidate brought the following technical contributions:

- implementation and optimization of the two circuit examples,
- joint contribution in the development of the scripted co-optimization flow,
- implementation and application of the diagnosis methods to the MEMS and IC parts together,
- obtaining of the experimental results.

3 Modeling for MEMS-IC Sizing

The following publications, although not explicitly included in the thesis, are part of the same work:

- F. Burcea, A. Herrmann, A. Gupta and H. Graeb, "A New Robustness Optimization Methodology for MEMS-IC Systems," 2017 14th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), Giardini Naxos, 2017, pp. 1-4.
- F. Burcea, A. Herrmann, B. Li and H. Graeb, "MEMS-IC Optimization Considering Design Parameters and Manufacturing Variation from both Mechanical and Electrical Side," 2018 25th IEEE International Conference on Electronics, Circuits and Systems (ICECS), Bordeaux, 2018, pp. 625-628.
- F. Burcea, A. Herrmann and H. Graeb, "Towards MEMS-IC Robustness Optimization", Frontiers in Analog Circuit (FAC) Synthesis and Verification," July 2017.
- F. Burcea, A. Herrmann, B. Li and H. Graeb, "MEMS-IC Yield Optimization with Electrical and Mechanical Process Parameters" CDNLive, May 2019.
- F. Burcea, A. Herrmann, B. Li and H. Graeb, "Robustness Optimization of MEMS-based Sensor Circuits" VDE/VDI-GMM MikroSystemTechnik Kongress, October 2019.

MEMS-IC Robustness Optimization Considering Electrical and Mechanical Design and Process Parameters

FLORIN BURCEA, ANDREAS HERRMANN, BING LI, and HELMUT GRAEB, Technical University of Munich, Chair of Electronic Design Automation, Germany

MEMS-based sensor circuits are traditionally designed separately using CAD tools specific to each energy domain (electrical and mechanical). The paper presents a complete approach for combined MEMS-IC robustness optimization. Advanced methods for robustness analysis and optimization considering design, operating and process parameters, developed for integrated circuits, are transferred to MEMS-IC systems. Both electrical and mechanical design and process parameters are included in the optimization. The methodology is exemplified on two demonstrator examples: a MEMS microphone and a MEMS accelerometer, each with an integrated readout circuit. A successful optimization requires the simultaneous inclusion of design parameters and process tolerances from both energy domains. To save CPU time, a reduced-order, circuit-level model is used for the MEMS part and this model is created only when necessary. To integrate the generation of the simplified model into the optimization flow, a simulation-in-a-loop flow based on commercial tools for both the electrical and the mechanical domain has been implemented.

CCS Concepts: • **Computing methodologies** → **Modeling and simulation**; • **Hardware** → Electronic design automation.

Additional Key Words and Phrases: MEMS-CMOS, robustness optimization, diagnosis, circuit-level model, parallel simulation

1 INTRODUCTION

In the robustness design of sensor circuits, usually Monte-Carlo and/or corner-case simulation are applied. The former technique serves the analysis of local process variations, which lead to the mismatch of transistors who are nominally designed equal for proper circuit functionality. The latter technique allows analyzing the effect of dominant global process variations. While the Monte-Carlo analysis raises prohibitive simulation costs for large values of the robustness to be analyzed (e.g. 3-sigma, 6-sigma), corner cases give no indication about the statistical relevance of the results. For integrated circuits, advanced methods for robustness analysis and optimization have been developed, e.g. worst-case analysis, yield analysis, yield optimization [10]. Diagnosis methods have also been applied in IC design [8, 9]. They support the designer in the task of circuit sizing by revealing the relevant, dominant and redundant parameters and performances at any design stage. These robustness analysis and optimization and diagnosis methods are aimed to be transferred to sensor circuits.

Unlike integrated circuits, sensor circuits realized with MEMS (MEMS-IC systems) incorporate both a mechanical and an electrical part. These parts belong to different energy domains and involve different simulation methods and software platforms. For this reason, the two components are traditionally designed and optimized separately and only analyzed together. However, as the mechanical and electrical component interact, it is desirable to treat the MEMS-IC as a complete system in the design/optimization process. This requires the development of a methodology specific

Authors' address: Florin Burcea, florin.burcea@tum.de; Andreas Herrmann, andreas.herrmann@tum.de; Bing Li, b.li@tum.de; Helmut Graeb, graeb@tum.de, Technical University of Munich, Chair of Electronic Design Automation, Arcisstr. 21, 80333, Munich, Germany .

© 2019 Association for Computing Machinery.

This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Design Automation of Electronic Systems*, <https://doi.org/10.1145/3325068>.

to the task of sensor design, which properly handles together the different analysis methods and component parameters of the two energy domains. This paper presents a new design/optimization methodology that bridges the gap between the simulation accuracy required by the mechanical part and the simulation speed required by the electrical part. The method, generally applicable to MEMS-IC systems, is exemplified on two demonstrator examples: a MEMS accelerometer and a MEMS microphone, each of them together with an integrated readout circuit.

The rest of the paper is organized as follows. Section 2 mentions the state-of-the-art and lists the new contributions of the paper. Section 3 presents the tasks of circuit sizing and diagnosis. Section 4 describes the demonstrator examples used in the paper. Section 5 discusses MEMS modeling. In section 6, the MEMS-IC optimization methodology is presented. Section 7 shows the experimental results and section 8 concludes the paper.

2 STATE OF THE ART AND NEW CONTRIBUTIONS

The problem of MEMS-IC co-design has been treated in previous papers. In [14, 22], a platform for MEMS-IC co-simulation was presented, however, not until the point of co-optimization. In [18, 19], the MEMS and the IC were co-optimized. In all these papers, the model for the MEMS part was manually implemented and not obtained in an automated way from simulation. Also, none of these papers considered mechanical process variation.

This work provides the following contributions:

- combined optimization of the MEMS and the IC parts, in which design parameters from both domains are simultaneously adjusted;
- nominal and yield optimization of MEMS-IC, in consideration of operating and process parameters (including mechanical process parameters);
- dynamic automatic modeling of the MEMS part during optimization;
- efficient control of calling mechanical model generation;
- scripted combination of available tools: Cadence Spectre for integrated circuits, Coventor MEMS+ for MEMS and MunEDA WiCkeD for optimization;
- diagnosis (for insight and for reducing complexity in optimization by leaving out parameters) for the MEMS and IC parts together.

In [2], a new design flow for the robustness optimization of combined MEMS-IC systems was introduced. This is a scripted design flow which brings together in an exemplary way state-of-the-art tools from the MEMS domain and from the IC domain: Spectre, MEMS+ and WiCkeD. The method can be transferred to other tools. The main features of the new method are the following:

- For the MEMS, a circuit-level model is used. This type of model finds the best trade-off between accuracy and computational effort.
- Both electrical and mechanical design parameters are considered simultaneously. Moreover, it was shown that successful optimization needs to consider design parameters from both domains.

In [3], an improved version of the method from [2] was presented. The main additional features of the improved method are the following:

- Both electrical and mechanical process parameters are considered. Moreover, it was shown that mechanical parameters have a major impact on the system robustness.
- The circuit-level MEMS model is generated only when necessary and saved in a database. A control flow was developed for accessing the database with support for simulation on multiple, parallel threads.

This paper is a summarizing overview of [2] and [3], with extended explanations. In addition, co-diagnosis of the MEMS and IC parts is included. Diagnosis is applied for the first time to a combined MEMS-IC system. Through diagnosis, relevant parameters and performances are identified at every design stage and the irrelevant ones are left out to reduce complexity. Corresponding experimental results are presented.

3 MEMS-IC CO-SIZING AND CO-DIAGNOSIS

3.1 Sizing

A circuit has a set of parameters \mathbf{p} and a set of performances \mathbf{f} . Parameters are divided into three categories: design (\mathbf{d}), operating ($\boldsymbol{\theta}$) and statistical (\mathbf{s}) parameters [10]. For extending to the MEMS-IC case, design and statistical parameters are each subdivided into electrical (\mathbf{d}_e and \mathbf{s}_e) and mechanical (\mathbf{d}_m and \mathbf{s}_m). Operating parameters (typically supply voltage and temperature) are not subdivided, because they refer to both the electrical and the mechanical component.

Design parameters are adjusted by the designer or by the automatic sizing tool until the specifications are met. Their nominal value \mathbf{d}_0 , lower bound \mathbf{d}_L and upper bound \mathbf{d}_U need to be specified. They typically refer to transistor, capacitor and resistor sizes. In the context of MEMS-IC co-design they also refer to the geometrical properties of the MEMS.

A circuit must fulfill the specifications for a given operating range, described by the operating parameters. For these parameters, the nominal value $\boldsymbol{\theta}_0$, the lower bound $\boldsymbol{\theta}_L$ and the upper bound $\boldsymbol{\theta}_U$ need to be specified.

Integrated circuits and systems suffer inevitable variations during manufacturing, described by statistical parameters. These parameters are represented by their distribution (any type, internally transformed to gaussian), the nominal values \mathbf{s}_0 (which are typically 0), the standard deviations and the correlation coefficients, the latter two being grouped into the covariance matrix \mathbf{C} .

Performances (\mathbf{f}) quantitatively describe the behavior of a circuit. Their dependence on the circuit parameters \mathbf{p} is computed through numerical circuit simulation for any set of parameter values. Every performance needs to fulfill a lower and/or upper bound, grouped into the specifications \mathbf{f}_B (a lower and an upper bound for the same performance will lead to two specifications). To each specification $f_{B,i}$ a worst-case distance (WCD) β_i is associated [10]. The worst-case distance, expressed in sigma values, is a normalized measure for the circuit robustness with respect to the given specification. The worst-case distance is positive when the specification is fulfilled and negative when it is violated. The worst-case distance directly relates to the yield, e.g., $\beta_i = 3[\sigma]$ corresponds to $Y_i = 99.86\%$.

The goal of analog circuit sizing is to find a set of design parameters \mathbf{d} for which the given specifications \mathbf{f}_B are fulfilled for the whole range of the operating parameters $\boldsymbol{\theta}$ and with high probability (the yield) in consideration of the statistical parameters \mathbf{s} . The sizing can be done in two steps:

- (1) nominal optimization
- (2) yield optimization

The nominal optimization aims to find the set of design parameters $\mathbf{f}(\mathbf{d}_1)$ for which the optimization objective $F(\mathbf{d})$ is minimized:

$$\mathbf{d}_1 = \arg \min_{\mathbf{d}} \{F(\mathbf{d})\} \quad (1)$$

Only nominal process conditions (\mathbf{s}_0) are considered. The optimization is done at worst-case operating conditions, to ensure fulfillment of specifications for the entire operating range. The worst-case operating point $\mathbf{p}_{o,w,i} = (\mathbf{d}_0^T \boldsymbol{\theta}_{w,i}^T \mathbf{s}_0^T)^T$ is defined for each individual specification $f_{B,i}$ and contains the set of operating parameters $\boldsymbol{\theta}_{w,i}$ for which this performance takes the worst

value within the operating range $\theta_L \leq \theta \leq \theta_U$. The set of design parameters $f(\mathbf{d}_1)$ is computed in iterative steps and the worst-case operating points $\theta_{w,i}$ are recalculated at every iteration. The nominal optimization stops when all specifications are fulfilled.

The yield optimization aims to maximize the yield of the circuit, defined as the probability that the circuit fulfills all specifications under all operating conditions. This most simulation-intensive step benefits from starting at the nominally optimized point \mathbf{d}_1 . While the nominal optimization considers only nominal process conditions, the yield optimization takes into account process variations, described by the statistical parameters \mathbf{s} .

The yield is the percentage of circuits satisfying the specifications. It can be estimated by a Monte-Carlo analysis. In this paper, we estimate the yield by a deterministic approach, which is based on so-called worst-case distances [1, 8, 10, 11]. For each specification, a specific parameter set is computed. It is defined as the parameter set with maximum probability (density) to violate the specification value. After a transformation of the probability density function, this specific parameter set, which actually is the worst-case parameter set for a performance specification $f_{B,i}$, can be computed by solving a nonlinear optimization problem with quadratic objective function and one nonlinear constraint [1, 8, 10, 11]. The distance between worst-case parameter set and nominal parameter set is denoted as worst-case distance β_i . It has been shown that the worst-case distance is particularly suited to estimate the yield for one specification according to:

$$Y_i(\mathbf{d}) = P \left\{ \forall_{\theta_L \leq \theta \leq \theta_U} f_{B,i}(\mathbf{d}, \theta, \mathbf{s}) \geq \mathbf{0} \right\} = P \{ f_{B,i}(\mathbf{d}, \theta_{w,i}, \mathbf{s}) \geq \mathbf{0} \} = \int_{-\infty}^{\beta_i} \frac{1}{\sqrt{2\pi}} e^{-t^2} dt. \quad (2)$$

The last term in (2) leads to an interpretation of the worst-case distance as a multiple of a virtual standard deviation. This is a user-friendly metric to measure the robustness of a specification in terms of β_i -sigma design.

The overall yield optimization problem is then formulated by combining the individual yield values in a cost function $cost(\mathbf{d})$ that is to be minimized:

$$\mathbf{d}_2 = \arg \min_{\mathbf{d}} \{ cost(\mathbf{d}) \} \quad (3)$$

Like for the nominal optimization, \mathbf{d}_2 is calculated iteratively and the worst-case point is updated at every iteration. Cost functions will be discussed in Section 3.2.

3.2 Diagnosis

Diagnosis supports the designer in the task of circuit sizing. Diagnosis can be used for the preparation of a circuit for automatic optimization and/or as an instrument for interactive circuit sizing.

The purpose of diagnosis is to examine the circuit dependencies, helping the designer to better understand the circuit. The main features examined are the relevance, dominance and redundancy of parameters and performances.

Circuit diagnosis is based on numerical circuit simulation. The sensitivity of the performances of interest vs. the circuit parameters, representing the partial derivative computed by finite difference approach, plays the central role here.

Diagnosis supports circuit sizing for the final goal of yield optimization (or design centering), through methods which reveal the relevant, dominant and redundant parameters and performances. The diagnosis concepts and methods introduced in [8, 9] for integrated circuit sizing will be applied, in an extended version, to systems consisting of an integrated circuit (IC) and a micro-electro-mechanical system (MEMS).

The central idea of diagnosis is the linearized dependence between a specific objective and specific parameters. In the following, we will formulate diagnosis for worst-case distances and

design parameters. The formulations hold analogously for, e.g., performance objectives and/or statistical parameters.

The linearized dependence between worst-case distances and design parameters is expressed as:

$$\Delta\boldsymbol{\beta} = \mathbf{S}_{\boldsymbol{\beta}d} \cdot \Delta\mathbf{d} \quad (4)$$

The matrix $\mathbf{S}_{\boldsymbol{\beta}d}$ consists of the worst-case distance sensitivities vs. design parameters, meaning that the i -th row of $\mathbf{S}_{\boldsymbol{\beta}d}$ contains the gradient $\nabla_{\mathbf{d}}\beta_i$ of the worst-case distance β_i at the worst-case point (worst-case operating and worst-case process) $\mathbf{p}_{w,i} = (\mathbf{d}_0^T \boldsymbol{\theta}_{w,i}^T \mathbf{s}_{w,i}^T)^T$ with respect to all design parameters \mathbf{d} . The worst-case distance β_i can also be formulated as multiple of a standard deviation $\sigma_{\bar{f}_i}$ of the performance linearized in its worst-case parameter set according to [1, 8, 10, 11]. In addition, a gradient of the worst-case distance with regard to design parameters $\nabla_{\mathbf{d}}\beta_i$ has been derived based on the gradient of the performance feature with regard to statistical parameters $\nabla_{\mathbf{s}}f_i$ and with regard to design parameters $\nabla_{\mathbf{d}}f_i$ [1, 8, 10, 11]:

$$\nabla_{\mathbf{d}}\beta_i = \frac{-k}{\sigma_{\bar{f}_i}} \cdot \nabla_{\mathbf{d}}f_i \quad \text{with} \quad \sigma_{\bar{f}_i} = \sqrt{\nabla_{\mathbf{s}}^T f_i \cdot \mathbf{C} \cdot \nabla_{\mathbf{s}} f_i} \quad (5)$$

\mathbf{C} is the covariance matrix of statistical parameters. The sign coefficient k is 1 for lower and -1 for upper bounds.

In this paper, the diagnosis methods are applied to the extended MEMS-IC case. To investigate how much contribution to the worst-case distance is from the electrical side and how much from the mechanical side, it is useful to split Eq. (4) as follows:

$$\underbrace{\Delta\boldsymbol{\beta}_e + \Delta\boldsymbol{\beta}_m}_{\Delta\boldsymbol{\beta}} = \underbrace{\begin{bmatrix} \mathbf{S}_{\boldsymbol{\beta}d,e} & \mathbf{S}_{\boldsymbol{\beta}d,m} \end{bmatrix}}_{\mathbf{S}_{\boldsymbol{\beta}d}} \cdot \underbrace{\begin{bmatrix} \Delta\mathbf{d}_e \\ \Delta\mathbf{d}_m \end{bmatrix}}_{\Delta\mathbf{d}} \quad (6)$$

Parameters and performances consist of physical quantities whose absolute values can differ by several orders of magnitude. For the numerical methods to perform best, normalization is necessary. The worst-case distance is expressed in units of σ and already represents a normalized quantity. Therefore, only normalization of design parameters is needed in Eq. (4). This is done based on the allowed range for the design parameters, $\mathbf{d}_U - \mathbf{d}_L$. The normalized sensitivity matrix and vector of design parameters are given as:

$$\begin{aligned} \mathbf{S}_{\boldsymbol{\beta}d}^* &= \mathbf{S}_{\boldsymbol{\beta}d} \cdot \mathbf{N}_d, \quad \Delta\mathbf{d}^* = \Delta\mathbf{d} \cdot \mathbf{N}_d^{-1}, \\ \mathbf{N}_d &= \text{diag}(d_{U,1} - d_{L,1}, \dots, d_{U,n_d} - d_{L,n_d}) \end{aligned} \quad (7)$$

To show how much relative contribution each of the electrical and the mechanical part bring to the improvement of each worst-case distance, the relative influences $\Delta\beta_{e,i}[\%]$ and $\Delta\beta_{m,i}[\%]$ can be calculated as:

$$\begin{aligned} \Delta\beta_{e,i}[\%] &= \frac{\Delta\beta_{e,i}}{\Delta\beta_{e,i} + \Delta\beta_{m,i}}, \quad \Delta\beta_{m,i}[\%] = \frac{\Delta\beta_{m,i}}{\Delta\beta_{e,i} + \Delta\beta_{m,i}}, \\ \Delta\boldsymbol{\beta}_e &= \text{abs}(\mathbf{S}_{\boldsymbol{\beta}d,e}^*) \cdot \text{abs}(\Delta\mathbf{d}_e^*), \quad \Delta\boldsymbol{\beta}_m = \text{abs}(\mathbf{S}_{\boldsymbol{\beta}d,m}^*) \cdot \text{abs}(\Delta\mathbf{d}_m^*). \end{aligned} \quad (8)$$

The elements of the sensitivity matrices $\mathbf{S}_{\boldsymbol{\beta}d,e}^*$, $\mathbf{S}_{\boldsymbol{\beta}d,m}^*$ and the parameter variations $\Delta\mathbf{d}_e^*$ and $\Delta\mathbf{d}_m^*$ are taken in their absolute value, such that each parameter change causes a worst-case distance change in the same direction.

In order to optimize the circuit yield, the worst-case distances have to be increased simultaneously. To convert a set of objectives into a scalar objective function, which shall be minimized, several types of cost functions can be used [21]:

(1) the exponentially weighted sum:

$$Q_{exp} = \sum_i^{n_B} e^{-2\beta_i}, \quad (9)$$

However, in this function it can happen that the cumulated influence of many large worst-case distances can shade a small worst-case distance.

(2) the truncated least-squares approach:

$$Q_{lsq} = \sum_i^{n_B} (\max(0, \varepsilon_i))^2, \quad \varepsilon_i = \sigma_{obj} - \beta_i \quad (10)$$

The threshold σ_{obj} depends on the design objective. A typical value is 3 (3-sigma design). In this case study, the value 4 will be used. This function always finds a suitable point (if it exists), but not necessarily the optimum one.

(3) the max-norm formulation:

$$Q_{max} = \max_i(-\beta_i) \quad (11)$$

As it is not continuously differentiable, it will not be further considered in this paper.

The diagnosis methods are suitable for examining circuit dependencies. In this way, irrelevant parameters and performances can be neglected, to reduce the computational effort of the sizing task. Four situations lead to such reductions:

- (1) design parameters dependent on each other
- (2) performances dependent on each other
- (3) non-relevant circuit specifications (a specification is relevant when increasing its WCD significantly improves the overall yield)
- (4) non-relevant design parameters for improving the yield

To examine the relative dependencies of two design parameters d_i and d_j or two worst-case distances β_i and β_j according to (1) and (2), the cosine of the angle $\alpha_{i,j}$ between the two respective column vectors \mathbf{s}_i and \mathbf{s}_j or two row vectors \mathbf{z}_i and \mathbf{z}_j of the sensitivity matrix $\mathbf{S}_{d\beta}^*$ is used. If $|\cos(\mathbf{s}_i, \mathbf{s}_j)| = 1$ for two design parameters d_i and d_j , then they are linearly dependent, and if $\cos(\mathbf{s}_i, \mathbf{s}_j) = 0$, they have an orthogonal effect. If $\cos(\mathbf{z}_i, \mathbf{z}_j) = 1$ for two worst-case distances β_i and β_j , then they can be simultaneously improved, if $\cos(\mathbf{z}_i, \mathbf{z}_j) = -1$, then those performances form a trade-off, and if $\cos(\mathbf{z}_i, \mathbf{z}_j) = 0$ then they can be altered independently from each other.

A specification $f_{B,i}$ is considered non-relevant according to (3) when its worst-case distance β_i increases simultaneously with the worst-case distance β_j of another specification and $\beta_i > \beta_j$. This is quantitatively expressed by the coefficient:

$$r_{ij} = e^{-(\beta_i - \beta_j)} \cdot \alpha_{i,j} \quad (12)$$

A parameter η represents the threshold for considering a specification relevant. If $r_{ij} < \eta$ (if the angle $\alpha_{i,j}$ between \mathbf{z}_i and \mathbf{z}_j is very small and/or one worst-case distance is much larger than the other), the largest of the two worst-case distances is considered non-relevant. In this case study, η is chosen 0.1. The worst-case distances of the relevant specifications are saved in the vector $\boldsymbol{\beta}_{rel}$.

The relevance of a design parameter according to (4) is evaluated based on how much it is suited to increase a worst-case distance corresponding to a relevant specification. First, the cost functions (exponential and truncated least squares) which keep only the terms corresponding to the relevant

worst-case distances are defined:

$$Q_{exp,rel} = \sum_i e^{-2\beta_{rel,i}}, \quad (13)$$

$$Q_{lsq,rel} = \sum_i (\max(0, \sigma_{obj} - \beta_{rel,i}))^2. \quad (14)$$

The gradient of the cost function is used as criterion for the relevance of parameters:

$$-\nabla_d Q_{exp,rel} = \sum_i 2 \cdot \nabla_d \beta_{rel,i} \cdot e^{-2\beta_{rel,i}}, \quad (15)$$

$$-\nabla_d Q_{lsq,rel} = \sum_i 2 \cdot \nabla_d \beta_{rel,i} \cdot \max(0, \sigma_{obj} - \beta_{rel,i}). \quad (16)$$

The gradients from (15) and (16) can differ by several orders of magnitudes between design stages (because of the changing worst-case distances) and between the two cost functions. To have a consistent scale, the relevances are normalized by dividing the magnitude of the relevance of one parameter by the sum of magnitudes of relevances of all parameters:

$$|\nabla_d Q_{exp,rel,norm}| = \frac{|\nabla_d Q_{exp,rel}|}{\sum_j |\nabla_{d_j} Q_{exp,rel}|}, \quad (17)$$

$$|\nabla_d Q_{lsq,rel,norm}| = \frac{|\nabla_d Q_{lsq,rel}|}{\sum_j |\nabla_{d_j} Q_{lsq,rel}|}. \quad (18)$$

Diagnosis can also be formulated for other objectives and types of parameters, e.g., for performances f and process parameters s , with the linear dependence expressed as:

$$\Delta f = S_{fs} \cdot \Delta s. \quad (19)$$

The matrix S_{fs} contains the performance sensitivities vs. process parameters, meaning the i -th row contains the gradient $\nabla_s f_i$ with respect to all process parameters s .

Like for the design parameters, normalization is necessary. Performances are normalized by referring to the specification bound:

$$f_i^* = k \cdot \left(\frac{f_i}{f_{B,i}} - 1 \right), \quad \Delta f^* = N_f^{-1} \cdot \Delta f, \quad N_f = \text{diag}(f_{B,1}, \dots, f_{B,n_f}), \quad (20)$$

where the sign coefficient k is 1 for lower and -1 for upper bounds. The normalized performance expresses the deviation from the specification value in percentage of specification value. For the normalized performance, zero corresponds to the specification bound, a positive value means that the specification is fulfilled and a negative value that the specification is violated.

Process parameters are normalized through division by the standard deviation:

$$\Delta s^* = N_s^{-1} \cdot \Delta s, \quad N_s = \text{diag}(\sigma_1, \dots, \sigma_{n_s}). \quad (21)$$

A normalized process parameter expresses the deviation from the mean value in multiples of standard deviation.

