

Chapter 2

Previous Works on Automatic Analog IC Sizing

2.1 Analog IC Sizing Automation: An Historical Perspective

Historically, automatic analog IC sizing has been implemented using two different approaches, namely knowledge-based and optimization-based approaches [1], illustrated in Fig. 2.1. Knowledge-based approaches systematize the design using a design plan derived from expert knowledge. In these early strategies, a plan is built with design equations and a design strategy to produce the device sizes that meet the performance requirements. In [2], the designer expertise is captured in a design plan where all design equations are explicitly solved during the definition of the plan. Once the topology is selected, the plan is executed for the given specifications and produces a first design. In [3] the same overall strategy is used, but defines the circuits hierarchically, with a design plan for each sub-block. It also adds backtracking with design-reuse methodologies to recover from failed designs. A slightly different approach is found in [4–7], where designer’s knowledge captured in expert systems using artificial intelligence techniques. The main advantage of knowledge-based approaches is the short execution time. However, deriving the plan is hard and time-consuming, it requires constant maintenance to keep it up to date with technological evolution, and also, the results are not optimal, suitable only as a first-cut-design.

Aiming for optimality, the next generations of sizing tools apply optimization techniques to analog IC sizing. Optimization-based sizing tools come in many flavors, but they can be generally classified into two main subclasses: equation-based and simulation-based circuit optimization, from the bulk methods used to evaluate circuit’s performance. Equation-based methods use analytic design equations, whereas simulation-based use a circuit simulator to relate the circuit’s performance figures with the design variables. While the first take advantage of fast execution time, the development and maintenance of the equations are costly, in addition the lack of accuracy in the evaluation of the circuits’ performance greatly

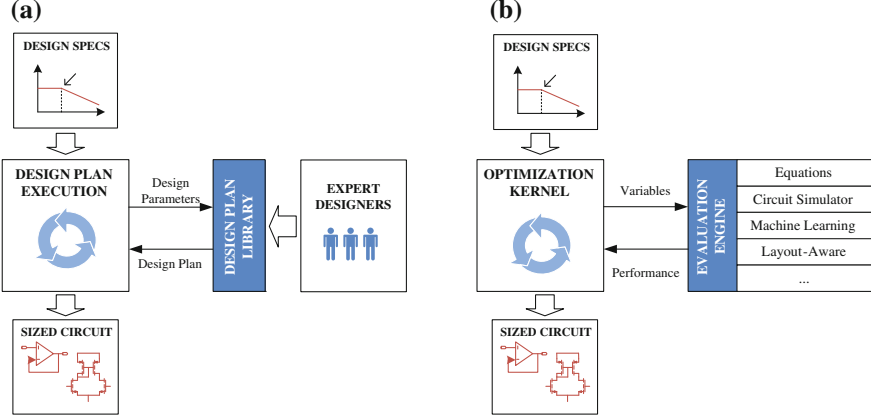


Fig. 2.1 Automatic circuit sizing approaches. **a** Knowledge-based. **b** Optimization-based

penalizes this approach. To tackle the accuracy issue, simulation-based evaluation takes advantage of the circuit simulator, e.g., SPICE [8], to provide accurate performance figures. The cost of this approach is obviously execution time, but with the computation resources available today, quasi-optimal solutions can be obtained with acceptable execution timeframes [9].

In the current state-of-the-art, optimization-based is the most common approach to automatic sizing at circuit level, and some flavors of it are already integrated and available in the toolkits provided by most EDA providers, like the circuit optimizer feature of Cadence’s Virtuoso Custom Design Platform GXL [10], Synopsys Titan ADX