The sensitivities are then normalized, to express the linearized dependence between normalized performances and normalized statistical parameters:

$$\Delta f^* = S_{fs}^* \cdot \Delta s^* \quad \text{with} \quad S_{fs}^* = N_f^{-1} \cdot S_{fs} \cdot N_s. \quad (22)$$

The measure used in this paper for the relevance of a process parameter is the sum of the magnitudes of the normalized sensitivities of all performances vs. this parameter:

$$R_s = \sum_i |\nabla_s f_i|. \quad (23)$$

To compare the process parameters from the point of view of their influence, the relevance of a parameter is normalized by dividing it by the sum of the relevances of all process parameters:

$$R_{norm,s} = \frac{R_s}{\sum_j R_{s_j}}. \quad (24)$$

4 DEMONSTRATOR EXAMPLES

4.1 MEMS Microphone with Integrated Readout Circuit

The MEMS capacitive microphone can be perceived as a parallel-plate capacitor between a fixed backplate and a movable pressure-sensitive membrane, separated by a small air gap acting as dielectric [16]. The 3D model is illustrated in Fig. 1 (a) and the simplified structure in Fig. 1 (b). An incident sound wave causes the membrane to deflect, thus changing the capacitance between the membrane and the backplate due to varying air gap thickness. The first-order model of the MEMS microphone is a variable capacitance $C_{mic} + C_{\Delta}$, shown in Fig. 1. C_{mic} is the static capacitance in the absence of membrane deflection and C_{Δ} is the capacitive variation due to the membrane deflection, proportional to the input acoustic pressure P_{in} .

The capacitive variation ΔC is converted to an electrical signal by the readout circuit. A constant-charge approach is employed in this example. The principle is illustrated in Fig. 2 (a). A very large resistance R_0 is connected in series with the microphone capacitance C_{mic} . The series connection is supplied by the reference voltage V_{ref} . The DC voltage on C_{mic} is V_{ref} . Because of the very large $R_0 C_{mic}$ time constant, the charge Q on C_{mic} remains nearly constant over time under capacitive variations:

$$Q = const = C_{mic} \cdot V_{ref} = (C_{mic} + \Delta C) \cdot (V_{ref} + \Delta V). \quad (25)$$

In this way, capacitive variations ΔC are translated to voltage variations ΔV . Assuming small capacitive relative variations ($\Delta C \ll C_{mic}$), ΔV depends linearly on ΔC and consequently on the input pressure P_{in} :

$$\Delta V = \frac{-\Delta C \cdot V_{ref}}{C_{mic} + \Delta C} \approx \frac{-\Delta C \cdot V_{ref}}{C_{mic}} \propto P_{in}, \quad \Delta C \ll C_{mic}. \quad (26)$$

However, small capacitive variations lead to a low gain of ΔV vs. ΔC . The signal ΔV must be further amplified. This is done by an operational amplifier with capacitive feedback. A Miller opamp was preferred to a cascode, in order to have larger output swing. The resistance R_b is needed to bias the opamp negative input and is realized with PMOS transistors in series connected as back-to-back diodes [13]. The resistance R_0 serves for biasing the opamp positive input and is implemented as a diode-connected PMOS transistor, biased in deep subthreshold region.

The sensor circuit parameters determine its performances. In IC design, the design parameters relate only to the electrical devices (e.g. transistors, resistors). In MEMS-IC systems, there are additional parameters of the mechanical component. In this MEMS-IC example, the electrical parameters are sizes of the transistors and capacitors in the circuit (15 electrical parameters in total), while the mechanical parameters are the electrode radius and the difference between the membrane radius and the electrode radius. In the presented MEMS microphone example, the process parameters refer only to the electrical part of the system. No process parameters were considered for the mechanical component. However, this is not a limitation of the optimization methodology, but of the available MEMS model for the presented case study. If the MEMS model provided such parameters, they would be incorporated into the optimization tool in the same way as for the electrical counterpart. The operating parameters are the temperature and the supply

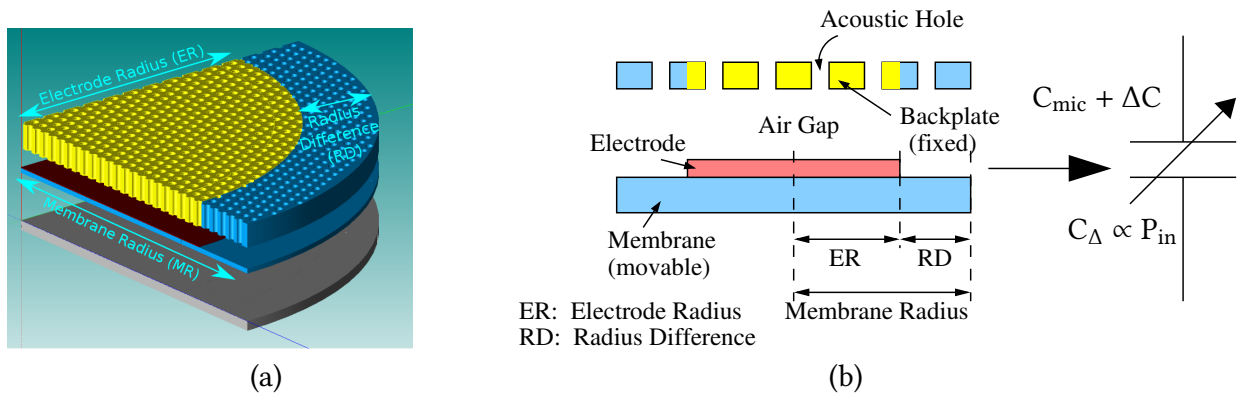


Fig. 1. MEMS capacitive microphone: a) 3D model showing one quarter of the structure [7] b) simplified mechanical structure and electrical model.

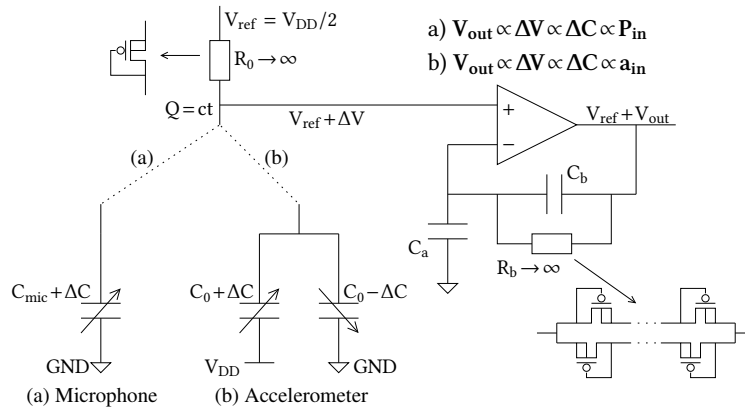


Fig. 2. Principle schematic of the capacitive readout circuit a) for the MEMS microphone b) for the MEMS accelerometer.

Table 1. Number of microphone parameters from each category.

Design		Operating	Process
Electrical	Mechanical		
15	2	2	48

voltage, referring to both the electrical and the mechanical part. The number of parameters for each category is summarized in Table 1.

The selected example is described by 6 performances:

- the gain of the system ($Gain_{dB_1kHz}$) is the sensitivity of the output voltage vs. the input pressure measured in dB at the reference frequency 1kHz,
- the signal-to-noise-ratio (SNR_{1PaRMS}) is measured A-weighted [5] for an input pressure level of 1PaRMS,
- the total harmonic distortion (THD_{1PaRMS}) of the output signal is measured for a sinusoidal input signal of 1kHz with sound pressure level of 1PaRMS [5],
- the phase margin PM_{Opamp} of the capacitive feedback loop over the opamp measures the stability,
- the low-frequency 3dB bandwidth (BW_{low}), as described in Fig. 4,
- the high-frequency 3dB bandwidth (BW_{high}), also described in Fig. 4.

The following analyses are used to measure the performances: AC analysis for the gain and for the bandwidths, transient analysis for THD, noise analysis for SNR and stability analysis for the phase

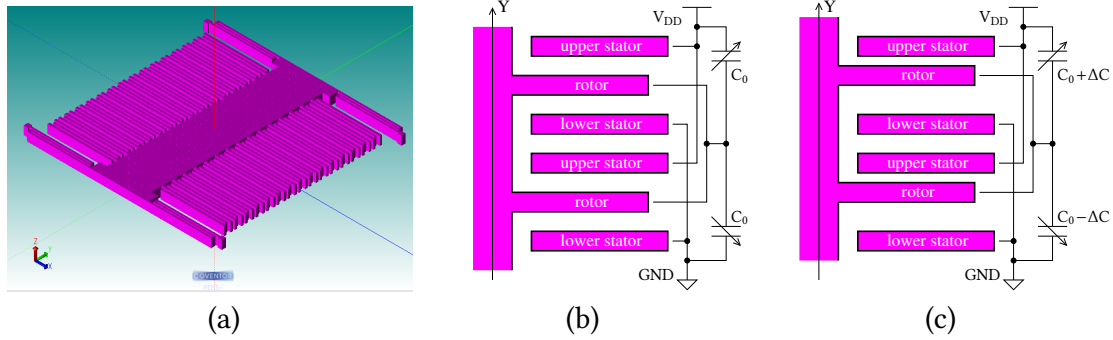


Fig. 3. MEMS accelerometer: a) 3D model [7] b) simplified representation without displacement c) simplified representation with displacement

Table 2. Number of accelerometer parameters from each category.

Design		Process		Operating
Electrical	Mechanical	Electrical	Mechanical	
16	15	56	2	2

margin [4]. The DC analysis is used for the electrical constraints. Out of these, transient analysis requires the highest computational effort.

4.2 MEMS Accelerometer with Integrated Readout Circuit

MEMS accelerometers have been commonly used as demonstrator examples in previous work related to MEMS-IC co-design [14, 18, 19, 22]. However, these approaches used a manually implemented and not an automatically generated MEMS model. Also, they did not account for process variation. In [12], process variation for a MEMS accelerometer was considered in the context of MEMS robustness analysis, however, not in the context of MEMS-IC co-optimization.

The 3D model of a MEMS accelerometer is shown in Fig. 3 (a), while its operating principle is depicted in Fig. 3 (b) and (c). The MEMS accelerometer consists of three electrodes. Two of them (the upper and lower stators) are fixed, while the third (the rotor) moves along the Y-axis under the influence of the acceleration. This movement results in a capacitive change [15], which is translated to an electrical signal by the readout circuit, as explained in Fig. 2 (b). The positive input of the opamp is a high impedance node, therefore the charge at that node remains constant over time. In this way, small relative capacitive changes ($\Delta C \ll C_0$) lead to proportional, small relative voltage variations ($\Delta V \ll V_{ref}$). Like for the microphone, the output signal is obtained from the signal ΔV amplified by a gain stage, implemented as an operational amplifier with capacitive feedback.

The electrical part of the system is made up of 21 transistors and 3 capacitors. From the point of view of the optimization problem, these circuit elements are represented by a total of 16 design parameters, referring to transistor and capacitor sizes. The inherent process variations are described by global process and local mismatch parameters, 56 in total.

The mechanical part is represented by the MEMS accelerometer shown in Fig. 3. The 14 design parameters of this MEMS describe its geometry. Also, two mechanical statistical parameters are subject to optimization. These are the mechanical layer thickness and the edge underetch. Both are global process parameters and both have a gaussian distribution. They were provided as foundry data for an industrial process.

The system has two operating parameters, the global temperature and the supply voltage. They relate to both the electrical and the mechanical parts. The parameter count from each category is summarized in Table 2.

This MEMS-IC example is described by 4 performances:

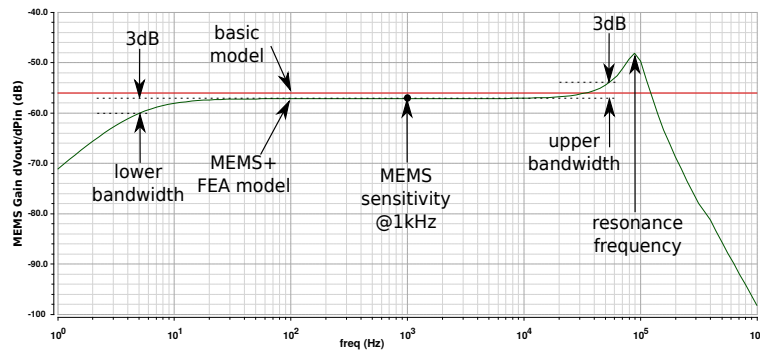


Fig. 4. Frequency responses for the basic model and the MEMS+ model. The latter models more realistically the MEMS behaviour. The characteristics of the frequency response are described on the plot. Out of these, only the *sensitivity* at 1kHz is applicable to the basic model.

- the resonance frequency of the MEMS (*FreqRes*),
- the gain of the output signal V_{out} vs. the input acceleration, expressed in mV/g (*Gain_mV_g*),
- the nonlinearity, measured through the total harmonic distortion of the output signal for a sinusoidal input acceleration with 40g amplitude (*THD_40g*),
- the phase margin for the capacitive feedback loop over the opamp (*PM_Opamp*).

The resonance frequency and the gain are measured with the AC analysis, the THD with transient analysis and the phase margin with stability analysis. The DC analysis is used for the electrical constraints.

5 AUTOMATIC MEMS MODELING FOR MEMS-IC CO-OPTIMIZATION

MEMS-IC systems include a mechanical and an electrical part. These belong to two different energy domains, each requiring specific analysis methods and software platforms. In the topic of MEMS-IC co-design, a very important issue is the modeling of the MEMS part. This can be done at different levels of complexity.

First, a basic model can be used. This is made up of one or a few simple circuit elements which capture the principle behavior. Examples are the microphone modeled as a linearly variable capacitance [5] and the accelerometer modeled as two linearly variable capacitances [19]. Such models are very suitable for understanding the principle behavior and can be a good starting point in a design. They also provide fast simulation and optimization capabilities. However, they are too simple and not sufficiently accurate, as the behavior of the MEMS is much more complex. This is illustrated in Fig. 4 for the frequency response of a microphone. With the basic model, the response is flat at all frequencies, such that it does not capture the low-frequency roll-off, the resonant peak and the high-frequency roll-off.

Analytical modeling has also been used in previous MEMS-IC co-design approaches [14, 22]. This consists of differential equations describing the MEMS behavior. The implementation is typically done in a hardware description language (HDL) like Verilog-A. One disadvantage of this modeling approach (which holds for the basic models as well) is the lack of generality, since each type of MEMS has its own equations. The other disadvantage of such manually implemented models is the still limited accuracy.

Significantly more accurate and more automated is the modeling based on finite element analysis (FEA). Such a capability is provided by specialized CAD tools like Coventor MEMS+. In MEMS+, MEMS designs are assembled out of the elements of a library of parametric building blocks, which have underlying FEA-based models. MEMS+ assembles the individual models of the building blocks

Table 3. Problem sizes for the mechanical and the electrical part

Example	Mechanical part		Electrical part
	Degrees of freedom		Transistor count
	FEA model	Circuit-level model	
Microphone	180	3	17
Accelerometer	6	1	21

into a compact model (still based on FEA) consisting of a system of equations with a significantly smaller number of degrees of freedom (and hence with lower complexity) than a conventional FEA model, where the degrees of freedom represent the internal dynamics of the system [20, 23]. In Fig. 4, the FEA-based model shows a realistic frequency response in comparison to the basic model. In this paper, the MEMS+ FEA-based model was integrated into the electrical CAD environment. Cadence Virtuoso was used for schematic editing and circuit simulation and MunEDA WiCkeD was used for circuit optimization. The properties of the mechanical part (for the microphone example, the electrode radius and the difference between the membrane radius and the electrode radius) were exposed [7] and used as design parameters in the optimization, together with the design parameters of the electrical devices. This setup allows the robustness optimization of the MEMS and the IC as a complete system.

Although this methodology provides much better accuracy, the computational cost associated with the transient analysis can become prohibitive (because of the FEA model complexity). However, the accuracy of the FEA model is not needed at circuit level. The computational cost is overcome by using the model reduction feature of MEMS+ [20, 23]. Model reduction creates a Verilog-A (VA) file, which is a low-order polynomial version of the MEMS+ schematic. The reduced-order model of MEMS+ automatically includes nonlinearities inherent in the model. In MEMS+ the model order reduction is based on a mechanical eigen-decomposition of the system for which selected modes of interest can be preserved. The reduced-order Verilog-A model, which will be further on called "circuit-level model", is exported and replaces the FEA model [6].

Although simplified, the circuit-level model keeps the essential characteristics needed for circuit simulation. Each mechanical component has several intrinsic modal frequencies (modes). With model order reduction, the MEMS+ mechanical degrees of freedom are replaced by reduced degrees of freedom corresponding to preserved modes. Ideally, one reduced degree of freedom is needed for each preserved mode; nevertheless, in practice more reduced degrees of freedom are needed to correctly reproduce preserved modes and operating points [6]. For the MEMS microphone, it is sufficient that the exported circuit-level model keeps only the first mode (corresponding to the resonance frequency, shown in Fig. 4). The frequency response for the circuit-level model has the same shape as for the FEA model and the numerical characteristics of the curve are very similar, as shown in Table 4. The accuracy is very little affected when using the reduced-order model, because although it is a simplified model, it keeps the essential characteristics of the MEMS. Table 3 shows the number of degrees of freedom before (with the FEA model) and after (with the circuit-level model) model order reduction for both MEMS examples from this paper. In addition to the degrees of freedom, the table contains the number of transistor from the readout circuit, to illustrate the problem sizes for the electrical and the mechanical parts. This paper uses demonstrative examples, for which the number of degrees of freedom is not very large compared to industrial MEMS. The industrial example from [20] has 4000 degrees of freedom before and 20 after model order reduction. However, even for the microphone from this paper, with 180 degrees of freedom before and 3 after, the CPU time of the optimization is already reduced approximately 4 times. The circuit-level model of the accelerometer only has one degree of freedom (because this MEMS has a translational movement along a single axis). This accelerometer is a very simple example from this point of view, but it is a complete example from the point of view of parameter types.

Table 4. Comparison between the complete MEMS+ model (FEA) and the circuit-level (reduced-order) model.

Mechanical Parameters (μm)	Model	$dV_{\text{out}}/dP_{\text{in}}$ Gain @1kHz (dB)	Lower 3dB Bandwidth (Hz)	Upper 3dB Bandwidth (Hz)	Resonance Frequency (Hz)
ER = 250 RD = 250	FEA	-61.22	6.79	73.19k	125.9k
	circuit-level	-60.84	6.50	74.22k	125.9k
ER = 400 RD = 250	FEA	-57.44	4.67	57.94k	100k
	circuit-level	-57.25	4.62	58.36k	100k
ER = 600 RD = 150	FEA	-57.02	4.89	52.75k	89.1k
	circuit-level	-57.17	4.90	52.46k	89.1k
ER = 800 RD = 200	FEA	-52.54	2.58	42.49k	70.8k
	circuit-level	-52.71	2.60	42.20k	70.8k

However, this methodology does not allow yet the co-optimization of the MEMS part and the IC part. The circuit-level model exported from MEMS+ is not parameterized. Consequently, the properties of the MEMS cannot be considered as design parameters in this approach. When co-optimizing the MEMS and IC part, the mechanical parameters change during the optimization process, which requires updating the circuit-level model for every new set of parameters. This feature is not available in the commercial software for analysis and optimization.

6 OPTIMIZATION METHODOLOGY

To overcome this limitation, a new modeling&simulation-in-a-loop flow was implemented. It is based on the commercial tools WiCkeD [17] for optimization, Spectre [4] for continuous simulation and MEMS+ [7] for MEMS simulation, but can be extended to other tools.

In this work, WiCkeD is used to perform optimization for finding the optimal sizing. The optimization methods provided by WiCkeD are mostly based on sequential quadratic programming. These optimization algorithms, already existing in the commercial tools, have been previously applied for integrated circuits. This paper advances their application to the new combined MEMS and IC domains.

In Fig. 5 (a), the original simulation call is shown. In every step of the optimization, one electrical or mechanical parameter is changed based on the optimization algorithm of the WiCkeD software. The runscript executes one Spectre simulation with the modified set of parameters and a netlist which already includes the path to the accurate, FEA-based MEMS+ model. During the run of the Spectre simulation, MEMS+ binary libraries with the FEA model are loaded.

In the new approach implemented in this paper, a script is executed before Spectre is called, as shown in Fig. 5 (b). This script removes the path to the FEA MEMS+ model of the original netlist and reads in all mechanical parameters. Next, it checks if a circuit-level (VerilogA) model already exists with these parameters. In case it does not exist, a reduced model is created and included in the new netlist, otherwise the existing reduced model is included in the netlist. Spectre then runs with the circuit-level model instead of the FEA model. This approach saves CPU time by using a reduced-order model for the MEMS part or re-using the already existing MEMS model.

Exporting a reduced-order model includes calling MEMS+ and executing the model order reduction in MEMS+. For the demonstrator example of the accelerometer, the runtime for the export operation is approximately 2 seconds, and for the microphone, approximately 4 seconds. Because an optimization typically requires thousands of simulations, the export operation should be executed only when the model is not yet available. The circuit model of the MEMS depends on the mechanical design and statistical parameters and on the operating parameters (temperature and supply voltage). Therefore, if at least one of these parameters was altered, a new, updated model is required and needs to be exported from MEMS+. However, during the optimization it is possible to revisit the same parameter set multiple times.

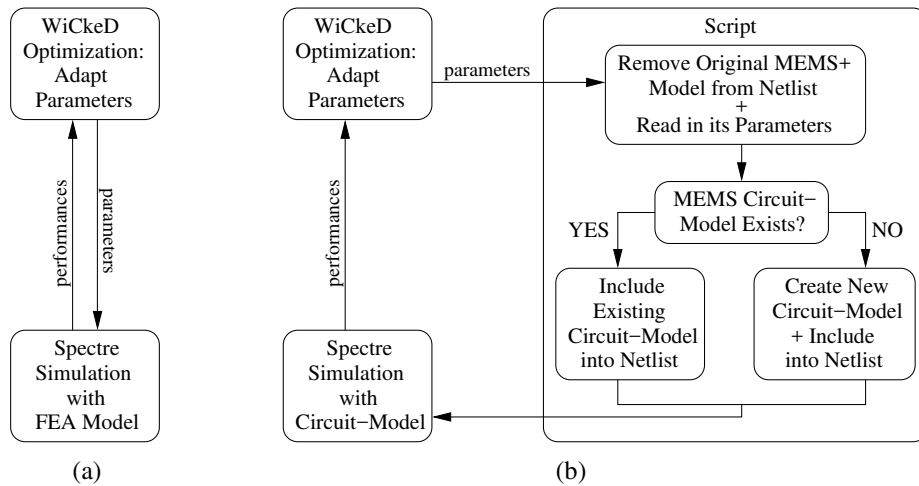


Fig. 5. a) State-of-the-art optimization flow. b) New proposed optimization flow.

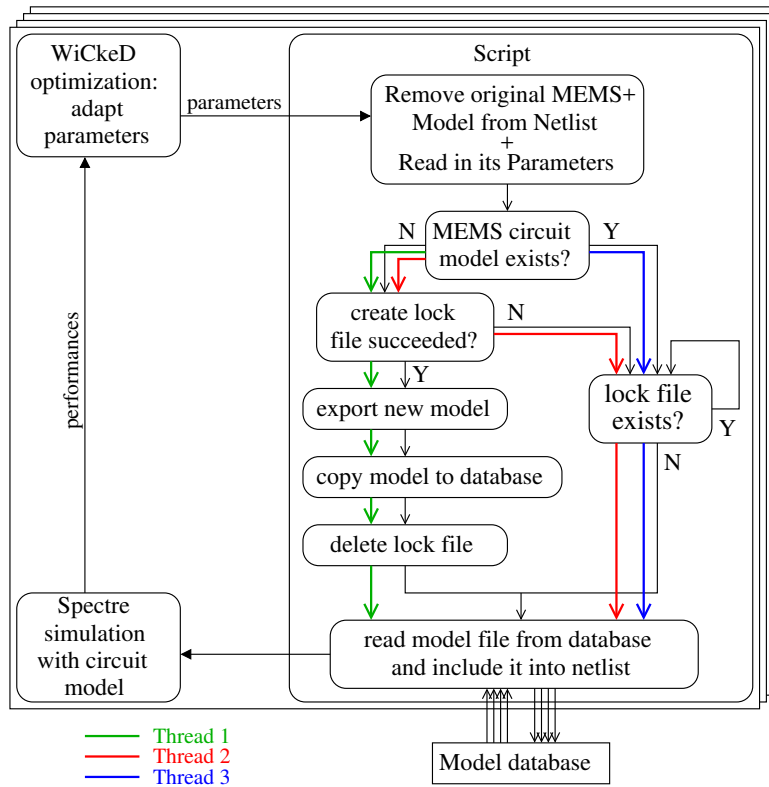


Fig. 6. New proposed optimization flow with flow control for database access and parallel simulation.

The simple approach (as in [2]) would be as follows: when the mechanical parameters change, the model is updated and, when only the electrical parameters change, the existing model (from the immediate previous simulation) is reused. Fig. 7 explains why this solution is not optimal.

The more complex and more efficient approach (as in [3]), is to store all previously exported models in a central MEMS model database (MDB). In this way, a circuit model corresponding to a specific set of parameters is exported only once during optimization. Afterwards, it is not overwritten anymore and it is retrieved from the database and included in the netlist whenever it is needed again. Additional time is saved by executing the export command less often. The MEMS model database access is straightforward when simulating on a single thread, because only one

Step	Parameter set	MEMS model	Export new MEMS model?	
			w/o MDB	w/ MDB
1:	$\{e_1, m_1, e_2, m_2, e_3, e_4\}$	\rightarrow M1 (m_1, m_2)	y	y
2:	$\{e_1^*, m_1, e_2, m_2, e_3, e_4\}$	\rightarrow M1 (m_1, m_2)	n	n
3:	$\{e_1, m_1^*, e_2, m_2, e_3, e_4\}$	\rightarrow M2 (m_1^*, m_2)	y	y
4:	$\{e_1, m_1, e_2^*, m_2, e_3, e_4\}$	\rightarrow M1 (m_1, m_2)	y	n
5:	$\{e_1, m_1, e_2, m_2^*, e_3, e_4\}$	\rightarrow M3 (m_1, m_2^*)	y	y
6:	$\{e_1, m_1, e_2, m_2, e_3^*, e_4\}$	\rightarrow M1 (m_1, m_2)	y	n
7:	$\{e_1, m_1, e_2, m_2, e_3, e_4^*\}$	\rightarrow M1 (m_1, m_2)	n	n
...

Fig. 7. Parameter changes and MEMS model update in the sensitivity analysis, for two approaches. In the first approach, without a MEMS model database (MDB), the mechanical parameters are compared only to the immediate previous simulation. In the second approach, MEMS models are saved in a database. The electrical parameters are e_1 to e_4 and the mechanical parameters are m_1 and m_2 . The * marks a parameter change from the initial point. The MEMS model depends only on m_1 and m_2 . When comparing the mechanical parameter set only with the previous step as in the approach without MDB, a new MEMS circuit model is exported at 5 of the 7 shown steps, even though the analysis needs only 3 different models, as in the approach with MDB.

process reads and writes the database. However, it requires additional considerations in parallel simulation, to prevent data corruption.

First, it must be ensured that a model file is not written simultaneously by two threads. Second, it must be ensured that a model file is read only after having been completely written. Fig. 6 presents the new optimization flow with flow control for database access and parallel simulation. The coloured arrows illustrate thread examples. The current thread first checks if the circuit-level MEMS model exists in the database. If not, the thread (green in Fig. 6) *exclusively creates a lock file* (only this process can execute the operation) corresponding to that model. Then, the model file is exported and copied to the database, after which the lock file is deleted. If the lock file creation fails, the thread (red in Fig. 6) waits until the lock file is deleted and then reads the model from the database. If the current thread (blue in Fig. 6) finds the model from the beginning in the database, it waits until the lock file is deleted and then reads the model.

7 EXPERIMENTAL RESULTS

The optimization of the two demonstrator examples was performed in the commercial tool WiCkeD [17]. Co-optimization will be presented first for the microphone, then for the accelerometer. In the second example, the role of diagnosis in identifying the relevant parameters and performances will be emphasized. Also, the impact of the mechanical parameters and the importance of electrical and mechanical co-optimization will be discussed for both examples. Finally, CPU runtime aspects will be discussed, showing the speed improvement when using the circuit-level MEMS model and also the advantage of using a database of previously exported models.

7.1 Microphone and Readout Co-Optimization

The microphone example emphasizes the importance of the simultaneous consideration of both electrical and mechanical design parameters in the optimization. Also, it is an illustrative example for the computational time reduction by using the circuit-level model. Runtime aspects will be discussed in Section 7.3.

In the initial design, before optimization, some performances (gain and SNR) violate the specification at both nominal and worst-case operating conditions, as illustrated in Table 5, 1. In general, there is a mixed influence of mechanical and electrical parameters on the MEMS-IC performances. For this example, some performances (signal-to-noise ratio) are sensitive to parameters from both domains, others (lower and upper bandwidth) only to mechanical parameters, while others (phase

Table 5. Microphone performance values at nominal and worst-case operation before and after nominal optimization.

Performance	Specification	1. Initial design		2. Nominally optimized design			
		Nominal	Worst-case	a. only electrical	b. only mechanical	c. electrical+ mechanical	
				Worst-case	Worst-case	Nominal	Worst-case
SNR_{1PaRMS} [dB]	> 55	51.15	49.37	52.18	53.72	57.28	55.63
$Gain_{dB_{1kHz}}$ [dB]	> -25	-26.11	-28.41	-26.49	-24.18	-20.86	-23.06
THD_{40g} [%]	< 0.5	0.025	0.035	0.027	0.037	0.019	0.026
PM_{Opamp} [deg]	> 60	68.96	68.08	72.97	63.09	72.44	71.84
BW_{high} [Hz]	> 40k	57.46k	48.4k	48.4k	44.81k	50.15k	46.16k
BW_{low} [Hz]	< 10	8.396	8.577	8.577	4.896	5.266	5.397

margin) mostly to electrical parameters. This mixed influence suggests that optimization should account for design parameters from both domains.

The nominal optimization searches a set of design parameters for which all specifications are fulfilled (at nominal process conditions) [17]. When considering only the electrical design parameters for optimization, after 6 iterations the nominal optimization still doesn't find a point where all specifications are fulfilled. Table 5, 2.a shows the performance values at worst-case operating conditions after optimization, where the SNR and the gain violate the specification. When considering only the mechanical design parameters, the nominal optimization fails as well. After 2 iterations, the SNR still violates the specification at worst-case operating conditions, but the optimization algorithm sees no further improvement from that point. Table 5, 2.b shows the performance values at worst-case operating conditions after optimization, where the SNR violates the specification. However, when design parameters from both domains are subject to optimization, the nominal optimization succeeds in finding a point where all specifications are fulfilled, after only 1 iteration. Table 5, 2.c shows the performance values at nominal and worst-case operating conditions after optimization. All specifications are fulfilled in this case. Considering both parameter types provides additional degrees of freedom which contribute significantly to the success of the optimization. Also, the mechanical and the electrical part are optimized not independently, but in mutual interaction.

The nominally optimized design is not guaranteed to be robust against process variations, because the nominal optimization accounts only for nominal process conditions. However, in this particular case, the worst-case distance is larger than 4-sigma for every performance, making the total yield practically 100%. Since the design is already robust, a yield optimization is not necessary. This is nevertheless not a general situation. The explanation for the small tolerance of performances is that only electrical process variation is considered. This is a limitation not of the presented optimization methodology, but of the available mechanical process data for the MEMS microphone. The mechanical statistical parameters can have a strong impact on the robustness, as it will be illustrated in the next demonstrator example.

7.2 Accelerometer and Readout

The accelerometer is a complete example of MEMS-IC co-optimization, as it includes all parameter types, as shown in Table 2. In addition to the microphone example, mechanical process parameters are present as well. Nominal optimization and then yield optimization are applied on the combined MEMS-IC. At each design stage, diagnosis is used to identify the relevant parameters and performances. The global flow of MEMS-IC co-optimization, with the inclusion of diagnosis, is depicted in Fig. 8.

7.2.1 Nominal Optimization. As in the case of the microphone, the initial accelerometer design does not fulfill all specifications. As shown in Table 6, a, the resonance frequency violates the specification

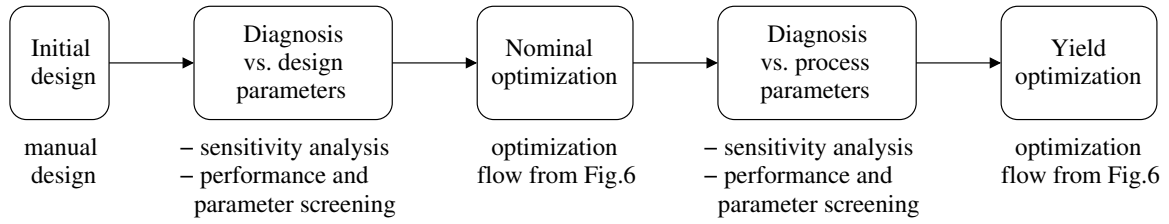


Fig. 8. Global flow of MEMS-IC co-optimization.

Table 6. Accelerometer performance values before and after nominal optimization.

Performance	Specification	Performance values (nominal process conditions)			
		a. Initial design		b. Nominally optimized design	
		Nominal operation	Worst-case operation	Nominal operation	Worst-case operation
<i>FreqRes</i> [Hz]	>15k	14.94k	14.94k	16.11k	16.11k
<i>Gain_mV_g</i> [mV/g]	>20	20.27	19.57	24.51	23.66
<i>THD_40g</i> [%]	<0.5	0.037	0.066	0.049	0.144
<i>PM_Opamp</i> [deg]	>60	61.59	61.32	62.06	61.68

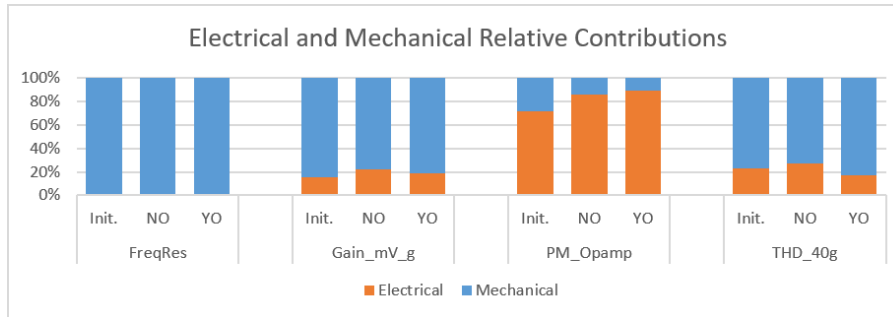


Fig. 9. Relative influence (in %) from the electrical and the mechanical side on the worst-case distance of each accelerometer performance, at the initial (Init.), nominally optimized (NO) and yield optimized (YO) design stage.

at both nominal and worst-case operating conditions, and the gain violates its specification at worst-case operating conditions.

The diagnosis methods from Section 3.2 were applied in order to identify the relevant parameters and performances, at each design stage: initial, nominally optimized and yield optimized. The nominal optimization and the yield optimization were run in WiCkeD, based on the flow presented in Section 6. The diagnosis methods were implemented and run in Matlab, where the worst-case distances β_i and the sensitivities $\nabla_d f_i$ and $\nabla_s f_i$, computed in WiCkeD, were given as inputs.

Diagnosis allows to get insight into the relative influence of the electrical part and the mechanical part on each worst-case distance. Fig. 9 shows, for each performance at each design stage, how much of the influence (in percentage) on the worst-case distance is electrical and how much mechanical. The relative influences were calculated according to eq. (8). The normalized parameter changes $\Delta d_{e,j}^*$, $\Delta d_{m,j}^*$ were taken as 1%.

The resonance frequency depends only on the mechanical properties of the MEMS and is therefore sensitive only to the mechanical parameters, as confirmed by the diagnosis results illustrated in Fig. 9. For this reason, the mechanical design parameters must be subject to optimization, in order to obtain a design that fulfills the specification for all performances, including the resonance frequency. For the other performances, there is a mixed influence from design parameters from both energy domains, according to the diagnosis results from Fig. 9. For the phase margin, the electrical influence dominates, while for the gain and for the THD, the influence is predominantly mechanical. Also, the relative influence is consistent over design stages. The results in Fig. 9 show that both the electrical and the mechanical parts have an important influence overall.

Table 7. Relevant worst-case distances for the accelerometer performances at different design stages

Performance	Initial	Nominally optimized	Yield optimized
<i>Gain_mV_g</i>	-0.413	5.553	5.567
<i>FreqRes</i>	-0.058	1.037	4.992
<i>THD_1PaRMS</i>	4.548	3.006	4.260
<i>PM_Opamp</i>	0.371	0.550	6.418

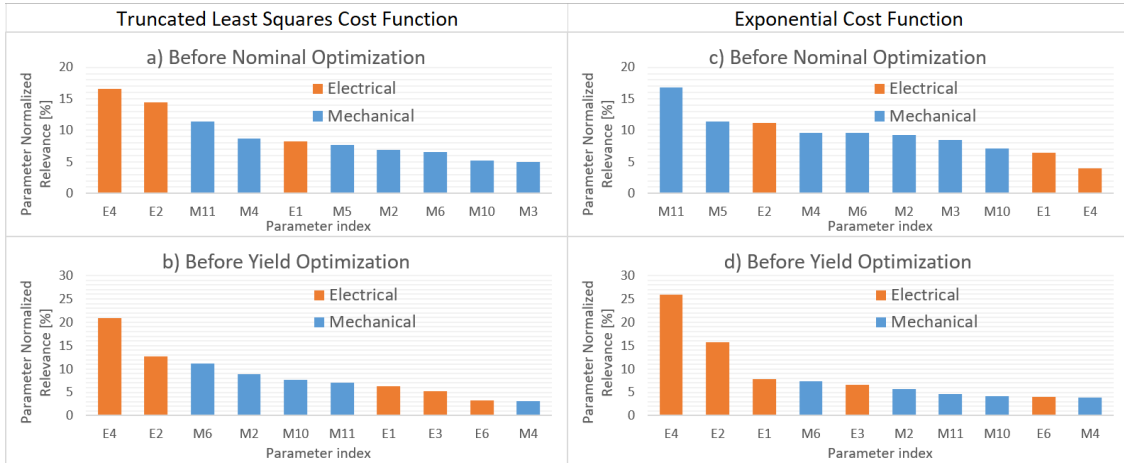


Fig. 10. Most relevant parameters, based on the truncated least squares cost function (a-b) and the exponential cost function (c-d), for the accelerometer: a,c) before nominal optimization, b,d) before yield optimization.

To reduce the computational effort, the parameters with little or no influence are left out, based on the diagnosis results. Diagnosis allows to examine the influence of the design parameters on the system robustness at every design stage. Table 7 shows the worst-case distances for the accelerometer performances at the three design stages. The worst-case distances corresponding to the relevant specifications, according to (12), are given in bold font. The relevance of specifications was determined according to the description from Section 3.2, using (12) and a threshold $\eta = 0.1$. Fig. 10 shows the 10 most relevant parameters at the initial (before nominal optimization) and nominally optimized (before yield optimization) stage, based on the cost functions $Q_{lsq,rel}$ (Fig. 10, a-b) and $Q_{exp,rel}$ (Fig. 10, c-d). The relevance was calculated using the magnitude of (18) and (17), respectively.

An important result regards the consistency with respect to the design stage and to the cost function. First, for both the initial and the nominally optimized design there is a mixed influence of electrical and mechanical parameters, regardless of the cost function. Second, at both stages the most relevant parameters are similar (for both cost functions, 8 parameters are in top 10 at both stages). Third, at each stage the same parameters are in top 10 most relevant for both cost functions, even though the hierarchy and the magnitude of the relevance are slightly different, because of the different expression of the cost function. For this case study, no dominant or redundant parameter or performance was found.

For this demonstrator example, the number of design parameters was reduced from 16 to 13 for the electrical component and from 15 to 11 for the mechanical component, by leaving out the parameters with a relevance below 0.1% for the truncated least squares cost function. Then, nominal optimization was applied to the design. It considered both electrical and mechanical design parameters and succeeded in its goal of finding a set of design parameters for which all specifications are fulfilled at nominal and worst-case operating conditions (Table 6, b). Up to this point, the design has been optimized only for nominal process conditions.

Table 8. Accelerometer performance values before and after yield optimization.

Performance	Specification	Performance values (worst-case operating conditions)			
		a. Nominally optimized		b. Yield optimized	
		Nominal process	Worst-case process (3σ)	Nominal process	Worst-case process (3σ)
<i>FreqRes</i> [Hz]	>15k	16.11k	12.97k	21.63k	17.57k
<i>Gain_mV_g</i> [mV/g]	>20	23.66	22.13	22.7	22.47
<i>THD_40g</i> [%]	<0.5	0.144	0.496	0.225	0.419
<i>PM_Opamp</i> [deg]	>60	61.68	54.89	78.55	72.32

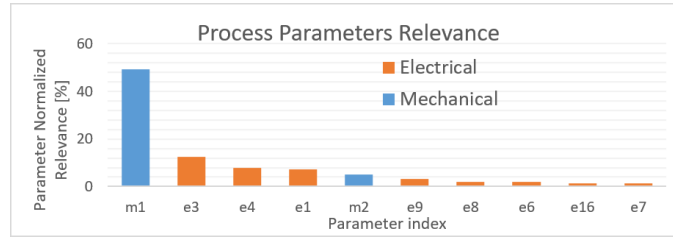


Fig. 11. Accelerometer: relevance of process parameters at the nominally optimized design stage. One mechanical parameter dominates, having almost as much influence as all the other parameters together.

Table 9. Accelerometer worst-case distances before and after yield optimization.

Performance	Worst-case distances [σ] (worst-case operation)			
	a. Nominally optimized	b. Yield optimized		
		i. electrical+mechanical	ii. only electrical	iii. only mechanical
<i>FreqRes</i>	1.037	4.992	1.037	3.156
<i>Gain_mV_g</i>	5.553	5.667	9.487	1.457
<i>THD_40g</i>	3.006	4.260	2.482	5.255
<i>PM_Opamp</i>	0.550	6.418	6.957	1.026

7.2.2 Yield Optimization. A nominally optimized design is usually not maximally robust against process variations. Table 8 shows the performance values at nominal and worst-case process conditions, before and after yield optimization. All values from the table are at worst-case operating conditions. Worst-case process corresponds to a 3-sigma tolerance. The results show that the resonance frequency and phase margin violate the specification at worst-case process conditions (Table 8, a). The yield optimization improves the robustness of the design. After the yield optimization, all specifications are fulfilled at worst-case process conditions (Table 8, b).

To reduce the computational effort, process parameters with insignificant influence were left out before starting the yield optimization, based on diagnosis of performances vs. process parameters at the nominally optimized point. Fig. 11 shows the 10 most relevant process parameters, as calculated using (24), according to the description from Section 3.2. A mechanical parameter is on the first place, having almost 50% relevance, equivalent to being almost as influential as all other process parameters together. To reduce complexity, process parameters with a relevance below 0.1% were left out: both mechanical parameters and 18 out of 56 electrical parameters remained after this.

The mechanical process has a strong impact on the performances, which is the main reason for the low robustness of the nominally optimized design. Illustrative in this sense is also the tolerance of the resonance frequency: this performance, which is sensitive only to mechanical (design and process) parameters, has a 3-sigma tolerance of around 20% (both before and after yield optimization, as shown in Table 8, a. and b).

Table 9 shows the worst-case distances before and after yield optimization. All worst-case distances correspond to worst-case operating conditions. For the nominally optimized design (Table 9, a), the resonance frequency and phase margin have worst-case distances significantly below 3σ , indicating a low robustness, in agreement with the results from Table 8, a. The yield

Table 10. CPU Runtime Comparison for FEA and Circuit-Level Model (Microphone Example)

Analysis	Simulations	1 thread		8 parallel threads	
		FEA	Circuit-level	FEA	Circuit-level
Sensitivity vs. design parameters	90	35 min	9 min	5 min	2 min
Nominal optimization	112	45 min	11 min	11 min	3 min
Sensitivity vs. process parameters	240	1.5 h	23 min	15 min	3 min
Yield analysis	1544	10 h	2.5 h	2 h	27 min

Table 11. CPU Runtime Comparison for a Single Transient Analysis (Microphone Example)

Model	Transient analysis runtime
FEA-based	90s
Circuit-level, new model generated	5s + 4s (model generation from MEMS+)
Circuit-level, old model reused	5s
Basic	4s

optimization tunes the design parameters until all worst-case distances are larger than 4-sigma, as shown in Table 9, b, i, achieving an overall 4-sigma design.

The yield optimization succeeded only when optimizing the electrical and the mechanical part simultaneously. In this case, the worst-case distances after optimization are all larger than 4σ (Table 9, b.i), making the yield practically 100%. Because the resonance frequency depends only on mechanical parameters, yield optimization only with electrical design parameters cannot bring any improvement in the yield associated with this performance, despite improving it for all other performances (Table 9, b.ii). Yield optimization only with mechanical design parameters fails as well. It raises the worst-case distance of the resonance frequency above 3σ , but fails doing this for all performances simultaneously (Table 9, b.iii). The other three performances strongly depend on both types of parameters, as shown by the diagnosis results from Fig. 9, and the number of degrees of freedom is not large enough to succeed in optimizing all four performances.

7.3 CPU Runtime Considerations

Using the circuit-level model for the MEMS part can significantly reduce the CPU runtime of the analysis/optimization. Illustrative in this sense is the demonstrator example of the microphone. Table 10 shows that with the simplified model the speedup achieved is approximately 4 times, regardless whether the analysis was executed on one simulation thread or multiple parallel threads. The runtime was compared for 4 types of analysis/optimization: sensitivity vs. design parameters, nominal optimized, sensitivity vs. process parameters and yield analysis, for one thread and 8 threads.

The runtime comparison between Spectre and MEMS+ is described in Table 11, having as a reference the runtime of a single transient analysis for the microphone example. With the basic model (a variable capacitor), a transient analysis in Spectre takes 4s. In this case, the contribution of the mechanical part is insignificant, as it is made up only of a simple circuit element. With the circuit-level model, the analysis takes 5s. Comparing with the basic model, one can estimate that the mechanical part modeled at circuit level contributes 1s. When an already existing model is reused, MEMS+ is not part of the simulation. When a new circuit-level model needs to be generated, additional 4s are needed, before the start of the transient analysis, for MEMS+ to generate the model. With the FEA-based model, a transient analysis takes 90s. Taking the difference to the basic model, one can estimate that the FEA-based model contributes 86s to the simulation.

The CPU runtime is additionally decreased by exporting a circuit-level MEMS model only when necessary. To illustrate this, the two approaches described in Section 6 and in Fig. 7 are compared in Table 12, for the case study of the MEMS accelerometer, for 4 types of analysis/optimization: sensitivity vs. design parameters, nominal optimization, sensitivity vs. process parameters and

Table 12. CPU Runtime Comparison with and without Model Database (Accelerometer Example)

Analysis	simulations	1 thread				8 parallel threads			
		with database		without database		with database		without database	
		models	time [min]	models	time [min]	models	time [min]	models	time [min]
Sensitivity vs. design parameters	128	16	14	16	14	16	2	68	3
Nominal Optimization	356	142	44	207	46	142	10	311	12
Sensitivity vs. process parameters	232	2	26	4	26	2	3	23	4
Worst-Case	524	235	57	322	67	235	16	454	20

Table 13. Accelerometer Performance Values After Optimization

Performance	FEA-based model	Circuit-level model
<i>FreqRes</i>	21.63k	21.63k
<i>Gain_mV_g</i>	22.7	22.7
<i>THD_40g</i>	0.229	0.225
<i>PM_Opamp</i>	78.55	78.55

worst-case. With the database, better runtimes are achieved, because the minimum number of models is exported (number which is the same regardless of the number of threads). Without database, the number of exported models is larger, as explained in Fig. 7. The larger the number of threads, the more times the export command is executed. The reason is that simulations are distributed in an uncontrolled manner, such that many simulations involving the same model end up on different threads, each thread then exporting the model for itself. Through parallel simulation, the runtime is greatly reduced, but not always proportionally to the number of threads (e.g. for DNO, 4 times for 8 threads). The limitation is caused by sequential dependencies in the iterative algorithms. However, in the sensitivity analysis, where there are no such bottlenecks, the reduction is proportional.

Using the circuit-level model saves a significant amount of runtime without affecting the accuracy. For the microphone, Table 4 shows that the accuracy of the performances of the mechanical part is little affected compared to the FEA-based model, for a wide range of design parameters. For the accelerometer, Table 13 shows the MEMS-IC performances after optimization. The optimization was run with the circuit-level model and the MEMS-IC was then simulated using both models with the optimized parameter values.

8 CONCLUSION

A complete approach to MEMS-IC robustness optimization was presented. All parameters (design, process and operating) of the electrical and the mechanical part were included in the optimization. The optimization methodology was exemplified on two demonstrator examples: a MEMS microphone and a MEMS accelerometer, each with an integrated readout circuit. The optimization succeeded only when simultaneously considering mechanical and electrical parameters. The robustness was strongly influenced by the mechanical process parameters.

A simulation-in-a-loop flow was implemented, based on the tools WiCkeD for optimization, Spectre for continuous electrical simulation and MEMS+ for MEMS simulation. Within this flow, a circuit-level model, exported from MEMS+ and described in VerilogA, is used for the MEMS. Using this model, a CPU time reduction of up to 4 times was achieved. A contribution to the CPU time reduction was also brought by exporting a VerilogA model only when necessary. Any newly exported MEMS model was saved in a database and retrieved from there when needed again, maximizing in this way the efficiency of the optimization flow. A control flow for accessing the database was implemented to support parallel simulation.

Mathematical methods of diagnosis were transferred from the IC level to the extended MEMS-IC level and applied to the MEMS accelerometer. The mixed mechanical and electrical influence on the improvement of the worst-case distances and on the overall yield was illustrated.

ACKNOWLEDGMENTS

The authors would like to thank Arnaud Parent and Gerold Schröpfer from Coventor for the technical support in using MEMS+ and XFAB for the mechanical process data.

This work has been supported in part by the German Federal Ministry of Education and Research (BMBF) within the collaborative project RoMulus under the sponsorship registration #16ES0365.

REFERENCES

- [1] K. J. Antreich, H. E. Graeb, and C. U. Wieser. 1994. Circuit analysis and optimization driven by worst-case distances. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 13, 1 (Jan 1994), 57–71. <https://doi.org/10.1109/43.273749>
- [2] F. Burcea, A. Herrmann, A. Gupta, and H. Graeb. 2017. A new robustness optimization methodology for MEMS-IC systems. In *IEEE SMACD*. 1–4. <https://doi.org/10.1109/SMACD.2017.7981590>
- [3] F. Burcea, A. Herrmann, B. Li, and H. Graeb. 2018. MEMS-IC Optimization Considering Design Parameters and Manufacturing Variation from both Mechanical and Electrical Side. In *IEEE ICECS*. 625–628. <https://doi.org/10.1109/ICECS.2018.8617959>
- [4] Cadence [n. d.]. *Cadence Virtuoso Documentation* (6.1.6 ed.). Cadence.
- [5] Amen Cheng. 2009. *DESIGN OF A READOUT SCHEME OF A MEMS MICROPHONE*. Master’s thesis. Delft University of Technology.
- [6] Coventor [n. d.]. *MEMS+ Reference* (6.0 ed.). Coventor.
- [7] Coventor [n. d.]. *MEMS+ Tutorials* (6.0 ed.). Coventor.
- [8] J. Eckmueller. 1999. *On the Computer-Aided Sizing of Analog Integrated Circuits under Special Consideration of Structural Properties*. Ph.D. Dissertation. Technical University of Munich.
- [9] J. Eckmueller, G. Strube, and H. Graeb. 1996. Diagnosis of Analog Integrated Circuits. In *GME/ITG-Workshop CAD Methods for Analog Circuit Development*.
- [10] Helmut E. Graeb. 2010. *Analog Design Centering and Sizing*. Springer.
- [11] H. E. Graeb, C. U. Wieser, and K. J. Antreich. 1993. Improved Methods for Worst-Case Analysis and Optimization Incorporating Operating Tolerances. In *30th ACM/IEEE Design Automation Conference*. 142–147. <https://doi.org/10.1145/157485.164641>
- [12] A. Herrmann, C. Hielscher, A. Mueller, G. Hoelzer, and H. Graeb. 2017. Realistic worst-case parameter sets for MEMS technologies. In *IEEE SMACD*. 1–4. <https://doi.org/10.1109/SMACD.2017.7981594>
- [13] T. C. C. Kevin et al. 2010. 118-dB dynamic range, continuous-time, opened-loop capacitance to voltage converter readout for capacitive MEMS accelerometer. In *2010 IEEE Asian Solid-State Circ. Conf.* <https://doi.org/10.1109/ASSCC.2010.5716626>
- [14] T. Konishi et al. 2013. A Single-Platform Simulation and Design Technique for CMOS-MEMS Based on a Circuit Simulator With Hardware Description Language. *J. of Microelectromechanical Syst.* 22, 3 (June 2013), 755–767. <https://doi.org/10.1109/JMEMS.2013.2243111>
- [15] C. Lu, M. Lemkin, and B. E. Boser. 1995. A monolithic surface micromachined accelerometer with digital output. *IEEE Journal of Solid-State Circuits* 30, 12 (Dec 1995), 1367–1373. <https://doi.org/10.1109/4.482163>
- [16] David Thomas Martin. 2007. *Design, fabrication, and characterization of a MEMS dual-backplate capacitive microphone*. Ph.D. Dissertation. The Graduate School Of The University Of Florida.
- [17] MunEDA [n. d.]. *WiCkeD User Manual* (6.7 ed.). MunEDA.
- [18] M. Pak, F. V. Fernandez, and G. Dundar. 2017. Optimization of a MEMS accelerometer using a multiobjective evolutionary algorithm. In *2017 14th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*. <https://doi.org/10.1109/SMACD.2017.7981574>
- [19] M. Pak, F. V. Fernandez, and G. Dundar. 2018. A novel design methodology for the mixed-domain optimization of a MEMS accelerometer. *Integration, the VLSI Journal* 62 (Jun 2018), 314–321.
- [20] A. Parent, A. Krust, G. Lorenz, and T. Piirainen. 2015. A novel model order reduction approach for generating efficient nonlinear verilog-a models of mems gyroscopes. In *2015 IEEE International Symposium on Inertial Sensors and Systems (ISISS) Proceedings*. 1–4. <https://doi.org/10.1109/ISISS.2015.7102377>
- [21] M. Pehl. 2012. *Discrete Sizing of Analog Integrated Circuits*. Ph.D. Dissertation. Technical University of Munich.
- [22] H. Toshiyoshi et al. 2013. A mixed-design technique for integrated MEMS using a circuit simulator with HDL. In *Proc. of the 20th Intern. Conf. Mixed Des. of Integr. Circuits and Syst. - MIXDES 2013*. 17–22.
- [23] B. Vernay, A. Krust, G. Schröpfer, F. Pêcheux, and M. Louërat. 2015. SystemC-AMS simulation of a biaxial accelerometer based on MEMS model order reduction. In *2015 Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS (DTIP)*. 1–6. <https://doi.org/10.1109/DTIP.2015.7160987>

ACM Copyright Form and Audio/Video Release

Title of the Work: MEMS-IC Robustness Optimization Considering Electrical and Mechanical Design and Process Parameters

Author/Presenter(s): Florin Burcea (Technical University of Munich); Andreas Herrmann (Technical University of Munich); Bing Li (Technical University of Munich); Helmut Graeb (Technical University of Munich)

Type of material: Paper

Publication: ACM Transactions on Design Automation of Electronic Systems

I. Copyright Transfer, Reserved Rights and Permitted Uses

* Your Copyright Transfer is conditional upon you agreeing to the terms set out below.

Copyright to the Work and to any supplemental files integral to the Work which are submitted with it for review and publication such as an extended proof, a PowerPoint outline, or appendices that may exceed a printed page limit, (including without limitation, the right to publish the Work in whole or in part in any and all forms of media, now or hereafter known) is hereby transferred to the ACM (for Government work, to the extent transferable) effective as of the date of this agreement, on the understanding that the Work has been accepted for publication by ACM.

Reserved Rights and Permitted Uses

(a) All rights and permissions the author has not granted to ACM are reserved to the Owner, including all other proprietary rights such as patent or trademark rights.

(b) Furthermore, notwithstanding the exclusive rights the Owner has granted to ACM, Owner shall have the right to do the following:

(i) Reuse any portion of the Work, without fee, in any future works written or edited by the Author, including books, lectures and presentations in any and all media.

(ii) Create a "Major Revision" which is wholly owned by the author

(iii) Post the Accepted Version of the Work on (1) the Author's home page, (2) the Owner's institutional repository, or (3) any repository legally mandated by an agency funding the research on which the Work is based.

(iv) Post an "Author-Izer" link enabling free downloads of the Version of Record in the ACM Digital Library on (1) the Author's home page or (2) the Owner's institutional repository;

(v) Prior to commencement of the ACM peer review process, post the version of the Work as submitted to ACM ("Submitted Version" or any earlier versions) to non-peer reviewed servers;

(vi) Make free distributions of the final published Version of Record internally to the Owner's employees, if applicable;

(vii) Make free distributions of the published Version of Record for Classroom and

Personal Use;

(viii) Bundle the Work in any of Owner's software distributions; and

(ix) Use any Auxiliary Material independent from the Work.

(x) If your paper is withdrawn before it is published in the ACM Digital Library, the rights revert back to the author(s).

When preparing your paper for submission using the ACM templates, you will need to include the rights management and bibstrip text blocks below to the lower left hand portion of the first page. As this text will provide rights information for your paper, please make sure that this text is displayed and positioned correctly when you submit your manuscript for publication.

Authors should understand that consistent with ACM's policy of encouraging dissemination of information, each work published by ACM appears with a copyright and the following notice:

If you are using Authorized ACM TeX templates, the following code will generate the proper statements based on your rights choices. Please copy and paste it into your TeX file between `\begin{document}` and `\maketitle`, either after or before CCS codes.

```
\setcopyright{acmcopyright}  
\acmJournal{TODAES}  
\acmYear{2019} \acmVolume{1} \acmNumber{1} \acmArticle{1}  
\acmMonth{1} \acmPrice{15.00}\acmDOI{10.1145/3325068}
```

If you are using Word, copy and paste these words in the space provided at the bottom of your first page:

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

Copyright © ACM 2019 1084-4309/2019/MonthOfPublication -
ArticleNumber \$15.00
<https://doi.org/10.1145/3325068>

NOTE: DOIs will be registered and become active shortly after publication in the ACM Digital Library

A. Assent to Assignment. I hereby represent and warrant that I am the sole owner (or authorized agent of the copyright owner(s)), with the exception of third party materials detailed in section III below. I have obtained permission for any third-party material included in the Work.

B. Declaration for Government Work. I am an employee of the National Government of my country and my Government claims rights to this work, or it is not copyrightable (Government work is classified as Public Domain in U.S. only)

Are any of the co-authors, employees or contractors of a National Government?
 Yes No
Country:

II. PERMISSION FOR CONFERENCE TAPING AND DISTRIBUTION (Check A and, if applicable, B)

A. Audio /Video Release

I hereby grant permission for ACM to include my name, likeness, presentation and comments in any and all forms, for the Conference and/or Publication.

I further grant permission for ACM to record and/or transcribe and reproduce my presentation as part of the ACM Digital Library, and to distribute the same for sale in complete or partial form as part of an ACM product on CD-ROM, DVD, webcast, USB device, streaming video or any other media format now or hereafter known.

I understand that my presentation will not be sold separately by itself as a stand-alone product without my direct consent. Accordingly, I give ACM the right to use my image, voice, pronouncements, likeness, and my name, and any biographical material submitted by me, in connection with the Conference and/or Publication, whether used in excerpts or in full, for distribution described above and for any associated advertising or exhibition.

Do you agree to the above Audio/Video Release? Yes No

III. Auxiliary Materials, not integral to the Work

* Your Auxiliary Materials Release is conditional upon you agreeing to the terms set out below.

[Defined as additional files, including software and executables that are not submitted for review and publication as an integral part of the Work but are supplied by the author as useful resources for the reader.]

I hereby grant ACM permission to serve files containing my Auxiliary Material from the ACM Digital Library. I hereby represent and warrant that any of my Auxiliary Materials do not knowingly and surreptitiously incorporate malicious code, virus, trojan horse or other software routines or hardware components designed to permit unauthorized access or to disable, erase or otherwise harm any computer systems or software.

I agree to the above Auxiliary Materials permission statement.

This software is knowingly designed to illustrate technique(s) intended to defeat a system's security. The code has been explicitly documented to state this fact.

IV. Third Party Materials

In the event that any materials used in my presentation or Auxiliary Materials contain the work of third-party individuals or organizations (including copyrighted music or movie excerpts or anything not owned by me), I understand that it is my responsibility to secure any necessary permissions and/or licenses for print and/or digital publication, and cite or attach them below.

- We/I have not used third-party material.
- We/I have used third-party materials and have necessary permissions.

V. Artistic Images

If your paper includes images that were created for any purpose other than this paper and to which you or your employer claim copyright, you must complete Part IV and be sure to include a notice of copyright with each such image in the paper.

- We/I do not have any artistic images.
- We/I have any artistic images.

VI. Representations, Warranties and Covenants

The undersigned hereby represents, warrants and covenants as follows:

- (a) Owner is the sole owner or authorized agent of Owner(s) of the Work;
- (b) The undersigned is authorized to enter into this Agreement and grant the rights included in this license to ACM;
- (c) The Work is original and does not infringe the rights of any third party; all permissions for use of third-party materials consistent in scope and duration with the rights granted to ACM have been obtained, copies of such permissions have been provided to ACM, and the Work as submitted to ACM clearly and accurately indicates the credit to the proprietors of any such third-party materials (including any applicable copyright notice), or will be revised to indicate such credit;
- (d) The Work has not been published except for informal postings on non-peer reviewed servers, and Owner covenants to use best efforts to place ACM DOI pointers on any such prior postings;
- (e) The Auxiliary Materials, if any, contain no malicious code, virus, trojan horse or other software routines or hardware components designed to permit unauthorized access or to disable, erase or otherwise harm any computer systems or software; and
- (f) The Artistic Images, if any, are clearly and accurately noted as such (including any applicable copyright notice) in the Submitted Version.

- I agree to the Representations, Warranties and Covenants
-

DATE: **04/10/2019** sent to florin.burcea@tum.de at **07:04:01**

4 Modeling for Capacitor Array Layout in Charge-Redistribution ADCs

This chapter presents a simple and efficient analytical model of the static non-linearity of charge-redistribution ADCs. The model developed earlier for binary-weighted capacitor ratios is enhanced to generalized capacitor ratios. The model expresses the integral and differential nonlinearity (INL and DNL) as linearized functions of capacitor variations. Three independent and additive sources of capacitor variation are included in the model: systematic, caused by global process gradients, statistical, caused by local, random variations, and parasitic, caused by routing. The model is used to conduct a placement and routing methodology for the capacitors that minimizes the static non-linearity of an ADC across all included sources of variation. In the experimental results, INL and DNL are evaluated using the model, showing the superiority of the proposed layout.

This chapter consists of the following publication:

- Y. X. Ding, F. Burcea, H. Habal and H. Graeb, "PASTEL: Parasitic Matching-Driven Placement and Routing of Capacitor Arrays with Generalized Ratios in Charge-Redistribution SAR-ADCs," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*.

The paper was written under the leading pen of the candidate. The candidate conceptualized the paper. He was responsible for the literature review. The principal technical contribution of the candidate is represented by Sections III and IV of the paper. He conceptualized and developed the linearized analytical model of INL and DNL for charge-redistribution ADCs with generalized capacitor ratios and implemented this model in Matlab for simulating the ADC non-linearity. He initiated the placement strategy presented in Section V through the idea of grouping the capacitors with a small number of units in the center of the array and dispersing the capacitors with a large number of units outside the center. He initiated the routing strategy presented in Section VI through the idea of matching the parasitic capacitances. The placement and routing methodology and algorithms were developed and the experimental results were obtained by the author Y. X. Ding in a Master thesis under the co-supervision of the candidate.

This work was published directly in a journal paper; there is no prior conference publication. This paper is a continuation of a previous publication in the same journal, not explicitly included in this thesis, but first-authored by the candidate:

- F. Burcea, H. Habal and H. E. Graeb, "A New Chessboard Placement and Sizing Method for Capacitors in a Charge-Scaling DAC by Worst-Case Analysis of Non-linearity," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 9, pp. 1397-1410, Sept. 2016.

4 Modeling for Capacitor Array Layout in Charge-Redistribution ADCs

In this paper, the initial linearized analytical model of INL and DNL, combined with the statistical model of capacitive mismatch and the model of systematic, global capacitive mismatch, and the chessboard placement method were developed by the candidate. In the new paper, the results are summarized in Sections III-A, IV-A and V-A,B.

PASTEL: Parasitic Matching-Driven Placement and Routing of Capacitor Arrays with Generalized Ratios in Charge-Redistribution SAR-ADCs

Ye X. Ding, Florin Burcea, Husni Habal, and Helmut Graeb, *Fellow, IEEE*

Abstract—The charge-redistribution data converter topology is constructed from an array of parallel-connected capacitors with a specific capacitor ratio, and is a fundamental component of SAR-ADC architectures. This paper presents complementary placement and routing algorithms for such capacitor arrays with the goal of nonlinearity minimization in the target SAR-ADC architecture. Starting from a state-of-the-art static nonlinearity model and its accompanying placement method, generalizations are made for both binary- and nonbinary-weighted capacitor ratios. Taking the generalized nonlinearity model and capacitor routing complexity into consideration, an improved placement method was developed, augmented with a matching-based routing technique driven by routing length adjustment. Analytical results compare the new methodology to the state-of-the-art, achieving superior resolution, linearity, and area.

Index Terms—charge-redistribution, SAR-ADC, capacitor matching, layout optimization, non-linearity minimization

I. INTRODUCTION

The Analog-to-Digital Converter (ADC) is a crucial component in modern integrated circuits. It is responsible for the acquisition of analog signals, followed by their conversion into digital form for further processing. A popular architecture is the Successive Approximation Register (SAR)-ADC, which offers numerous advantages, including low power consumption [1]–[5]. The basic SAR-ADC consists of a signal sample-and-hold circuit, a digital-to-analog converter (DAC) realized with a capacitor array, a voltage comparator, and an SAR with related digital control logic, as depicted in Fig. 1.

This paper targets capacitor array design, which is a primary contributor to the accuracy of the ADC transfer curve. In [1] it was shown that the ADC static performance metrics (integral and differential nonlinearity, *INL* and *DNL*) depend on the variations in the values of the capacitors that form the DAC, as well as suitable matching of these capacitor values, in order to minimize nonlinearity according to equations that are, in turn, specific to the used SAR-ADC architecture. Three principle sources of capacitor variation are recognized:

- local random variations modeled statistically (sta);
- systematic process biases and gradients (sys); and

Ye X. Ding and Husni Habal are with Infineon Technologies AG, Am Campeon 1–15, Neubiberg 85579, Germany.

Florin Burcea and Helmut Graeb are with the Chair of Electronic Design Automation, Technical University of Munich, Munich 80333, Germany.

©2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, including reprinting/republishing this material for advertising or promotional purposes, collecting new collected works for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. Authors version. Definitive version published in IEEE Trans. CAD, <https://ieeexplore.ieee.org/document/8695813>.

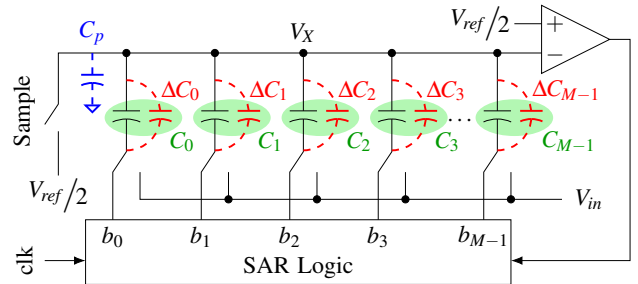


Fig. 1. A schematic of a charge-redistribution SAR-ADC, with its variations: top plate-to-substrate parasitic (blue), and individual capacitor variation (red).

- interconnection parasitic capacitances (par), bearing a systematic influence.

The impact of statistical and systematic (placement-based) variations can be reduced by proper placement; for the former, dispersion, and for the latter, symmetry. Only the parasitic capacitances appearing in parallel with the nominal capacitances affect static linearity. Their effects can be reduced by suitable placement and routing methods to best match interconnect parasitics, which serves as the target topic of this paper.

Typical capacitor unit values lie in the range of ones to tens of fF based on the most recent publications [6]–[8], owing to aggressive trends in technology downscaling and the demand for reduced area and power. The sources of capacitor variation do not scale down by the same factors as the nominal unit values, so the variable component can be large enough to dominate the nominal (intended) value. Careful attention must be paid in order to mitigate performance degradation.

Consequently, a large number of past works concerned with or related to this topic [1], [4], [5], [9]–[13] placed emphasis on the necessity of capacitor matching due to its impact on performance, motivating ongoing research and development.

A linear model was developed in [1] to analytically determine *INL* and *DNL* as linear functions of the three aforementioned sources of capacitor variation, and a small nonlinear error term. The final equations for evaluating these metrics are simplified to a simple matrix product of the linear sum of the variations from these three sources.

For the rest of this paper, the following terms shall be used:

- *(Capacitor) array*: The DAC formed by $C_0 \cdots C_{M-1}$ as shown in the schematic (Fig. 1).
- *Array capacitor*: Each capacitor in this array.
- *Unit (capacitor)*: Each of the individual capacitor cells that are connected in parallel to form an array capacitor.
- *(Capacitor) matrix*: The resulting grid formed by all unit capacitors after they are placed.

- *Net*: The electrical connection between the bottom plates of all the unit capacitors of a single array capacitor.
- *Parasitics*: The parasitic capacitances between net routing (of each capacitor bottom plate) and the top plates.

All row vectors and matrix rows are presented with the most and least significant bits on the left and right, respectively; for column vectors and matrix columns, the most and least significant bits are presented at the top and bottom, respectively.

For each *array capacitor*, the nominal routing problem is to connect its *unit capacitors'* bottom plates with its corresponding exit pin. The problem is extended to also consider routing parasitics and their impact on ADC performance metrics.

This paper is organized as follows. Section II reviews existing literature and outlines the new contributions from this work. Section III introduces the binary capacitor ratio and extends it for generalized (e.g. nonbinary-weighted) capacitor ratios. Section IV reviews the previously developed nonlinearity model for binary-weighted ratios, then extends it for generalized ratios. Section V introduces a new placement method modified from the method in [1], and depicts its algorithmic implementation. Section VI details the novel routing style designed for the new placement method, and its algorithmic implementation. Analysis and experimental results are presented in Section VII. Section VIII concludes the paper.

II. STATE-OF-THE-ART AND NEW CONTRIBUTIONS

Optimization of capacitance ratio accuracy in capacitor arrays is a popular research topic. Many papers attempted to achieve this by maximizing the overall correlation between capacitances with respect to an exponential spatial correlation model [1], [5], [13]–[16]. Since capacitance ratio accuracy was the general objective, the resulting solutions could be suboptimal for a different objective function, such as the linearity of a particular SAR-ADC architecture.

In [1], a simple analytical model was derived for a charge-redistribution ADC with binary-weighted capacitor ratios, describing the *INL* and *DNL* as linear expressions of capacitor variations. Combining this with the exponential spatial correlation model produced an optimal placement methodology for minimizing the impact of statistical (random fluctuations) and systematic (process gradients) variations on ADC nonlinearity (as measured by *INL* and *DNL*). The resulting placement style, named “chessboard”, complied with general layout guidelines for this problem [17]: coincidence, symmetry, dispersion, and compactness, with dispersion being the key criterion.

However, the fundamental drawback with the chessboard placement style is that the routing problem was not considered. Interconnection parasitics between the top (common to all array capacitors) and bottom plates (individual to each array capacitor) appear in parallel with the nominal capacitances (depicted by the red-colored parallel capacitors in Fig. 1), additively contributing to their values. Mismatch of these parasitics results in a large mismatch in overall capacitance. Moreover, maximal dispersion implies maximal routing complexity.

The work outlined in [5] considers the spatial correlation model as well, producing a novel placement method. A custom routing technique was not developed, rather one adapted from previous works was used. Furthermore, the resultant *INL* and *DNL* were evaluated only for binary-weighted architectures.

In [2], [9], [10], a strong coupling between the placement and routing styles can be seen; each was designed with the other in mind. It was concluded that the use of “globally distributed, locally connected” placements provided the best trade-off between placement-based mismatch and parasitics.

Routing was also considered in [11], where the entire task was formulated as an Integer Linear Programming problem. Finally, Ho et al. presented a routing technique by controlling routing lengths in order to precisely maintain the capacitance ratio [12]. Inter-routing coupling was also considered, based on the distances between routing. In both [11] and [12], the goals were limited to reducing variation in general, without being directed towards metric (e.g. nonlinearity) improvement.

In all of [2], [5], [10]–[12], the results of evaluating *INL* and *DNL* were presented, however complete definitions were used and not with the linearized, simple model from [1].

This paper provides the following contributions:

- A novel hybrid placement algorithm to place capacitor arrays with generalized weightings is presented. The algorithm considers the tradeoff between capacitor dispersion and routability not considered in the chessboard placement algorithm described in [1].
- A novel routing method to complement the proposed hybrid placement is introduced. Routing lengths (and by extension capacitances) are iteratively adjusted to reach the overall goal of minimizing ADC *INL* and *DNL*. This method is inspired by [12], where a complex optimization problem is presented to precisely match capacitor ratio values. However, in this paper, the algorithm is different and more simplified (wire length adjustment). Moreover, the optimization goal is different (to minimize overall *INL* and *DNL*), achieved using a different routing solution.
- A linearized analytical model of *INL* and *DNL* for any SAR-ADC with a generalized capacitor ratio is defined. The model is then used to evaluate three sources of nonlinearity independently, as well as in connection with each other: statistical, systematic and parasitic. The analytical model guides the resultant layout automation algorithm, named “PASTEL”, towards minimization of *INL* and *DNL* to desirably low levels, by consideration of (a) capacitor variation caused by placement-related mismatch (due to placement dispersion), and (b) the components caused by interconnection parasitics (due to routing).

III. THE GENERALIZED CAPACITOR RATIO AND REDUNDANCY IN ADCS

The SAR-ADC includes a DAC comprised of a capacitor array, as per Fig. 1. Let c be the ordered array of M DAC capacitors, and C_T the total sum of capacitor values; with respect to Fig. 1, each effective array capacitor C_j (shaded in green) is the parallel combination (and hence, sum) of a nominal capacitance (black) and its associated variation (red):

$$c^T = [C_{M-1}, C_{M-2}, \dots, C_1, C_0], \quad C_T = \sum_{j=0}^{M-1} C_j. \quad (1)$$

At any step in the conversion process, the voltage V_X at the DAC output (the capacitor top plate) is compared to $V_{ref}/2$ [18]. Initially, $V_X = V_{ref}/2$ is sampled on the top plate,

and the input voltage, V_{in} , is sampled on the bottom plates of C_{M-1} to C_0 . The total charge introduced into the array is:

$$Q = \left(\frac{V_{ref}}{2} - V_{in} \right) \cdot C_T + \frac{V_{ref}}{2} \cdot C_p, \quad (2)$$

where C_p is the top plate-to-substrate parasitic from Fig. 1.

Throughout the conversion, the bottom plates are driven by the SAR logic and thus connected to either V_{ref} or ground. Let the bit vector \mathbf{b} denote the ordered M bottom-plate connections in Fig. 1; 1 and 0 represent V_{ref} and VSS respectively:

$$\mathbf{b}^T = [b_{M-1}, b_{M-2}, \dots, b_1, b_0], \quad b_j \in \{0, 1\}. \quad (3)$$

After sampling, the top plate is disconnected and remains floating. Thus, the total charge stays constant, expressed as:

$$Q = \sum_{j=0}^{M-1} (V_X \cdot C_j - V_{ref} \cdot b_j \cdot C_j) + V_X \cdot C_p \stackrel{(1),(3)}{=} V_X \cdot C_T + V_X \cdot C_p - V_{ref} \cdot \underbrace{\mathbf{b}^T \cdot \mathbf{c}}_{\sum_{j=0}^{M-1} b_j \cdot C_j}. \quad (4)$$

The total charge, Q , remains constant between (2) and (4). Equating the expressions and rearranging the terms gives:

$$V_{in} = \left(\frac{V_{ref}}{2} - V_X \right) \cdot \left(1 + \frac{C_p}{C_T} \right) + V_{ref} \cdot \left(\frac{\mathbf{b}^T \cdot \mathbf{c}}{C_T} \right). \quad (5)$$

From (5), a comparison of $V_{ref}/2$ with V_X is equivalent to a comparison of V_{in} and $V_{ref} \cdot \mathbf{b}^T \cdot \mathbf{c} / C_T$:

$$V_{ref}/2 > V_X \iff V_{in} > V_{ref} \cdot \mathbf{b}^T \cdot \mathbf{c} / C_T = V(\mathbf{b}). \quad (6)$$

$V(\mathbf{b})$ in (6) defines the current state of the ADC. The SAR logic output, \mathbf{b} , can be an intermediate or the final conversion result. To distinguish the final result, an index i is used, so that $\mathbf{b}(i)$ is the binary representation of i . $V(\mathbf{b}(i))$ then denotes the voltage at which the ADC transfer curve (discrete output level) changes from level $i-1$ to i . This is the ADC counterpart to the output voltage of a DAC and hence bears the same expression, but without a top-plate parasitic capacitance (C_p) term.

Although present in (2) and (4), C_p does not appear in the expression of $V(\mathbf{b}(i))$. This parasitic capacitance does not influence the conversion thresholds, because its bottom plate is connected to the (always grounded) substrate, and the top plate voltage V_X converges to the initial value, $V_{ref}/2$ [19].

(6) holds irrespective of any variation source, modeled as a part of \mathbf{c} . Ideal voltages are obtained by using nominal capacitances, replacing \mathbf{c} with \mathbf{c}_0 , and C_T with C_{T0} , such that:

$$\mathbf{c} = \mathbf{c}_0 + \Delta \mathbf{c}, \quad \Delta \mathbf{c}^T = [\Delta C_{M-1}, \Delta C_{M-2}, \dots, \Delta C_0], \quad (7)$$

$$C_T = C_{T0} + \Delta C_T, \quad \Delta C_T = \sum_{j=0}^{M-1} \Delta C_j. \quad (8)$$

\mathbf{c}_0 depends on the architecture and shall be defined later.

A. Binary-Weighted Capacitor Ratios

In the binary-weighted case there are $M = N + 1$ array capacitors, where N of them are ordered in increasing powers of 2, such that the capacitor indices range from 0 to N . Taking C_u to represent the fixed unit capacitance, the capacitances are equal to $C_u \cdot 2^{j-1}$, $1 \leq j \leq N$, with one additional capacitor with a grounded bottom plate forming the nominal device of

\hat{C}_0 . Note that symbols specific to the binary ratio bear a hat (^) above them, so as to distinguish them as a special case.

This grounded capacitor is introduced so that the total sum of the capacitances is $C_{T0} = C_u \cdot 2^N$, allowing a normalized resolution of $1/2^N$ instead of $1/(2^N - 1)$. It also fixes \hat{b}_0 at 0, so $\hat{b}_{M-1}\hat{b}_{M-2}\dots\hat{b}_1$ is the binary representation for the integer i , $0 \leq i \leq 2^N - 1$. The *INL* is defined for this range, but the *DNL* excludes $i = 0$. The vector representations of the binary ratio ($\hat{\mathbf{n}}$) and the (nominal) array capacitor values ($\hat{\mathbf{c}}_0$) are:

$$\hat{\mathbf{n}}^T = [\hat{n}_{M-1}, \hat{n}_{M-2}, \dots, \hat{n}_1, \hat{n}_0] = [2^{N-1}, \dots, 2, 1, 1], \quad (9)$$

$$\hat{\mathbf{c}}_0^T = C_u \cdot \hat{\mathbf{n}}^T \stackrel{(9)}{=} C_u \cdot [2^{N-1}, \dots, 2, 1, 1]. \quad (10)$$

Note that $\hat{\mathbf{n}}$ includes the multiplicities of all capacitors, because the input voltage is sampled on all capacitors.

B. Generalized Capacitor Ratios

For generalized ratios, there are M array capacitors, more than the number of bits N . An equivalent vector \mathbf{n} can also be defined in the same way as (9), and consequently the nominal array capacitances would be $C_u \cdot n_j$, $0 \leq j \leq M - 1$. Since the sum of the multiplicities is still 2^N , the resolution V_{LSB} is $V_{ref}/2^N$, and (10) is generalized to:

$$\mathbf{c}_0^T \stackrel{(10)}{=} C_u \cdot \mathbf{n}^T = C_u \cdot [n_{M-1}, n_{M-2}, \dots, n_1, n_0]. \quad (11)$$

The specific generalized capacitor ratio in the 12-bit case study used for the purposes of results in this paper is:

$$\mathbf{n}^T = [1868, 1012, 552, 303, 168, 90, 48, 26, 14, 8, 4, 2, 1]. \quad (12)$$

Definitions such as (10) can be extended to their generalized forms. Consider, for example, a 4-bit ADC ($N = 4$). With binary-weighted ratios, it would consist of capacitors \hat{C}_4 to \hat{C}_0 , with $\hat{\mathbf{n}} = [8, 4, 2, 1, 1]$. Each integer between 0 and $2^4 - 1$ can be uniquely represented as a sum of the capacitor multiplicities (excluding the rightmost element, corresponding to \hat{C}_0). For example, $13 = 1 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1$. In other words, 1101 is the representation of the integer 13. Considering the iterative conversion process, the conversion result, in a 4-bit unsigned binary representation, is obtained after 4 iterations.

However, a 4-bit ADC can also be realized with a (nonbinary) ratio $\mathbf{n} = [8, 3, 2, 2, 1]$. In this case, certain integers have multiple representations as a sum of the capacitor multiplicities (here, all capacitors between C_4 to C_0 are considered in the sum). Hence, 13 can be written as $1 \cdot 8 + 1 \cdot 3 + 1 \cdot 2 + 0 \cdot 2 + 0 \cdot 1$, as $1 \cdot 8 + 1 \cdot 3 + 0 \cdot 2 + 1 \cdot 2 + 0 \cdot 1$, or as $1 \cdot 8 + 0 \cdot 3 + 1 \cdot 2 + 1 \cdot 2 + 1 \cdot 1$. In other words, the integer 13 can be represented by either 11100, 11010, or 10111. The conversion process is still iterative, and so the conversion result is a 5-bit value obtained after 5 iterations. This takes one more iteration than the binary case, as well as an additional bit for representing integers in the same range, 0 and $2^4 - 1$. The bit weighting is certainly no longer binary. For example, the second bit from the left has a weight of 3 instead of 4. To map the conversion results 11100, 11010, and 10111 to 1011 (the common binary form of 13), additional digital logic for decoding is needed.

Table I depicts the intermediate and final bit patterns, as well as the comparison thresholds (representing voltages normalized to the resolution V_{LSB}) for this 4-bit ADC example. If each comparator decision were fault-free (and the capacitor

TABLE I
OUTPUT LEVELS FOR A NONBINARY ADC WITH $\mathbf{n} = [8, 3, 2, 2, 1]^T$.

Intermediate Pattern			Output	i_x	Threshold
1□□□	11□□	111□	1111□	1111	16₁
			1110□	1110	15₁
			1110□	1110	14₂
		1110□	1110	13₃	
		1101□	1101□	14	
		1100□	1101□	13	
	10□□	101□	1100□	1100	12₂
			1100□	1100	11₃
			1100□	1100	11
		1011□	1011□	13	
		1011□	1011□	12	
		1010□	1010□	11	
100□	100□	1010□	1010	10₂	
		1010□	1010	10	
		1010□	1010	11	
	1001□	1001□	11		
	1001□	1001□	10		
	1000□	1000□	9		
0□□□	01□□	011□	1000□	1000	8₂
			1000□	1000	8
			1000□	1000	7
		0111□	0111□	8	
		0111□	0111□	7	
		0110□	0110□	6	
	010□	010□	0110□	0110	6₂
			0110□	0110	5₃
			0110□	0110	5
		0101□	0101□	6	
		0101□	0101□	5	
		0100□	0100□	4	
00□□	001□	0100□	0100	4₂	
		0100□	0100	3₃	
		0100□	0100	3	
	0011□	0011□	5		
	0011□	0011□	4		
	0010□	0010□	3		
000□	000□	0010□	0010	2₂	
		0010□	0010	2	
		0010□	0010	3	
	0001□	0001□	3		
	0001□	0001□	2		
	0000□	0000□	1		

TABLE II
COMPARISON OF THE CONVERSION PROCESS OF INPUT i USING TABLE I, WHERE $13 < i < 14$, WITHOUT AND WITH A BIT ERROR IN THE PROCESS.

Step	Nominal Behavior			Behavior due to a Bit Error		
	Decision	Result	Pattern	Decision	Result	Pattern
1	$i > 8$	True	1□□□	$i > 8$	True	1□□□
2	$i > 11$	True	11□□	$i > 11$	False	10□□
3	$i > 13$	True	111□	$i > 10$	True	101□
4	$i > 15$	False	1110	$i > 12$	True	1011
5	$i > 14$	False	1110 (13)	$i > 13$	True	1011 (13)

ratio had nominal values without variations, to be discussed in Section IV), the final bit pattern would be one of the entries in bold in the *Output* column of Table I, with the corresponding value of i in the next column. These patterns shall be referred to as the *nominal* patterns. Out of several conversion results corresponding to one integer, the nominal one is the uppermost occurrence in the table. Since the conversion starts from the most significant bit, this bit is the first to be recorded as part of the final result. For the input $13 V_{LSB} < V_{in} < 14 V_{LSB}$, for example, its result of 13 bears the nominal representation 11100, not 11010, nor 10111.

The equivalent of Table I for the 4-bit binary-weighted case is constructed in the same way. For the *Intermediate Pattern* columns, since there are now N bits rather than M , and considering $N = M - 1$ in this example, they are the same as either the top or bottom halves of the table, but without the most significant bit. The i values represent each integer from 0 to 15 ($2^N - 1$) without duplicates, and the corresponding *Threshold* column values range from 1 to (also) $2^N - 1$.

The following example demonstrates the benefit of redun-

dancy. Table II shows all conversion steps for two cases: when all comparator decisions are correct (nominal behavior), and when one decision is erroneous. In the first case, the nominal result, 11100, is obtained. In the second case, a wrong comparator decision is taken in step 2, such that the second bit is 0 instead of (the correct) 1. From here, the process follows a different branch of the decision tree, compared to the nominal case. This conversion process continues with no further errors, producing 10111, which also corresponds to 13. Due to the redundancy, the final result is correct, despite the error. It can be concluded that from an algorithmic perspective, the use of redundancy lowers efficiency.

A high-frequency ADC demands a fast and precise comparator, but such a comparator is power-hungry, yet power efficiency is a fundamental requirement in modern designs. Therefore, it is more convenient to use a slower and less precise comparator, which also reduces power consumption. The erroneous decisions of this comparator are compensated by the redundancy in the conversion algorithm. Power reduction is thus paid for by additional conversion cycle(s) [5], [20].

Interestingly, SAR-ADCs with redundancy can actually operate at higher frequencies than the binary variant, as the cycle period can be decreased by purposely permitting *incomplete* voltage settling per cycle [3], [20]. This is possible due to their ability to recover from bit errors in the conversion process. Furthermore, because the algorithm implicitly corrects any comparator offset, further analog calibration is not needed [21]. Nevertheless, the sum of the capacitor multiplicities (the total numbers of units) stays the same (2^N) for the same number of bits (N), in order to maintain the same resolution. Consequently, total capacitor matrix area does not increase.

The advantages above are related to ADC dynamic behavior. However, the scope of this paper is to study the ADC static linearity. The previous discussion was intended to: justify the importance of these architectures due to the advantages of redundancy, and to review the principles of ADC architectures with generalized capacitor ratios, as a prelude to the derivation of an analytical reduced-order model in Subsection IV-B.

IV. ANALYTICAL REDUCED-ORDER MODEL OF NONLINEARITY ERRORS

A. Binary-Weighted Architectures

The main source of static nonlinearity for an ADC's transfer curve lies in the capacitor ratio's inaccuracies.

In [1], an analytical model was derived, describing the *INL* and *DNL* of a charge-redistribution DAC as linear expressions of the capacitor variations. Applying the derivation from [1], the *INL* and *DNL* for a binary-weighted charge-redistribution ADC can be written in terms of the capacitor variations $\Delta\mathbf{c}$:

$$INL(i) = (1 + \Delta C_T/C_{T0})^{-1} \cdot \overline{INL}(i), \quad (13)$$

$$\overline{INL}(i) = \mathbf{b}(i)^T \cdot (\mathbf{I}_{N+1} - 2^{-N} \cdot \mathbf{n} \cdot \mathbf{1}_{N+1}^T) \cdot \Delta\mathbf{c}/C_u; \quad (14)$$

$$DNL(i) = (1 + \Delta C_T/C_{T0})^{-1} \cdot \overline{DNL}(i), \quad (15)$$

$$\overline{DNL}(i) = (\Delta\mathbf{b}(i)^T - 2^{-N} \cdot \mathbf{1}_{N+1}^T) \cdot \Delta\mathbf{c}/C_u. \quad (16)$$

$\mathbf{1}_x$ is a column vector of x 1's and \mathbf{I}_x the $x \times x$ identity matrix.

INL(i) and *DNL*(i) were written in this form in order to separate them into a product of a nonlinear term $(1 + \Delta C_T/C_{T0})^{-1}$, and a linear expression with respect to capacitor variations,

Δc . If capacitor variations are small, then the nonlinear term can be neglected [1], referred to as the *linearity assumption* from here on. This linearity assumption must be checked with the presentation of any analysis results.

For each i , $\overline{INL}(i)$ is different, because it depends on $\mathbf{b}(i)$, which is unique for each i . There are $2^N - 1$ unique \overline{INL} components in total. Defining $\mathbf{B} = [\mathbf{b}(1); \mathbf{b}(2); \dots; \mathbf{b}(2^N - 1)]$ sets up the vector form of \overline{INL} :

$$\overline{INL} = \mathbf{B}^T \cdot (\mathbf{I}_{N+1} - 2^{-N} \cdot \mathbf{n} \cdot \mathbf{1}_{N+1}^T) \cdot \Delta \mathbf{c} / C_u. \quad (17)$$

The \overline{DNL} is based on the difference between a code and its preceding code, i.e. a transition. $\Delta \mathbf{b}(i)$ of (16) is defined as a *difference pattern*, representing the bit changes that occur during a transition from $i-1$ to i , that is (for the binary case):

$$\Delta \hat{\mathbf{b}}(i) = \mathbf{b}(i) - \mathbf{b}(i-1), \quad 1 \leq i \leq 2^N - 1. \quad (18)$$

In [1] it was shown that for an N -bit binary-weighted architecture there are only N unique difference patterns $\Delta \mathbf{b}(i)$, denoted as $\Delta \hat{\mathbf{b}}_k$, $1 \leq k \leq N$, grouped in the matrix $\Delta \hat{\mathbf{B}}$:

$$\Delta \hat{\mathbf{B}}^T = \begin{bmatrix} \Delta \hat{\mathbf{b}}_1^T \\ \Delta \hat{\mathbf{b}}_2^T \\ \vdots \\ \Delta \hat{\mathbf{b}}_{N-1}^T \\ \Delta \hat{\mathbf{b}}_N^T \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 & -1 & 0 \\ & & & & \vdots & & & \\ 0 & 1 & -1 & \dots & -1 & -1 & -1 & 0 \\ 1 & -1 & -1 & \dots & -1 & -1 & -1 & 0 \end{bmatrix}. \quad (19)$$

Consequently, there are only N unique \overline{DNL} expressions, denoted as \overline{DNL}_k , $1 \leq k \leq N$, grouped in the vector \overline{DNL} :

$$\overline{DNL}_k = \left(\Delta \hat{\mathbf{b}}_k^T - 2^{-N} \cdot \mathbf{1}_{N+1}^T \right) \cdot \Delta \mathbf{c} / C_u, \quad (20)$$

$$\overline{DNL} = \left(\Delta \hat{\mathbf{B}}^T - 2^{-N} \cdot \mathbf{1}_N \cdot \mathbf{1}_{N+1}^T \right) \cdot \Delta \mathbf{c} / C_u. \quad (21)$$

B. Architectures with Generalized Capacitor Ratios

A similar linearized analytical model with respect to capacitor variations can be derived for ADCs with nonbinary capacitor ratios. Consider an N -bit architecture with M capacitors. In the sampling phase, the situation is unchanged. This leads again to (2), where the vector $\mathbf{b}(i)$, defined in (3), describes the bottom-plate connections. Now however, $M-1$ does not necessarily equal N , b_0 is no longer tied to 0, and $\mathbf{b}(i)$ relates to i differently: $i = \mathbf{b}(i)^T \cdot \mathbf{n}$. The binary representation of i is $b_{M-1}b_{M-2} \dots b_1b_0$ (including b_0). Nevertheless, at any conversion step (4), and consequently (6) hold, as they are independent of the bit pattern and capacitor values.

The different mapping $\mathbf{b}(i) \mapsto i$ causes the major differences between the binary and nonbinary architectures. They will be explained using the 4-bit example from Subsection III-B.

1) *Bit Patterns*: Given multiple $\mathbf{b}(i)$ representations for i , let $p(i)$ denote the *number* of possible representations for each i . Generalizing $\mathbf{b}(i)$ from (3), $\mathbf{b}_x(i)$, $1 \leq x \leq p(i)$ denotes the indexed representations of i , for example:

$$\mathbf{b}_1(2) = [0, 0, 1, 0, 0]^T, \quad \mathbf{b}_2(2) = [0, 0, 0, 1, 0]^T; \quad p(2) = 2. \quad (22)$$

2) *Static Patterns*: The linearized model presented refers only to the static behavior, i.e. it is assumed that all comparator decisions are correct. Under this assumption, it can be shown that certain bit patterns cannot occur. If $\mathbf{b}(i_1) > \mathbf{b}(i_2)$, but $i_2 > i_1$, then the pattern $\mathbf{b}(i_2)$ cannot occur under static conditions and/or in the presence of small variations. An example is $\mathbf{b}(13) = [1, 0, 1, 1, 1]^T$, since $\mathbf{b}(13) = [1, 0, 1, 1, 1] < [1, 1, 0, 0, 1] = \mathbf{b}(12)$, as shown in Table I. This pattern can occur only when the conversion process is affected by dynamic errors; such a case is illustrated in the right-hand side of Table II. Hence, the pattern $[0, 0, 0, 1, 1]$ for $i = 3$ is not allowed. $p'(i)$, $p'(i) \leq p(i)$ is defined to denote only the number of patterns *valid in static conditions* (depicted in bold in the *Threshold* column of Table I), for example:

$$\mathbf{b}_1(3) = [0, 1, 0, 0, 0]^T, \quad \mathbf{b}_2(3) = [0, 0, 1, 0, 1]^T; \quad p'(3) = 2. \quad (23)$$

3) *Transition Voltages*: It can be shown that, from multiple bit patterns for i , only one will determine (under static conditions) the transition voltage from $i-1$ to i , namely the pattern with the smallest $\mathbf{b}(i)^T \cdot \mathbf{c} / C_u$. Consider the capacitor ratio $\mathbf{c} / C_u = [7.8, 3.1, 2.1, 1.9, 1]^T$. Then, $\mathbf{b}_1(2)^T \cdot \mathbf{c} / C_u = 2.1$, and $\mathbf{b}_2(2)^T \cdot \mathbf{c} / C_u = 1.9$. Therefore, $\mathbf{b}_2(2)$ will characterize the transition of $i-1$ to i . If $\mathbf{c} / C_u = [7.8, 3.1, 1.9, 2.1, 1]$, then $\mathbf{b}_1(2)^T \cdot \mathbf{c} / C_u = 1.9$, $\mathbf{b}_2(2)^T \cdot \mathbf{c} / C_u = 2.1$, and $\mathbf{b}_1(2)$ will characterize the transition.

Since the minimum is not a linear function, the approach of conservatively considering all possible transitions shall be taken, in order to retain linearity. We define α as the *number of valid static patterns that determine the transition voltages*:

$$\alpha = \sum_{i=1}^{2^N} p'(i). \quad (24)$$

For the 4-bit example in Table I, α is 23; for the 12-bit architecture described by \mathbf{n} in (12), α is 4459. All transition voltages $V(\mathbf{b}_k)$, $1 \leq k \leq \alpha$, can be obtained with (6), where \mathbf{b}_k represents all of the static patterns. Because $V(\mathbf{b}_k)$ has the same form as in the binary-weighted case, the \overline{INL} components are calculated similarly, obtaining (13), with the linear part also taking on the form of (14):

$$\overline{INL}_k = \mathbf{b}_k^T \cdot (\mathbf{I}_M - 2^{-N} \cdot \mathbf{n} \cdot \mathbf{1}_M^T) \cdot \Delta \mathbf{c} / C_u, \quad 1 \leq k \leq \alpha. \quad (25)$$

Vectorizing all values of \mathbf{b}_k as the matrix \mathbf{B} , the full set of \overline{INL}_k equations is hence:

$$\overline{INL} = \mathbf{B}^T \cdot (\mathbf{I}_M - 2^{-N} \cdot \mathbf{n} \cdot \mathbf{1}_M^T) \cdot \Delta \mathbf{c} / C_u. \quad (26)$$

The number of components has increased from $2^N - 1$ to α , because of the conservative approach of considering all possible transitions. Also, the \mathbf{b}_k vectors are different from the $\mathbf{b}(i)$ vectors from (14) because of the differing architectures.

4) *Difference Patterns*: Difference patterns are defined as the *element-wise subtraction of two (consecutive) transition patterns*. With the existence of multiple possible representations of i and having identified the transition patterns, the difference patterns can be obtained, for calculating the \overline{DNL} . Each possible representation of i shall be denoted as i_x (as per the corresponding column in Table I), enumerating the occurrences of multiple patterns for the same value of i . For the i corresponding to the *nominal* output pattern (in bold font), its value of x , is equal to $p(i)$.

For $1 \leq i \leq 2^N - 1$, the $DNL(i)$ is calculated based on $V(\mathbf{b}(i))$ and $V(\mathbf{b}(i-1))$. However, i and $i-1$ could have multiple representations, so they are enumerated using y and z : i_y , $1 \leq y \leq p'(i)$, and $(i-1)_z$, $1 \leq z \leq p'(i-1)$. The conservative approach of considering all possible combinations will be taken, in order to retain linearity. Therefore, starting from the original definition of the DNL , the equivalent of $DNL(i)$ becomes $DNL(i_y, (i-1)_z)$:

$$DNL(i_y, (i-1)_z) = \frac{[V(\mathbf{b}_y(i)) - V(\mathbf{b}_z(i-1))] - V_{LSB}}{V_{LSB}}. \quad (27)$$

As per the binary-weighted architecture, the expression of $DNL(i_y, (i-1)_z)$ can be linearized, obtaining a result similar to (16), in which $\Delta\mathbf{b}(i)$ is replaced by $\Delta\mathbf{b}(i_y, (i-1)_z)$:

$$\overline{DNL}(i_y, (i-1)_z) = (\Delta\mathbf{b}(i_y, (i-1)_z)^\top - 2^{-N} \cdot \mathbf{1}_M^\top) \cdot \Delta\mathbf{c}/C_u, \quad (28)$$

$$\Delta\mathbf{b}(i_y, (i-1)_z) = \mathbf{b}_y(i) - \mathbf{b}_z(i-1), \quad 1 \leq i \leq 2^N - 1. \quad (29)$$

For example, in the case where $i = 3$, there are two different static patterns for both $i = 3$, and $i-1 = 2$, giving $2 \times 2 = 4$ possible transitions: $(3_1, 2_1)$, $(3_2, 2_1)$, $(3_1, 2_2)$, $(3_2, 2_2)$.

The size of a single difference pattern $\Delta\mathbf{b}(i_y, (i-1)_z)$ is $1 \times M$, corresponding to the number of capacitors in the array. We define β as the total number of such difference patterns:

$$\beta = \sum_{i=1}^{2^N} p'(i) \cdot p'(i-1). \quad (30)$$

Continuing the examples, the 4-bit example has a β of 32. Its difference patterns are:

$$\begin{aligned} \Delta\mathbf{b}(1, 0)^\top &= [0, 0, 0, 0, 1], \\ \Delta\mathbf{b}(2_1, 1)^\top &= [0, 0, 0, 1, -1], \\ \Delta\mathbf{b}(2_2, 1)^\top &= [0, 0, 1, 0, -1], \\ &\dots \\ \Delta\mathbf{b}(14, 13_2)^\top &= [0, 0, 0, 0, 1], \\ \Delta\mathbf{b}(15, 14)^\top &= [0, 0, 0, 1, -1], \\ \Delta\mathbf{b}(16, 15)^\top &= [0, 0, 0, 0, 1]. \end{aligned} \quad (31)$$

\mathbf{n} from (12) has a β of 4823.

5) *Unique Difference Patterns*: Just like in the binary-weighted case, identical difference patterns exist. For example, it can be seen that $\Delta\mathbf{b}(1, 0) = \Delta\mathbf{b}(14, 13_2) = \Delta\mathbf{b}(16, 15)$ and $\Delta\mathbf{b}(2_1, 1) = \Delta\mathbf{b}(15, 14)$.

Thus, it is enough to consider only the unique patterns, greatly reducing the number of DNL components. The unique patterns shall be enumerated using k as $\Delta\mathbf{b}_k$, so $\Delta\mathbf{B}$ vectorizes the set of $\Delta\mathbf{b}_k$. In the case of generalized capacitor ratios, $\Delta\mathbf{B}$ no longer has a regular pattern, and must be calculated with programmatic aid. The number of unique difference patterns is now more than the number of bits. For the 4-bit example from Table I, there are 7 unique patterns, giving $\Delta\mathbf{B}$:

$$\Delta\mathbf{B}^\top = \begin{bmatrix} \Delta\mathbf{b}_1^\top \\ \Delta\mathbf{b}_2^\top \\ \Delta\mathbf{b}_3^\top \\ \Delta\mathbf{b}_4^\top \\ \Delta\mathbf{b}_5^\top \\ \Delta\mathbf{b}_6^\top \\ \Delta\mathbf{b}_7^\top \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 1 & -1 & -1 & -1 & 0 \end{bmatrix}. \quad (32)$$

For \mathbf{n} in (12), there are 19 unique difference patterns, and $\Delta\mathbf{B}$ is calculated to be:

$$\Delta\mathbf{B}^\top = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 1 & -1 & -1 & 0 & -1 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 0 & -1 & -1 & 0 & 0 & -1 & -1 & 0 \\ 0 & 1 & -1 & -1 & 0 & -1 & -1 & 0 & -1 & 0 & -1 & 0 & 0 \\ 1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 0 & -1 & -1 & 0 & -1 & -1 & 0 & -1 & 0 & -1 \end{bmatrix}. \quad (33)$$

Using this enumerated form, (28) becomes:

$$\overline{DNL}_k = (\Delta\mathbf{b}_k^\top - 2^{-N} \cdot \mathbf{1}_M^\top) \cdot \Delta\mathbf{c}/C_u. \quad (34)$$

We define γ as the number of unique difference patterns, so the set of all \overline{DNL}_k values is given by:

$$\overline{DNL} = (\Delta\mathbf{B}^\top - 2^{-N} \cdot \mathbf{1}_\gamma \cdot \mathbf{1}_M^\top) \cdot \Delta\mathbf{c}/C_u. \quad (35)$$

V. CAPACITOR PLACEMENT

A. Sources of Variation

As explained in section I, the three sources of variation investigated and modeled in this work are the statistical ($\Delta\mathbf{c}^{sta}$), systematic ($\Delta\mathbf{c}^{sys}$), and parasitic ($\Delta\mathbf{c}^{par}$) variations. The respective strategies applied to mitigate them are: maximizing unit dispersion, placement symmetry (e.g. adopting a common-centroid style), and unit clustering. Note that the mitigation techniques for $\Delta\mathbf{c}^{sta}$ and $\Delta\mathbf{c}^{par}$ oppose each other; this issue is addressed later in Subsection V-E.

B. Chessboard Placement Method

In [1], the analytical model served as the starting point to the derivation of a new ‘‘chessboard’’ placement method, concluding that in order to minimize the impact of random capacitor mismatch, σ_{DNL}^2 is to be minimized. This work only considered the placement, and as such, the objective was to minimize the impact of $\Delta\mathbf{c}^{sta}$ and $\Delta\mathbf{c}^{sys}$ on the DNL .

The rules for common-centroid layout are followed when considering the design of the placement: coincidence, symmetry, dispersion and compactness [17]. Coincidence and symmetry are obtained by orienting the axes of symmetry with respect to the center of the matrix. For the chessboard placement, all units obey these rules, except the two least significant capacitors \hat{C}_1 and \hat{C}_0 , since they have odd multiplicities. However, both being 1, their impact is insignificant.

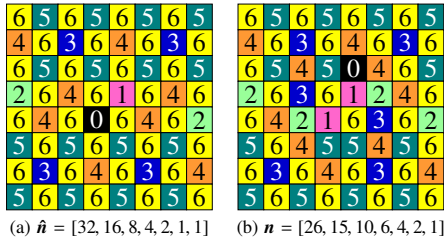


Fig. 2. Chessboard placements for $N = 6$: (a) Binary [1], (b) Nonbinary (Subsection V-C).

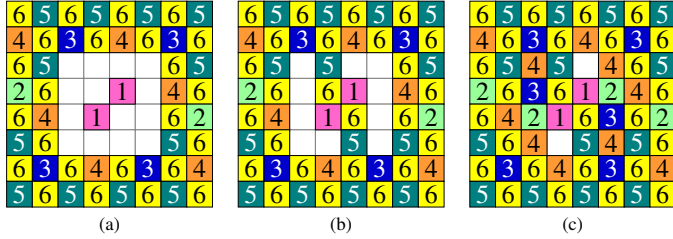


Fig. 3. The steps of the conversion process from a chessboard placement (Fig. 2a) for a binary capacitor ratio, into one for a nonbinary ratio (Fig. 2b).

A depiction of the chessboard placement for $N = 6$ is given in Fig. 2a. It can be seen from the alternating (chessboard) placement of the units, most prominently \hat{C}_N , that the level of dispersion is maximal, since individual unit capacitors are spread out as much as possible. Naturally, by using a square-shaped matrix, maximal compactness is achieved.

C. Modified Chessboard Placement for Nonbinary Ratios

The chessboard placement method is designed to fit a binary-weighted ratio, and hence is limited in this aspect. However, with the use of a nonbinary ratio a pure chessboard placement (i.e. with maximal dispersion) can no longer be achieved. A generalization of the chessboard placement for nonbinary ratios is necessary, so a new placement method must be devised, that retains the qualities of the chessboard method, yet still accommodates the changes in the ratio. A chessboard placement for an arbitrarily chosen nonbinary ratio n is depicted in Fig. 2b. Here, the total sum of n is equal to that of \hat{n} , but the individual multiplicities have changed.

The steps of the new method are depicted in Fig. 3. From Fig. 2a, excess unit capacitors are removed to form an empty square in the center, and C_1 is populated since its placement is fixed, giving Fig. 3a. In order to form the square, too many unit capacitors may have been removed; these are then repopulated from the center, giving Fig. 3b. Incomplete unit capacitors are then populated in “local chessboards” within the empty positions, giving Fig. 3c. Finally, the capacitors with odd multiplicities are spilled in the remaining empty positions, giving the final placement in Fig. 2b.

D. Minimization of the INL and DNL

The main intention of the placement method is to minimize the nonlinearity metrics INL and DNL . From [1], any operation that modifies one affects the other in the same way, and so this objective can be reduced to only minimizing the DNL , since it has fewer values. From (20), because 2^{-N} is small,

the expression is dominated by $\Delta\hat{B}$. Consider the (binary) case where $k = 10$:

$$\begin{aligned} DNL_{10} &\stackrel{(20)}{=} (\Delta\hat{b}_{10}^T - [2^{-N}, \dots, 2^{-N}]) \cdot \Delta c / C_u \\ &\stackrel{(19)}{\approx} [0, 0, 1, -1, \dots, -1, 0]^T \cdot \Delta c / C_u \\ &\propto [0, 0, 1, -1, \dots, -1, 0]^T \cdot [\Delta C_{M-1}, \Delta C_{M-2}, \dots, \Delta C_0] \\ &= \Delta C_{10} - \Delta C_9 - \Delta C_8 - \dots - \Delta C_1. \end{aligned} \quad (36)$$

The result from (37) tells us that the DNL is approximately proportional to the matrix product $\Delta\hat{b}_k^T \cdot \Delta c$. Since the goal is to minimize nonlinearity and hence reduce $|DNL|$ as much as possible, the total variation for net k must equal the sum of all variations for nets with indices $< k$. Since the sum of the lower-indexed capacitor multiplicities is $n_k - 1$, the simplest way to perform matching is to ensure that the variations for all the affected (individual) units are identical. The approximation in (36) is only used to describe the matching process; for calculations, the complete definitions are used.

Qualitatively, this means that each row of $\Delta\hat{B}$ indicates the indices of the array capacitances that are to be matched with each other; ΔC for each array capacitor corresponding to a non-zero entry of $\Delta\hat{b}_k$ must be matched. This result is also valid for the nonbinary case. Since routing variations are a component of Δc , this notion is also followed when routing.

The chessboard style fulfills the symmetry and dispersion criteria, adhering to the matching conditions such as (37), and hence is designed to minimize the worst-case DNL^{sta} and DNL^{sys} , justified by the results presented in [1].

From (36), it can also be seen that reducing C_u results in increased sensitivity; larger mismatches between the components of Δc values would become amplified further.

E. The Hybrid Placement

Since the chessboard placement offers the best (lowest) DNL^{sta} , it should be retained where possible. However, as identified earlier, the maximized dispersion of chessboard placement directly minimizes its routability, since unit capacitors are as separated from each other as possible.

In Fig. 7 of [1], it was shown that across a number of placement styles, the most significant (MSB, highest-indexed) capacitors consistently influence the largest worst-case components of DNL^{sta} ($3\sigma_{DNL^{sta}}$). In turn, the (unnecessary) dispersion of the least significant (LSB, lowest-indexed) capacitors can be sacrificed in order to improve their ease of routing.

The MSB capacitors clearly contribute the most to parasitics, primarily due to the amount of routing they require. However, its individual units are not affected as much as those of the less significant. For example, if a track contributes Δc_0 to both C_0 and C_N , C_0 's units are affected by $\Delta c_0/n_0 = \Delta c_0$, while C_N 's units are affected by $\Delta c_0/n_N \ll \Delta c_0$. In order to preserve the advantages that the dispersion brings without a significant sacrifice in routing variations, the MSB capacitors shall retain their chessboard placement.

Targeting nonlinearity minimization, a new *hybrid* placement style was developed that balances the qualities of the chessboard style for the MSB capacitors (low DNL^{sta} and DNL^{sys}) with the reduction of routing complexity using a grouped style for the LSB capacitors. Use of a common-centroid placement means symmetry is ensured, so the DNL^{sys} is maintained to be as low as possible.

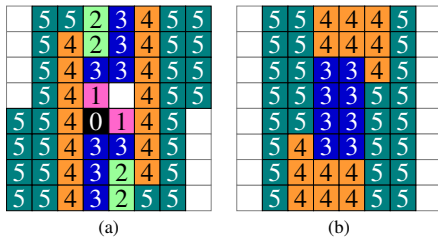


Fig. 4. (a) New grouped placement of size 8×8 . (b) Modified grouped placement as described in Subsection V-F.

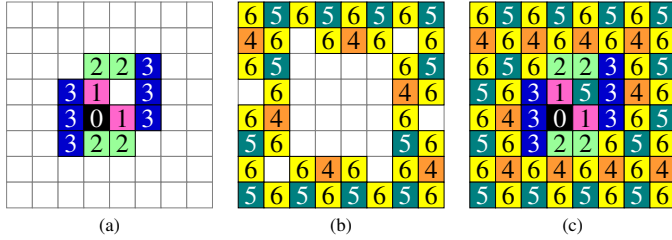


Fig. 5. Combining (a) a modified grouped placement of size 4, and (b) a chessboard placement with excess units removed, to form (c) a hybrid placement for \mathbf{n} in Fig. 2b. The modified C_X placement is omitted due to the smaller multiplicities in \mathbf{n} , where X has been determined to be 4. Note that the modifications suggested in Subsection V-F are omitted due to the smaller multiplicities in \mathbf{n} .

1) *The Grouped Placement*: We first introduce a new placement style, shown in Fig. 4a, that maintains the coincidence and symmetry qualities. Compactness is also evident from inspection. Dispersion, however, is the criterion that has been subject to change. Since there are two clusters of units for each array capacitor, this is the least dispersed a placement can be, while still retaining the other common-centroid qualities. Consequently, this placement has an *increased DNL^{sta}*.

Since the sum of the capacitor multiplicities does not fit exactly into a square, there are hence unpopulated positions; these are filled by other units during the placement process.

2) *Construction*: The hybrid placement is a square region of the same size as the chessboard placement. It consists of a square subregion in the center, placed in the grouped style, which is surrounded by capacitors placed in the chessboard style. First, the least significant array capacitor C_X not populated in the grouped style is determined. The size of the grouped subregion is the smallest square with even dimensions that fits *all* unit capacitors of the array capacitors from C_{X-1} to C_0 . Excess spaces are filled by the units of more significant array capacitors. Hence, the choice of X directly affects the size of the grouped region, and vice versa. An analysis of *DNL^{sta}* in Subsection VII-A determines the selection of X .

The process outlined above and described in Fig. 5 refers to the nonbinary case. The conversion for the binary architecture is similar, but results in more clustered units.

After placing the grouped region (Fig. 5a) in the center of the chessboard, the multiplicities are corrected: any grouped units are removed from the outer region (Fig. 5b), and then in these newly-removed positions the chessboard units are inserted, maintaining a chessboard-like structure, to form Fig. 5c.

F. Modifications to Placement Based on Initial Routing

Two changes were made to the hybrid placement considering the results detailed in Subsection VII-A. These results are summarized here to complete the overview of the placement.

The grouped placement is modified so the units are moved towards the center. For example, starting from Fig. 4a, C_3 would newly occupy the centermost region, as per Fig. 4b.

The units of C_X are changed to form “rings” surrounding the grouped subregion. Excess units are then distributed along the two (vertically) centermost rows, forming *side branches*.

Finally, this modified grouped placement is combined into the chessboard placement and the affected chessboard capacitors are rearranged to produce the complete hybrid placement.

It was recognized that the array capacitors fall into four different categories of placement styles, and therefore their unit capacitors would be routed correspondingly. Applying this categorization we rewrite (11) as:

$$\begin{aligned} \mathbf{c}_0^T &= [\mathbf{c}_{MSB} | C_X | \mathbf{c}_{group} | \mathbf{c}_{LSB}] \\ &= [C_N, \dots, C_{X+1} | C_X | C_{X-1}, \dots, C_3 | C_2, \dots, C_0]. \end{aligned} \quad (38)$$

Based on the results in Subsection VII-A, the most feasible choice for the 12-bit architecture described by \mathbf{n} in (12) is $X = 8 = N - 4$. From here, the (nominal) hybrid placement shall refer to this case. This corresponds to a hybrid placement with a size 14 grouped region for the nonbinary-weighted case; for the binary variant, this applies to a size 12 grouped region.

G. Algorithmic Implementation of the Placement Method

A PLACEMENT algorithm produces the placement, which itself executes the following subalgorithms: CHESSBOARD_NONBIN, GROUPED, and HYBRID.

For $N = 6$, $M = 7$, $\mathbf{n} = [26, 15, 10, 6, 4, 2, 1]$, and $X = 4$, the outputs of the above subalgorithms are depicted in Figs. 2b, 5a, and 5c respectively.

1) CHESSBOARD_NONBIN: *Nonbinary Chessboard*: Produces a chessboard-like placement \mathbf{P} for the nonbinary capacitor ratio specified in \mathbf{n} , and is not required for the binary case. It is described in Algorithm 1.

2) GROUPED: *Grouped Subregion*: Produces a grouped-style placement \mathbf{G} given X . Since this placement is bottom left-top right symmetric, all written steps are for the left half of the subregion. Any placements are also diagonally duplicated for the region below the axis of symmetry, i.e. where $\mathbf{P}[i][j] = k$, $\mathbf{P}[\dim - 1 - i][\dim - 1 - j] = k$ is repeated implicitly. The GROUPED algorithm is simple enough that it will be described textually.

First, a matrix \mathbf{G} for the grouped region is created. C_3 is distributed in the two centermost columns of \mathbf{G} . Population starts from the top of the center-left column and “snakes” towards the bottom left corner: it continues downwards, and when the edge is reached, it moves one position to the left and continues upwards, and so on. When population of \mathbf{c}_{group} and \mathbf{c}_{LSB} is complete, empty margins on all four sides are added. C_X units are then populated in these empty positions.

3) HYBRID: *Hybrid Placement*: Produces a hybrid placement \mathbf{H} and is described in Algorithm 2. For the binary ratio, it combines the output of GROUPED with the original chessboard placement. For the nonbinary ratio, it combines the outputs of the two previous subalgorithms.

Algorithm 1 Chessboard-style placement generator for the nonbinary architecture.

$P = \text{CHESSBOARD_NONBIN}(N, n, \hat{n})$
 $P = \text{CHESSBOARD}(N)$ \triangleright Generate pure chessboard
for $k := N : 3$ **step** -1 **where** $\hat{n}(k) > n(k)$
 \triangleright Remove all units of k in a square from the center with side dimension $d_1 = 2\lfloor\sqrt{\hat{n}(k) - n(k)}/2\rfloor$
 \triangleright Remove every second unit of k from the center until the edges of the square with side dimension $d_2 = 2\sqrt{2}^{j'}$, where j' is the largest net where $\hat{n}(j) < n(j)$
 \triangleright If too many were removed, repopulate back by spiralling out from center, only placing where they were originally removed from.
while *population incomplete*
 \triangleright Generate a smaller chessboard based on number of unpopulated positions, starting at j' . If j' is fully populated, generate using $j' - 1$, and so on.
 \triangleright Interleave the smaller chessboard in P ; place only in empty positions. If ratio reached, populate the next largest and valid unit (no units next to another of the same unit).

Algorithm 2 Combines P and G by placing G in the center of P and correcting the capacitor .

$P = \text{HYBRID}(N, n, P, G, X)$
 \triangleright Remove all values $\leq X$
 \triangleright Populate c_{LSB} since they are fixed
 \triangleright Insert G in the center of P
 \triangleright Populate C_X in the empty spaces of the two rows at $\text{dim}/2 - 1$ and $\text{dim}/2$, $\text{dim} = \sqrt{2^N}$
 \triangleright Replace all missing C_N back to where they were removed from in the original chessboard placement.
 \triangleright Populate remaining capacitors, i.e. $c_{MSB} \setminus C_N$. All steps are for the top half of the matrix. Any placements from here are duplicated for the bottom half, i.e. where $P[i][j] = k$, $P[\text{dim} - 1 - i][\text{dim} - 1 - j] = k$ is repeated implicitly.
for $k := N - 1 : X + 1$ **step of** -1
 $\delta_0 = \delta = n[k] - \text{COUNT}(P == k)$
 \triangleright Only consider rows with empty positions; the specific rows with empty positions will change as certain iterations fill their empty positions. Place only at empty places, then $\delta = 1$.
while $\delta > 0$
if $k \bmod 2 == 1$ \triangleright Odd capacitors
 \triangleright Place at next odd position only on odd rows
else \triangleright Even capacitors
if $(\delta_0 - \delta) \bmod 2 == 0$ \triangleright Even iterations
 \triangleright Place at next position on odd rows
else \triangleright Odd iterations
 \triangleright Place at next even position on even rows
 \triangleright If there are leftovers, populate at next position, searching across rows, starting from the top.

VI. PARASITIC MATCHING-BASED ROUTING METHOD

The routing is separated into two phases: the first must be run once for minimal (but complete) connectivity; the second is repeatedly run, but only after a parasitic extraction is performed, in order to iteratively adjust parasitics using parameters extracted from the preceding routing pass.

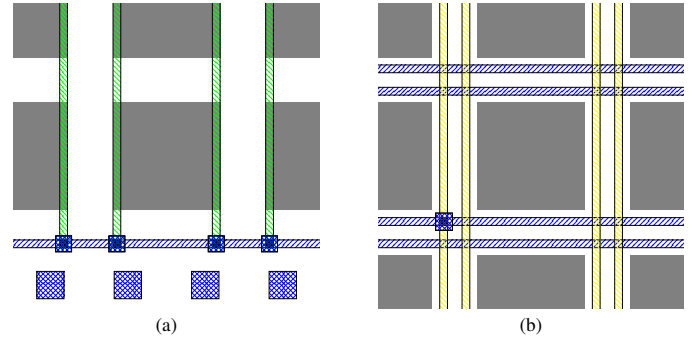


Fig. 6. Depictions of routing methodology: (a) Top plate routing and exit pins. (b) Routing channels within the gaps between the unit capacitors.

A. Routing (Nominal)

1) *Top Plate*: Since all top plates are connected together, routing is trivial and dependent on the geometry of the unit capacitor. A main (horizontal) trunk is constructed along the bottom of the array, with (vertical) subtrunks connecting all unit capacitors in one column to the main trunk. Visually this resembles a comb, as depicted in Fig. 6a.

For the bottom plate routing, subtrunks follow the style shown in Fig. 6b. In order to balance area whilst ensuring adequate connectivity, two routing channels were deemed necessary. Hence, there is sufficient spacing between unit capacitors to allow this. The comb structure from the top plate routing is retained. Finally, a via is depicted for inter-layer connections, and connections from the subtrunks to the unit capacitors are omitted since they are technology-specific.

2) c_{MSB} : The proposed matched routing style for c_{MSB} can accommodate up to 4 nets. Since the routing technique is a parasitic mismatch-driven routing technique, this style is retained even for smaller N . The categorization from (38) can be appended with the condition $C_X = C_{N-4}$, but will retain its original definition to remain generic.

Routing for these nets is implemented with a horizontal trunk across the top of the matrix. Vertical subtrunks run to the bottom of the matrix, in the channels of every gap. In odd-indexed gaps, the trunks are connected to nets C_N and C_{N-1} . In even-indexed gaps, they are connected to the other two nets.

3) C_X : For C_X , the units surrounding the grouped region form a ring and hence are connected to each other directly. The side branches are then connected to the outermost ring, before the trunk is finally brought to the opposing outer edge, then down and across to the exit pin.

4) c_{group} : From opposite corners, a tree is constructed amongst units of the same net, always with the smallest routing distance. This routing is mirrored across the two halves.

Connections between unit capacitors and trunks, or other unit capacitors (in the case of the grouped capacitors), are naturally dependent on the geometry of the unit capacitor itself. The routing of the grouped capacitors hence is also dependent on this, as it primarily involves direct connections.

For the smallest two capacitors in this category, their placements within the very center of the matrix cause them to be highly susceptible to parasitics. Their routing is hence carefully controlled by bringing them to opposite outer edges, then following that edge down and across to the exit pin.

For the other capacitors, their connections to the corresponding exit pins are performed with the shortest path.

5) c_{LSB} : Due to the small number of unit capacitors and their intended placements at the edge of the matrix, these are routed directly to the exit pins with the shortest paths.

B. Routing (Parasitic Compensation)

With the first pass of routing completed, parasitic extraction is performed to identify the parasitic capacitances between the top and bottom plates of each array capacitor. Extracted values from the first pass are used to perform a comparison across all nets. These results are used as the starting point for further, repeated passes of routing.

The aim of the iterated routing process is to adjust the lengths of the routing and hence the coupled parasitics between its routing tracks, and either the top plate or VSS , in order to match the parasitics affecting every array capacitor.

Since the nets of c_{MSB} are the most sensitive and hence should be prioritized in the matching process, they are analyzed first in order to reduce the number of iterations. For these nets, parasitics are added by copying their vertical trunks on a different metal layer. Reduction of parasitics is achieved by adding parallel tracks connected to VSS .

Although routing of the trees within the grouped regions is not subjected to large parasitics, the routing that connects these regions to the exit pins is vulnerable to the parasitics from the more complex routing. To resolve this, the exit routing for c_{MSB} is purposely laid close to the substrate.

C. Algorithmic Implementation of the Routing Method

Routing is performed to achieve the layouts shown in Fig. 6. Both top plate and c_{MSB} routing styles follow the style of Fig. 6a. Algorithm 3 describes the algorithm for nominal routing, necessary for complete connectivity and hence basic working functionality, Algorithm 4 depicts how the parasitics are parametrically controlled, and Algorithm 5 depicts the complete place and route procedure.

VII. EXPERIMENTAL RESULTS

The experimental results were produced using commercial tools. The abstract placement generation was done in MATLAB. The actual circuit synthesis was performed using Virtuoso Layout Suite L User Guide Product Version IC6.1.7. The parasitics were extracted with Cadence Quantus QRC Extraction 18.1.

A. Selection and Feasibility of Placements

In order to verify that the hybrid placement can be justifiably used instead of the chessboard placement, the changes in \overline{DNL}^{sta} and Δc^{par} were evaluated against the chessboard placement, shown in Fig. 7. Results for Δc^{sys} are omitted due to their insignificance compared to the other sources. Unit capacitors with dimensions of $2.35 \times 2.35 \mu\text{m}$ and capacitance $C_u = 12.913 \text{ fF}$ were used. The worst-case value ($3\sigma_{\overline{DNL}^{sta}}$, as defined in [4]) and its components were calculated for varying sizes of the grouped region. In all cases, both horizontal and vertical spacing between unit capacitors were fixed at $1 \mu\text{m}$.

Algorithm 3 Deterministically route all capacitors to achieve bare minimum connectivity.

```

L = ROUTE(P):
  ▷ Base trunk from left to right edge below bottom edge
  for  $i := 0 : \text{dim} - 1$ 
    ▷ Route top plate
    ▷ Subtrunks both side of column, base trunk  $\rightarrow$  top
    ▷ Connect from base trunk to exit pin
    for  $k := N : C_{N-3}$ 
      ▷ Route  $c_{MSB}$ 
      ▷ Base trunk from left to right edge above top edge
      for  $i := 0 : \text{dim} - 2$  step of 2
        if  $i \bmod 2 == 0$ 
          ▷ Even:  $N, N - 1$ 
          ▷ Subtrunk right channel of column  $i \rightarrow N$ 
          ▷ Subtrunk left channel of column  $i+1 \rightarrow N - 1$ 
        else if  $i \bmod 2 == 1$ 
          ▷ Odd:  $N - 2, N - 3$ 
          ▷ Subtrunk right channel of column  $i \rightarrow N - 2$ 
          ▷ Subtrunk left channel of column  $i+1 \rightarrow N - 3$ 
        ▷ Connect rightmost subtrunk to exit pin  $k$ 
        for  $k := X : 3$ 
          ▷ Route  $C_X$  and  $c_{group}$ 
          ▷ Perform on left half first, then repeat for right half
          ▷ Connect units of  $k$  on bottom row horizontally
          ▷ Connect units of  $k$  down each column vertically
          if  $k == X$ 
            ▷ Connect units on side regions, then  $\rightarrow$  exit pin  $X$ 
          else if  $5 \leq k \leq X - 1$ 
            ▷ Direct exit along channel closest to exit pin  $k$ 
          else if  $3 \leq k \leq 4$ 
            ▷ Go to side edge (left for 3, right for 4), down the
            edge to the row of exit pins, go across to exit pin  $k$ 
            ▷ Route  $c_{LSB}$  direct  $\rightarrow$  exit pins

```

Algorithm 4 Based on factors for each net, control the amount of parasitics on that net.

```

PARASITIC_COMPENSATION(L,  $N$ , f):
  ▷ Remove any compensation-related routing
  for  $k := 0 : 2$ 
    ▷  $c_{LSB}$ 
    ▷  $f[k]$  controls distance between routing to  $V_X$  trunk
  for  $k := 3 : X - 1$ 
    ▷  $c_{group}$ 
    if  $f[k] < 0.0$ 
      ▷  $VSS$  on either side of vertical trunk, length  $\sim f[k]$ 
    else if  $f[k] > 0.0$ 
      ▷  $V_X$  on either side of vertical trunk, length  $\sim f[k]$ 
  if  $f[X] < 0.0$ 
    ▷  $C_X$ 
    ▷  $VSS$  on either side of side units, length  $\sim f[X]$ 
  else if  $f[X] > 0.0$ 
    ▷  $V_X$  on either side of side units, length  $\sim f[X]$ 
  for  $k := X + 1 : N$ 
    ▷  $c_{MSB}$ 
    ▷ Duplicate nominal routing on the next vertical layer.
    Subtrunks are populated inwards from the outer edges. The
    number of subtrunks is controlled by  $f[k]$ , such that the final
    subtrunk may not span the full height of the matrix.

```

1) \overline{DNL}^{sta} : Due to the statistical nature of these variations, the $3\sigma_{\overline{DNL}^{sta}}$ values are evaluated for each component of the \overline{DNL}^{sta} , sweeping the size of the grouped region. These components refer to each of the γ elements of the vector that results from the evaluation of (35). Two trends are clearly visible from Fig. 7. Firstly, the DNL components with higher

Algorithm 5 Full algorithm of placement, routing, and parasitic compensation.

PASTEL(N, M, n, X):
 $P = \text{PLACEMENT}(N, M, n, X)$ \triangleright Subsection V-G
 \triangleright Instantiate unit capacitors based on P onto layout L
 $\text{threshold} = 0.001$, $\text{compliant} = \text{False}$, $\text{iterations} = 0$
 $\delta_{\max} = 0$, $f = [0, \dots, 0]$ \triangleright Initialize to M 0s
while $\text{compliant} == \text{False}$ **and** $\text{iterations} < \text{max_iterations}$
 $L = \text{ROUTE}(P)$ \triangleright Or re-route, from Algorithm 3
 $p = \text{EXTRACT_PARASITICS}()$ \triangleright Parasitics for each net
 \triangleright Generate new factors for parasitic compensation
 \triangleright Adjust elements of f based on impact on DNL
 $\text{PARASITIC_COMPENSATION}(N, f)$ \triangleright Algorithm 4
for $n := 0 : N - 1$ \triangleright Compare every difference
for $m := n + 1 : N$
if $|p[n] - p[m]| > \delta_{\max}$ **then** $\delta_{\max} = |p[n] - p[m]|$
 $\text{compliant} = \delta_{\max} < \text{threshold}$, $\text{iterations} ++$

indices tend to have increased values; these components correspond with the MSB capacitors. Secondly, $\max(3\sigma_{DNL^{sta}})$ also increases with larger grouped regions. This is consistent with what was expected, since a larger grouped region translates to a smaller chessboard region.

To choose the size of the grouped region, there is a trade-off between routing complexity (and hence, parasitics) and nonlinearity. Smaller grouped region sizes (i.e. 4, 6, 8) offer no significant improvement with regards to ease of routing; if the grouped region is too large, the chessboard quality is lost.

In general, a value would be defined for the maximum amount a $\max(3\sigma_{DNL^{sta}})$ for a given size is allowed to worsen that of the size 0 case. The largest X that is still in compliance would be the one that is selected. For $N = 12$, a grouped region of size 14 ($X = 8$) best balances routing complexity with an acceptable sacrifice in $\overline{DNL^{sta}}$, and hence shall be selected for further tests of routing benchmarks. This placement has a $\max(3\sigma_{DNL^{sta}})$ value of $0.955 V_{LSB}$, an acceptable increase of $0.169 V_{LSB}$ from the $0.786 V_{LSB}$ of the chessboard (size 0 grouped region). As another point of reference, the grouped placement of size 26 ($X = 10$) was also considered.

2) Δc^{par} : Based on the results of the previous step, certain sizes of the grouped region are excluded. Routing for nets other than c_{LSB} and c_X was performed using ROUTE from Algorithm 3. Routing for c_X adheres to that for c_{MSB} , and c_{LSB} to c_{group} , respectively. Parasitic extraction was then performed, and Δc^{par} used for comparison. Fig. 8 depicts these results from the routing of the selected placements.

The significant parasitics on c_{LSB} are clearly noticeable. For grouped regions of sizes 14 and 26, X is 8 and 10 respectively. It can be seen that in both cases, there is a spike in the curve at each of the corresponding C_X values compared to $C_{X\pm 1}$, creating an undesirable mismatch in parasitics. The changes in Subsection V-F were devised to rectify these problems.

In accordance with the theory, the parasitics for the chessboard placement are clearly higher than that of the hybrid placements, as shown by the results of Fig. 8. Not only do they have lower total parasitics, but the ‘‘unit parasitics’’ across each net are also already more consistent (neglecting c_{LSB}), simplifying the matching process, which is based on (37).

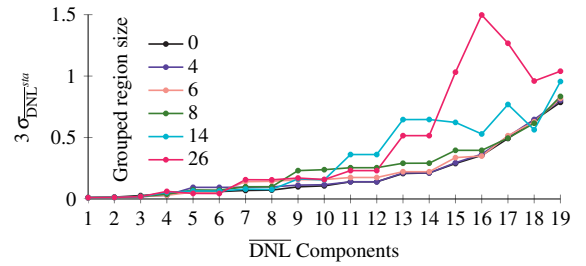


Fig. 7. Components of $3\sigma_{DNL^{sta}}$ across increasing grouped region size (indicated by the legend).

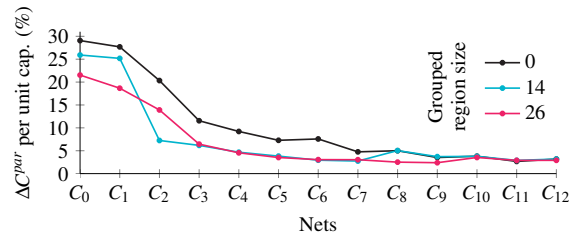


Fig. 8. Comparison between the extracted parasitics per unit capacitor of each net, after routing.

The parasitic performance of the size 26 did not offer a significant improvement over that of the size 14. For most of the nets, the parasitic values for both placements were almost identical. Regardless, any such mismatches can still be controlled by the compensation phase of the routing.

The final hybrid placement for the 12-bit architecture described by n in (12) has a size 14 grouped region, includes the placement changes to C_X and c_{LSB} , and is depicted in Fig. 9.

The unit capacitors are indexed in the right half of the figure. The left half depicts the routing styles of the different nets. To achieve matching, the MSB capacitors have two vertical trunks within one gap (as per Fig. 6b). The pairs that are closer together represent the two channels within a single gap. These pairs alternate through the gaps in order to cover all units.

The trees formed to connect the grouped capacitors, as per Algorithm 3, are also shown. Connections to the exit pins are omitted due to its simplicity: simply route along the unused vertical channel closest to the corresponding exit pin.

B. Nonlinearity Analysis and Comparison

1) *Advantages of the Hybrid Placement*: The results presented in Subsection VII-A are only valid for the grouped and hybrid placements without the changes detailed in Subsection V-F, since these results were used to establish the validity of the proposed placements by comparing them to the chessboard placement. From here, all presented results belong to the placement with the modifications from Subsection V-F (c_{LSB} moved to the matrix edge, and redistribution of C_X in rings), for both 12-bit and 10-bit hybrid placements, implicitly repeating the previous tests in order to ensure that the improvements are still retained.

First, we evaluate the worst-case $\overline{DNL^{sta}}$ of the hybrid placement compared to the chessboard (most dispersed), and the grouped (least dispersed) placements, depicted in Fig. 10. The worst-case value for the hybrid is $1.483 V_{LSB}$, which is larger than that of the chessboard ($1.151 V_{LSB}$) by $0.332 V_{LSB}$,

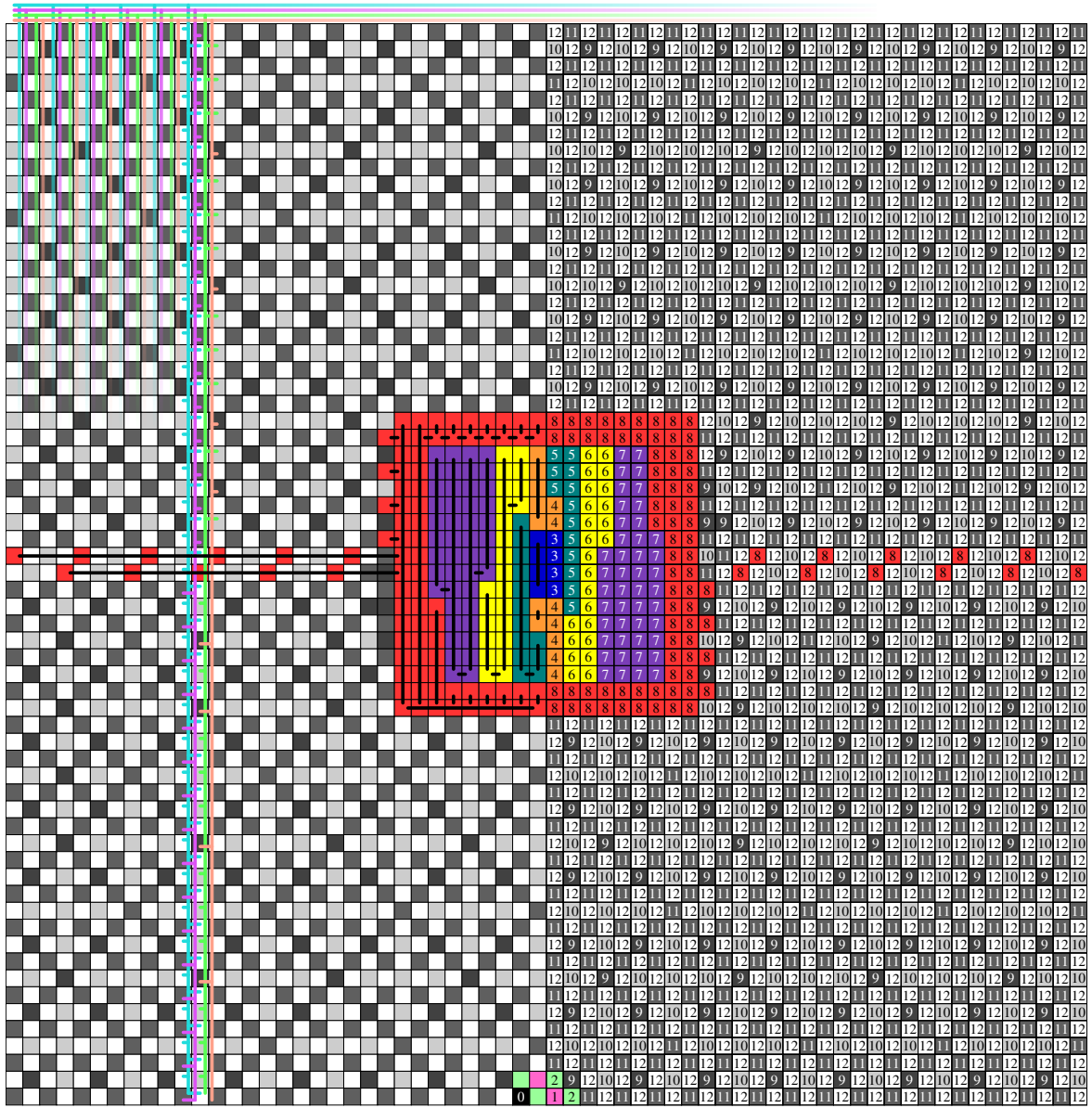


Fig. 9. Final hybrid placement with a grouped region of size 14, for the 12-bit nonbinary architecture characterized by (12), also depicting routing.

an acceptable value. The grouped placement has a significantly larger worst-case value, $5.859 V_{LSB}$. Since its placement already suffers from such a significant DNL^{sta} , its routing will not be considered, in favor of the other placements.

Using the first-order model from [1] for systematic variations, the $|\overline{DNL}^{sys}|_{\max}$ for these placements were all evaluated to be < 0.001 , which is much less than the other variation sources. Since the common-centroid placement rules were followed, the impact due to $\Delta \mathbf{c}^{sys}$ has been minimized, as explained in [22], and experimentally justified in [1].

2) *Compensation*: The \overline{DNL}^{par} is evaluated next. From Fig. 11, the results before compensation for both placements show $|\overline{DNL}^{par}|_{\max} > 15$. With compensation, the $|\overline{DNL}^{par}|_{\max}$ for the chessboard could be reduced, however it still greatly violates thresholds defined by [2] for $|DNL|$, which should be < 0.5 in order to have no missing codes. The hybrid placement provides the ability to reduce $|\overline{DNL}^{par}|_{\max}$ to a value < 0.1 .

PASTEL drives the $|\overline{DNL}^{par}|_{\max}$ to be as low as possible

without the parasitics becoming excessively large, or when an iteration limit is reached. From the nominal routing with ROUTE (before compensation), it can be seen that the maximum values of $|\overline{DNL}^{par}|$ for both placements are > 0.5 .

As the iterations progress, values typically reduce to approximately $|\overline{DNL}^{par}|_{\max} = 0.1$, whilst further iterations do not change the value significantly. These results were obtained for the linearized definitions. When applying the complete definitions, very comparable results were obtained, which justifies the use of the linear model.

Demonstrating the functionality of PASTEL, these routing results for a selection of placements are collated in Table III. Other nonbinary architectures were omitted due to the complex task of generating new capacitor ratios altogether. Due to their close mathematical relationship, $|INL^{par}|_{\max}$ also indirectly converges to be approximately 0.1. In all cases, the $|INL^{par}|$ and $|DNL^{par}|$ were greatly reduced to be well within 0.5.

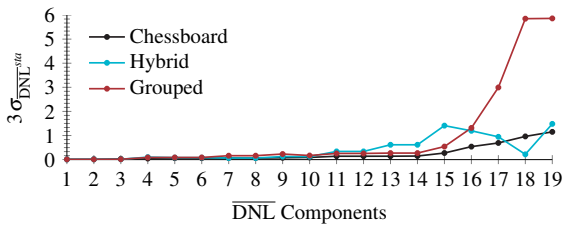


Fig. 10. Components of $3\sigma_{DNL^{sta}}$ across three placement styles of varying dispersion.

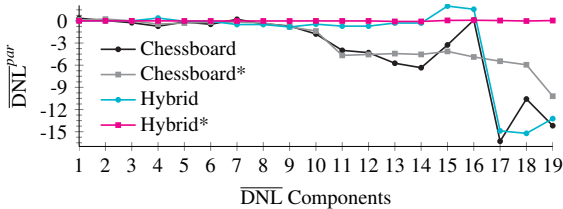


Fig. 11. Components of $\overline{DNL^{par}}$ for the chessboard and hybrid placements before, and after (*) compensation.

C. Comparison with the State-of-the-Art

The results for the 10-bit binary architecture PASTEL were used for the purposes of comparison against the results from Lin et al. in [2] and Huang et al. in [5], who presented results for up to 10- and 11-bit architectures respectively, in contrast to PASTEL, which is designed for up to 12-bit architectures.

According to the results presented in Table IV, PASTEL was superior with regards to C_u , and total area, achieving smaller (better) values compared to [2] and [5], as well as a number of older works [9], [12], [13], where the comparisons were performed in Section V of [2]. The nonlinearity metrics in Table IV are starred, as the exact definitions presented vary across different works. Post-layout parasitic extractions are always included in all sets of results, however whilst PASTEL and [2] consider the INL and DNL from the *largest* components, RMS values across all components are presented in [5], with maximum values only provided for a 9-bit architecture. Nevertheless, amongst the suitably-chosen nonlinearity metrics for comparison, PASTEL still demonstrates desirable values.

Specifically with regards to routing, the technique from [12] focused on stringently matching capacitance multiplicities, which when considering the analytical models of nonlinearity presented in this paper, implicitly results in minimized nonlinearity. However, the less significant a capacitor is, the less its impact, and hence the matching for these nets can actually be ignored to a certain extent, provided that the matching of the MSB nets is performed correctly. The routing in PASTEL is simplified such that the MSB nets are prioritized, and lengths are adjusted towards minimizing INL and DNL .

VIII. CONCLUSION

In this work, the analytical model from [1] was generalized to additionally account for nonbinary-weighted capacitor arrays. The chessboard style, also introduced in [1], was first extended to consider the nonbinary case, then improved to account for routing-based variations. A sequence of design steps led to the development of the hybrid placement style that

TABLE III
ROUTING RESULTS OF PASTEL'S HYBRID PLACEMENTS BEFORE AND AFTER COMPENSATION, USING COMPLETE DEFINITIONS OF INL AND DNL .

Placement	Compensation	$ INL^{par} _{max}$	$ DNL^{par} _{max}$
Nonbinary 12b	Before	12.536	15.226
	After	0.083	0.083
Binary 12b	Before	11.156	16.005
	After	0.105	0.103
Binary 10b	Before	4.433	5.638
	After	0.099	0.105

TABLE IV
ROUTING RESULTS OF PASTEL'S 10-BIT BINARY HYBRID PLACEMENT IN COMPARISON WITH LIN ET AL.'S [2] AND HUANG ET AL.'S [5] APPROACHES FOR 10-BIT ARCHITECTURES.

Method	C_u (fF)	Area (μm^2)	$ INL ^*$	$ DNL ^*$
PASTEL	12.913	11,278.44	0.099	0.105
Lin et al. [2]	82.000	244,877.96	0.241	0.180
Huang et al. [5]	32.700	Not given	0.370	0.430

combined the chessboard style from [1] with a new grouped style, inspired by and extended from [2].

The hybrid style was then augmented by a complementary routing technique simplified from that presented in [12]. The routing applies different techniques to connect the capacitors placed in different styles, with particular emphasis on matching routing parasitics based on the generalized analytical model. A new compensation technique is used to adjust the parasitics of each net based on the analysis of the parasitic extraction performed on the previous routing operation. The compensation repeats iteratively until a threshold is achieved, or a certain maximum number of iterations is exhausted.

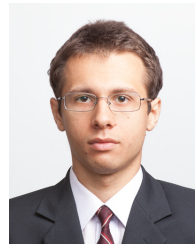
The complete design workflow ("PASTEL") that combined the new placement and routing techniques, targets the minimization of variations across all sources that were modeled in this paper, and hence minimization of the INL and DNL .

Through comparisons with other state-of-the-art works, PASTEL demonstrated superiority in several benchmark criteria, including area, total capacitance, INL and DNL .

REFERENCES

- [1] F. Burcea, H. Habal, and H. Graeb, "A New Chessboard Placement and Sizing Method for Capacitors in a Charge-Scaling DAC by Worst-Case Analysis of Nonlinearity," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 35, no. 9, pp. 1397–1410, Sep. 2016.
- [2] M. P.-H. Lin et al., "Parasitic-Aware Common-Centroid Binary-Weighted Capacitor Layout Generation Integrating Placement, Routing, and Unit Capacitor Sizing," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 36, no. 8, pp. 1274–1286, Aug. 2017.
- [3] T. Ogawa et al., "SAR ADC Algorithm with Redundancy," in *APCCAS 2008 - IEEE Asia Pacif. Conf. Circuits Syst.*, Nov. 2008, pp. 268–271.
- [4] F. Burcea, H. Habal, and H. Graeb, "Procedural Capacitor Placement in Differential Charge-Scaling Converters by Nonlinearity Analysis," in *2016 53rd ACM/EDAC/IEEE DAC*, 2016, pp. 1–6.
- [5] C. Huang, J. Chen, and C. Wey, "PACES: A Partition-Centering-Based Symmetry Placement for Binary-Weighted Unit Capacitor Arrays," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 36, no. 1, pp. 134–145, Jan. 2017.
- [6] J. Shen et al., "A 16-bit 16-MS/s SAR ADC with On-Chip Calibration in 55-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 53, no. 4, pp. 1149–1160, Apr. 2018.
- [7] W. Guo et al., "A Fully Passive Compressive Sensing SAR ADC for Low-Power Wireless Sensors," *IEEE J. Solid-State Circuits*, vol. 52, no. 8, pp. 2154–2167, Aug. 2017.
- [8] M. Zhang et al., "A 0.8–1.2 V 10–50 MS/s 13-bit Subranging Pipelined-SAR ADC Using a Temperature-Insensitive Time-Based Amplifier," *IEEE J. Solid-State Circuits*, vol. 52, no. 11, pp. 2991–3005, Nov. 2017.

- [9] M. P. Lin *et al.*, "Common-Centroid Capacitor Layout Generation Considering Device Matching and Parasitic Minimization," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 32, no. 7, pp. 991–1002, Jul. 2013.
- [10] M. P. Lin, V. W. Hsiao, and C. Lin, "Parasitic-Aware Sizing and Detailed Routing for Binary-Weighted Capacitors in Charge-Scaling DAC," in *2014 51st ACM/EDAC/IEEE DAC*, Jun. 2014, pp. 1–6.
- [11] P. Chou, M. P. Lin, and H. Graeb, "An Integrated Placement and Routing for Ratioed Capacitor Array Based On ILP Formulation," in *2016 VLSI-DAT*, Apr. 2016, pp. 1–4.
- [12] K. Ho *et al.*, "Coupling-Aware Length-Ratio-Matching Routing for Capacitor Arrays in Analog Integrated Circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 34, no. 2, pp. 161–172, Feb. 2015.
- [13] Y. Li *et al.*, "Placement for Binary-Weighted Capacitive Array in SAR ADC Using Multiple Weighting Methods," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 33, no. 9, pp. 1277–1287, Sep. 2014.
- [14] J. Chen, P. Luo, and C. Wey, "Placement Optimization for Yield Improvement of Switched-Capacitor Analog Integrated Circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 29, no. 2, pp. 313–318, Feb. 2010.
- [15] C. Lin *et al.*, "Mismatch-Aware Common-Centroid Placement for Arbitrary-Ratio Capacitor Arrays Considering Dummy Capacitors," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 31, no. 12, pp. 1789–1802, Dec. 2012.
- [16] C. F. T. Soares and A. Petraglia, "Automatic Placement to Improve Capacitance Matching Using a Generalized Common-Centroid Layout and Spatial Correlation Optimization," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 34, no. 10, pp. 1691–1695, Oct. 2015.
- [17] A. Hastings, *The Art of Analog Layout*, 2nd ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2006.
- [18] T. Chan Carusone, D. A. Johns, and K. W. Martin, *Analog Integrated Circuit Design*. Hoboken, NJ: John Wiley & Sons, 2012.
- [19] J. L. McCreary and P. R. Gray, "All-MOS Charge Redistribution Analog-to-Digital Conversion Techniques," *IEEE J. Solid-State Circuits*, vol. 10, no. 6, pp. 371–379, Dec 1975.
- [20] P. Harpe *et al.*, "A 10b/12b 40 kS/s SAR ADC With Data-Driven Noise Reduction Achieving up to 10.1b ENOB at 2.2 fJ/Conversion-Step," *IEEE J. Solid-State Circuits*, vol. 48, no. 12, pp. 3011–3018, Dec. 2013.
- [21] T. Ogawa *et al.*, "Non-Binary SAR ADC with Digital Error Correction for Low Power Applications," in *IEEE Asia Pacif. Conf. Circuits Syst.*, Dec. 2010, pp. 196–199.
- [22] C. C. McAndrew, "Layout symmetries: Quantification and application to cancel nonlinear process gradients," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 36, no. 1, pp. 1–14, Jan. 2017.



Florin Burcea received the B.Sc. degree in electronics and telecommunication engineering from University Politehnica of Bucharest (UPB) in 2013 and the M.Sc. degree in communications engineering from Technical University of Munich (TUM) in 2015. He is currently working towards PhD with the Chair of Electronic Design Automation, TUM, focusing on the design methodology and design automation of analog circuits.

In 2013, he won the first place in the Analog Integrated Circuits section of the Annual National Contest "Tudor Tănăsescu" in Romania. In 2015, 2016 and 2017 he won the first place in the Synopsys Annual International Microelectronics Olympiad of Armenia. Both contests are held in partnership with IEEE. In 2016, he received the Award of the Association for Electrical, Electronic and Information Technologies (VDE) for his Master Thesis.



Husni Habal received the B.Sc. degree in computer engineering in 2001 and the M.Sc. degree in electrical and computer engineering in 2004 from Oregon State University. He received the Dr.-Ing. Degree in Electrical Engineering from Technical University of Munich (TUM) in 2013.

Funded by the German Research Foundation, he held a post-doctorate position at the institute for Electronic Design Automation, TUM, until 2016 with a focus on the modeling of transistor aging mechanisms for numerical simulation and analog

circuit optimization.

He currently works at Infineon Technologies, AG on the design flow, implementation and verification methods of analog circuits. His focus is on analog and mixed-signal design and synthesis automation.



Helmut E. Graeb (M'02–SM'03–F'14) received the Dipl.-Ing., Dr.-Ing., and Habilitation degrees in electrical engineering from the Technical University of Munich (TUM), Munich, Germany, in 1986, 1993 and 2008, respectively.

He was with Siemens Corporation, Munich, from 1986 to 1987, where he was involved in the design of DRAMs. Since 1987, he has been with the Chair of Electronic Design Automation, TUM, where he has been the Head of a research group since 1993.

He has published around 190 papers, six of which were nominated for best papers at the Design Automation Conference (DAC), the International Conference on Computer-Aided Design (ICCAD), and the Design, Automation and Test in Europe (DATE) conference.

His current research interests include design automation for analog and mixed-signal circuits, with particular emphasis on Pareto optimization of analog circuits considering parameter tolerances, analog design for yield and reliability, hierarchical sizing of analog circuits, analog/mixed-signal test design, discrete sizing of analog circuits, structural analysis of analog and digital circuits, and analog layout.

Dr. Graeb was a recipient of the 2008 Prize of the Information Technology Society (ITG), the 2004 Best Teaching Award of the TUM EE Faculty Students Association, and the Third Prize of the 1996 Munich Business Plan Contest. He has served as the Vice-President Publications of the IEEE Council on Electronic Design Automation, an Executive Committee Member of the ICCAD, a member or the Chair of the Analog Program Subcommittees of the ICCAD, DAC, and DATE conferences, an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—PART II: ANALOG AND DIGITAL SIGNAL PROCESSING and the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, and a member of the Technical Advisory Board of MunEDA GmbH Munich, which he co-founded. He is a member of VDE ITG and VDE/VDI-Society Microelectronics, Microsystems, and Precision Engineering.



Ye X. Ding received the BE (Hons) degree in electrical engineering from the University of Queensland (UQ) in 2014, and the M.Sc. degree in communications engineering from the Technical University of Munich (TUM) in 2018.

From 2014 to 2016 he was a Digital IP Design Engineer at Nano Silicon Pty. Ltd, Brisbane, Australia. He placed second at the 2018 International Microelectronics Olympiad of Armenia (in cooperation with IEEE). In 2019, he was awarded the analog domain Cadence Master Thesis Award. He

is currently with Infineon Technologies AG, Munich, Germany, as a Circuit Design Engineer for embedded memories and e-fuses.

IEEE COPYRIGHT FORM

To ensure uniformity of treatment among all contributors, other forms may not be substituted for this form, nor may any wording of the form be changed. This form is intended for original material submitted to the IEEE and must accompany any such material in order to be published by the IEEE. Please read the form carefully and keep a copy for your files.

PASTEL: Parasitic Matching-Driven Placement and Routing of Capacitor Arrays with Generalized Ratios in Charge-Redistribution SAR-ADCs

Ding, Ye; Burcea, Florin; Habal, Husni; Graeb, Helmut

Transactions on Computer-Aided Design of Integrated Circuits and Systems

COPYRIGHT TRANSFER

The undersigned hereby assigns to The Institute of Electrical and Electronics Engineers, Incorporated (the "IEEE") all rights under copyright that may exist in and to: (a) the Work, including any revised or expanded derivative works submitted to the IEEE by the undersigned based on the Work; and (b) any associated written or multimedia components or other enhancements accompanying the Work.

GENERAL TERMS

1. The undersigned represents that he/she has the power and authority to make and execute this form.
2. The undersigned agrees to indemnify and hold harmless the IEEE from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.
3. The undersigned agrees that publication with IEEE is subject to the policies and procedures of the [IEEE PSPB Operations Manual](#).
4. In the event the above work is not accepted and published by the IEEE or is withdrawn by the author(s) before acceptance by the IEEE, the foregoing copyright transfer shall be null and void. In this case, IEEE will retain a copy of the manuscript for internal administrative/record-keeping purposes.
5. For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
6. The author hereby warrants that the Work and Presentation (collectively, the "Materials") are original and that he/she is the author of the Materials. To the extent the Materials incorporate text passages, figures, data or other material from the works of others, the author has obtained any necessary permissions. Where necessary, the author has obtained all third party permissions and consents to grant the license above and has provided copies of such permissions and consents to IEEE

BY TYPING IN YOUR FULL NAME BELOW AND CLICKING THE SUBMIT BUTTON, YOU CERTIFY THAT SUCH ACTION CONSTITUTES YOUR ELECTRONIC SIGNATURE TO THIS FORM IN ACCORDANCE WITH UNITED STATES LAW, WHICH AUTHORIZES ELECTRONIC SIGNATURE BY AUTHENTICATED REQUEST FROM A USER OVER THE INTERNET AS A VALID SUBSTITUTE FOR A WRITTEN SIGNATURE.

Florin Burcea

Signature

04-04-2019

Date (dd-mm-yyyy)

Information for Authors

AUTHOR RESPONSIBILITIES

The IEEE distributes its technical publications throughout the world and wants to ensure that the material submitted to its publications is properly available to the readership of those publications. Authors must ensure that their Work meets the

requirements as stated in section 8.2.1 of the IEEE PSPB Operations Manual, including provisions covering originality, authorship, author responsibilities and author misconduct. More information on IEEE's publishing policies may be found at http://www.ieee.org/publications_standards/publications/rights/authorrightsresponsibilities.html Authors are advised especially of IEEE PSPB Operations Manual section 8.2.1.B12: "It is the responsibility of the authors, not the IEEE, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it." Authors are also advised of IEEE PSPB Operations Manual section 8.1.1B: "Statements and opinions given in work published by the IEEE are the expression of the authors."

RETAINED RIGHTS/TERMS AND CONDITIONS

- Authors/employers retain all proprietary rights in any process, procedure, or article of manufacture described in the Work.
- Authors/employers may reproduce or authorize others to reproduce the Work, material extracted verbatim from the Work, or derivative works for the author's personal use or for company use, provided that the source and the IEEE copyright notice are indicated, the copies are not used in any way that implies IEEE endorsement of a product or service of any employer, and the copies themselves are not offered for sale.
- Although authors are permitted to re-use all or portions of the Work in other works, this does not include granting third-party requests for reprinting, republishing, or other types of re-use. The IEEE Intellectual Property Rights office must handle all such third-party requests.
- Authors whose work was performed under a grant from a government funding agency are free to fulfill any deposit mandates from that funding agency.

AUTHOR ONLINE USE

- **Personal Servers.** Authors and/or their employers shall have the right to post the accepted version of IEEE-copyrighted articles on their own personal servers or the servers of their institutions or employers without permission from IEEE, provided that the posted version includes a prominently displayed IEEE copyright notice and, when published, a full citation to the original IEEE publication, including a link to the article abstract in IEEE Xplore. Authors shall not post the final, published versions of their papers.
- **Classroom or Internal Training Use.** An author is expressly permitted to post any portion of the accepted version of his/her own IEEE-copyrighted articles on the author's personal web site or the servers of the author's institution or company in connection with the author's teaching, training, or work responsibilities, provided that the appropriate copyright, credit, and reuse notices appear prominently with the posted material. Examples of permitted uses are lecture materials, course packs, e-reserves, conference presentations, or in-house training courses.
- **Electronic Preprints.** Before submitting an article to an IEEE publication, authors frequently post their manuscripts to their own web site, their employer's site, or to another server that invites constructive comment from colleagues. Upon submission of an article to IEEE, an author is required to transfer copyright in the article to IEEE, and the author must update any previously posted version of the article with a prominently displayed IEEE copyright notice. Upon publication of an article by the IEEE, the author must replace any previously posted electronic versions of the article with either (1) the full citation to the IEEE work with a Digital Object Identifier (DOI) or link to the article abstract in IEEE Xplore, or (2) the accepted version only (not the IEEE-published version), including the IEEE copyright notice and full citation, with a link to the final, published article in IEEE Xplore.

Questions about the submission of the form or manuscript must be sent to the publication's editor.

Please direct all questions about IEEE copyright policy to:

IEEE Intellectual Property Rights Office, copyrights@ieee.org, +1-732-562-3966

5 Conclusion & Outlook

This thesis presented, for three application areas of electronic circuits and systems, the problem-specific computational modeling with a problem-specific trade-off between the computational cost for simulation and the accuracy of description of the behaviour to be modeled. The purpose of the modeling problem was to improve the efficiency of the design process. Each of the three applications had its own modeling requirements and each referred to a different stage in the design process.

In the first application, a new multi-harmonic model for the simulation of DC-DC converters was presented. For a Buck and for a Boost converter, both implemented at transistor level, the proposed method achieved a simulation speedup of more than one order of magnitude, with a loss of accuracy below 3%.

In the second application, a new and complete approach for the robustness optimization of integrated circuits with MEMS was presented. The optimization simultaneously considered design and process parameters from both the mechanical and the electrical side. By using a reduced-order model for the MEMS part, an optimization speedup of up to 4 times was achieved, with an accuracy difference below 1% compared to the original compact model.

In the third application, a simple and efficient analytical model of integral and differential non-linearity for charge-redistribution ADCs with generalized capacitor ratios was presented. The model was used to conduct a layout strategy with combined placement and routing with the goal of minimizing the static non-linearity of the ADC.

Bibliography

- [1] G. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8), 1965.
- [2] J. L. Hennessy and D. A. Patterson. A new golden age for computer architecture. *Communications of the ACM*, 62(2):48–60, 2019.
- [3] G. Gielen, E. Maricau, and P. De Wit. Analog circuit reliability insub-32 nanometer cmos: Analysis and mitigation. In *Proceedings of the 2011 Design, Automation & Test in Europe Conference & Exhibition*, pages 1–6, 2011.
- [4] S. Nassif, N. Mehta, and Y. Cao. A resilience roadmap. In *Proceedings of the 2010 Design, Automation & Test in Europe Conference & Exhibition*, pages 1011–1016, 2010.
- [5] M. G. R. Degrauwe, O. Nys, E. Dijkstra, J. Rijmenants, S. Bitz, B. L. A. G. Goffart, E. A. Vittoz, S. Cserveny, C. Meixenberger, G. van der Stappen, and H. J. Oguey. Idac: an interactive design tool for analog cmos circuits. *IEEE Journal of Solid-State Circuits*, 22(6):1106–1116, 1987.
- [6] R. Harjani, R. A. Rutenbar, and L. R. Carley. Oasys: a framework for analog circuit synthesis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 8(12):1247–1266, 1989.
- [7] H. Y. Koh, C. H. Sequin, and P. R. Gray. Opasyn: a compiler for cmos operational amplifiers. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 9(2):113–125, 1990.
- [8] J. P. Harvey, M. I. Elmasry, and B. Leung. Staic: an interactive framework for synthesizing cmos and bicmos analog circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11(11):1402–1417, 1992.
- [9] C. Basso. *Switch-Mode Power Supplies*. McGraw-Hill Education, 2014.
- [10] R. D. Middlebrook and S. Čuk. A general unified approach to modelling switching-converter power stages. *International Journal of Electronics Theoretical and Experimental*, 42(6):521–550, 1977.
- [11] A. Davoudi, J. Jatskevich, and T. De Rybel. Numerical state-space average-value modeling of pwm dc-dc converters operating in dcm and ccm. *IEEE Trans. on Power Electronics*, 21(4):1003–1012, 2006.

Bibliography

- [12] J. Sun, D. M. Mitchell, M. F. Greuel, P. T. Krein, and R. M. Bass. Averaged modeling of pwm converters operating in discontinuous conduction mode. *IEEE Trans. on Power Electronics*, 16(4):482–492, 2001.
- [13] V. Vorperian. Simplified analysis of PWM converters using model of PWM switch part I: Continuous conduction mode. *IEEE Trans. on Aerospace and Electronic Systems*, 26(3):490–496, 1990.
- [14] V. Vorpérián. Simplified analysis of pwm converters using model of pwm switch. ii. discontinuous conduction mode. *IEEE Trans. on Aerospace and Electronic Systems*, 26(3):497–505, 1990.
- [15] Y. Amran, F. Huliehel, and S. Ben-Yaakov. A unified SPICE compatible average model of PWM converters. *IEEE Trans. on Power Electronics*, pages 585–694, 1991.
- [16] D. Tannir, Y. Wang, and P. Li. Accurate modeling of nonideal low-power pwm dc-dc converters operating in ccm and dcm using enhanced circuit averaging techniques. *ACM Trans. on Design Automation of Electronic Systems*, 21(4):1–15, 2016.
- [17] S. R. Sanders, J. M. Noworolski, X. Z. Liu, and G. C. Verghese. Generalized averaging method for power conversion circuits. *IEEE Trans. on Power Electronics*, 6(2):251–259, 1991.
- [18] V. A. Caliskan, O. Verghese, and A. M. Stankovic. Multifrequency averaging of dc/dc converters. *IEEE Transactions on Power Electronics*, 14(1):124–133, 1999.
- [19] D. Maksimović, A. M. Stanković, V. J. Thottuvelil, and G. C. Verghese. Modeling and simulation of power electronic converters. *Proceedings of the IEEE*, 89(6):898–912, 2001.
- [20] J. M. Noworolski and S. R. Sanders. Generalized in-plane circuit averaging. In *Applied Power Electronics Conference and Exposition, 1991. APEC'91. Conference Proceedings, 1991., Sixth Annual*, pages 445–451. IEEE, 1991.
- [21] Y. Wang, D. Gao, D. Tannir, and P. Li. Multi-harmonic nonlinear modeling of low-power pwm dc-dc converters operating in ccm and dcm. In *Proceedings of the 2016 Design, Automation & Test in Europe Conference & Exhibition*. EDA Consortium, 2016.
- [22] Y. Wang, D. Gao, D. Tannir, N. Dong, P. Fang, W. Dong, and P. Li. Multiharmonic small-signal modeling of low-power pwm dc-dc converters. *ACM Trans. on Design Automation of Electronic Systems*, 22(4):1–16, 2017.
- [23] F. Misoc, M. M. Morcos, and J. Lookadoo. Fourier-series models of dc–dc converters. In *Proceedings of the North American Power Symposium*, pages 193–199, 2006.

- [24] R. Trincherro, P. Manfredi, I. S. Stievano, and F. G. Canavero. Steady-state analysis of switching converters via frequency-domain circuit equivalents. *IEEE Trans. on Circuits and Systems— II*, 63(8):748–752, 2016.
- [25] E. Setiawan, T. Hirata, and I. Hodaka. Accurate symbolic steady state modeling of buck converter. *International Journal of Electrical and Computer Engineering*, 7(5):2374–2381, 2017.
- [26] M. Forouzesh, Y. Siwakoti, S. A. Gorji, F. Blaabjerg, and B. Lehman. Step-up dc–dc converters: A comprehensive review of voltage-boosting techniques, topologies, and applications. *IEEE Trans. on Power Electronics*, 32(12):9143–9177, 2017.
- [27] R. Trincherro, I. S. Stievano, and F. G. Canavero. Steady-state analysis of switching power converters via augmented time-invariant equivalents. *IEEE Trans. on Power Electronics*, 29(11):5657–5661, 2014.
- [28] J. W. van der Woude, W. L. de Koning, and Y. Fuad. On the periodic behavior of pwm dc-dc converters. *IEEE Trans. on Power Electronics*, 17(4):585–595, July 2002.
- [29] Y. Yan, F. C. Lee, and P. Mattavelli. Unified three-terminal switch model for current mode controls. *IEEE Trans. on Power Electronics*, 27(9):4060–4070, Sep. 2012. doi:10.1109/TPEL.2012.2188841.
- [30] S. Tian, F. C. Lee, J. Li, Q. Li, and P. Liu. A three-terminal switch model of constant on-time current mode with external ramp compensation. *IEEE Trans. on Power Electronics*, 31(10):7311–7319, Oct 2016. doi:10.1109/TPEL.2015.2508037.
- [31] Simetrix Technologies Ltd. *SIMetrix/SIMPLIS – User’s Manual*, 2015.
- [32] Coventor. *MEMS+ Tutorials*, 6.0 edition.
- [33] A. Cheng. Design of a readout scheme of a mems microphone. Master’s thesis, Delft University of Technology, 2009.
- [34] M. Pak, F. V. Fernandez, and G. Dundar. A novel design methodology for the mixed-domain optimization of a mems accelerometer. *Integration, the VLSI Journal*, 62:314–321, Jun 2018.
- [35] M. Pak, F. V. Fernandez, and G. Dundar. Optimization of a mems accelerometer using a multiobjective evolutionary algorithm. In *2017 14th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, June 2017. doi:10.1109/SMACD.2017.7981574.
- [36] T. Konishi et al. A single-platform simulation and design technique for cmos-mems based on a circuit simulator with hardware description language. *J. of Microelectromechanical Syst.*, 22(3):755–767, June 2013. doi:10.1109/JMEMS.2013.2243111.

Bibliography

- [37] H. Toshiyoshi et al. A mixed-design technique for integrated mems using a circuit simulator with hdl. In *Proc. of the 20th Intern. Conf. Mixed Des. of Integr. Circuits and Syst. - MIXDES 2013*, pages 17–22, June 2013.
- [38] A. Parent, A. Krust, G. Lorenz, and T. Piirainen. A novel model order reduction approach for generating efficient nonlinear verilog-a models of mems gyroscopes. In *2015 IEEE International Symposium on Inertial Sensors and Systems (ISISS) Proceedings*, pages 1–4, March 2015. doi:10.1109/ISISS.2015.7102377.
- [39] B. Vernay, A. Krust, G. Schröpfer, F. Pêcheux, and M. Louërat. Systemc-ams simulation of a biaxial accelerometer based on mems model order reduction. In *2015 Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS (DTIP)*, pages 1–6, April 2015. doi:10.1109/DTIP.2015.7160987.
- [40] T. Bechtold. *Model Order Reduction of Electro-Thermal MEMS*. PhD thesis, Albert-Ludwigs University of Freiburg, 2005.
- [41] J. Klaus, R. Paris, and R. Sommer. Systematic mems asic design flow using the example of an acceleration sensor. In *2016 13th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, pages 1–4, 2016.
- [42] M. G. R. Degrauwe, O. Nys, E. Dijkstra, J. Rijmenants, S. Bitz, B. L. A. G. Goffart, E. A. Vittoz, S. Cserveny, C. Meixenberger, G. van der Stappen, and H. J. Oguey. Idac: an interactive design tool for analog cmos circuits. *IEEE Journal of Solid-State Circuits*, 22(6):1106–1116, 1987.
- [43] F. El-Turky and E. E. Perry. Blades: an artificial intelligence approach to analog circuit design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 8(6):680–692, 1989.
- [44] R. Harjani, R. A. Rutenbar, and L. R. Carley. Oasys: a framework for analog circuit synthesis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 8(12):1247–1266, 1989.
- [45] H. Y. Koh, C. H. Sequin, and P. R. Gray. Opasyn: a compiler for cmos operational amplifiers. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 9(2):113–125, 1990.
- [46] A. Hastings. *The Art of Analog Layout*. Pearson Prentice Hall, Upper Saddle River, NJ, 2nd ed. edition, 2006.
- [47] J. L. McCreary and P. R. Gray. All-MOS Charge Redistribution Analog-to-Digital Conversion Techniques. *IEEE J. Solid-State Circuits*, 10(6):371–379, Dec 1975. doi:10.1109/JSSC.1975.1050629.
- [48] F. Burcea. *Chessboard Placement for Capacitors in Charge-Scaling Converters Based on Nonlinearity Analysis*. Master thesis, Technische Universität München, 2015.

- [49] F. Burcea, H. Habal, and H. Graeb. A New Chessboard Placement and Sizing Method for Capacitors in a Charge-Scaling DAC by Worst-Case Analysis of Nonlinearity. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 35(9):1397–1410, September 2016. doi:10.1109/TCAD.2015.2511146.
- [50] F. Burcea, H. Habal, and H. Graeb. Procedural Capacitor Placement in Differential Charge-Scaling Converters by Nonlinearity Analysis. In *2016 53rd ACM/EDAC/IEEE DAC*, pages 1–6, 2016. doi:10.1145/2897937.2898073.
- [51] C. Huang, J. Chen, and C. Wey. PACES: A Partition-Centering-Based Symmetry Placement for Binary-Weighted Unit Capacitor Arrays. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 36(1):134–145, January 2017. doi:10.1109/TCAD.2016.2561403.
- [52] M. P. Lin et al. Common-Centroid Capacitor Layout Generation Considering Device Matching and Parasitic Minimization. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 32(7):991–1002, July 2013. doi:10.1109/TCAD.2012.2226457.
- [53] M. P. Lin, V. W. Hsiao, and C. Lin. Parasitic-Aware Sizing and Detailed Routing for Binary-Weighted Capacitors in Charge-Scaling DAC. In *2014 51st ACM/EDAC/IEEE DAC*, pages 1–6, June 2014. doi:10.1109/DAC.2014.6881492.
- [54] P. Chou, M. P. Lin, and H. Graeb. An Integrated Placement and Routing for Ratioed Capacitor Array Based On ILP Formulation. In *2016 VLSI-DAT*, pages 1–4, April 2016. doi:10.1109/VLSI-DAT.2016.7482535.
- [55] K. Ho et al. Coupling-Aware Length-Ratio-Matching Routing for Capacitor Arrays in Analog Integrated Circuits. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 34(2):161–172, February 2015. doi:10.1109/TCAD.2014.2379656.
- [56] Y. Li et al. Placement for Binary-Weighted Capacitive Array in SAR ADC Using Multiple Weighting Methods. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 33(9):1277–1287, September 2014. doi:10.1109/TCAD.2014.2323217.
- [57] J. Chen, P. Luo, and C. Wey. Placement Optimization for Yield Improvement of Switched-Capacitor Analog Integrated Circuits. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 29(2):313–318, February 2010. doi:10.1109/TCAD.2009.2035587.
- [58] C. Lin et al. Mismatch-Aware Common-Centroid Placement for Arbitrary-Ratio Capacitor Arrays Considering Dummy Capacitors. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 31(12):1789–1802, December 2012. doi:10.1109/TCAD.2012.2204993.
- [59] C. F. T. Soares and A. Petraglia. Automatic Placement to Improve Capacitance Matching Using a Generalized Common-Centroid Layout and Spatial Correlation Optimization. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 34(10):1691–1695, October 2015. doi:10.1109/TCAD.2015.2419624.

Bibliography

- [60] M. J. M. Pelgrom et al. Matching Properties of MOS Transistors. *IEEE J. Solid-State Circuits*, 24(5):1433–1440, October 1989.
- [61] C. C. McAndrew. Layout symmetries: Quantification and application to cancel nonlinear process gradients. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 36(1):1–14, January 2017. doi:10.1109/TCAD.2016.2561970.
- [62] M. P.-H. Lin et al. Parasitic-Aware Common-Centroid Binary-Weighted Capacitor Layout Generation Integrating Placement, Routing, and Unit Capacitor Sizing. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 36(8):1274–1286, August 2017. doi:10.1109/TCAD.2017.2685598.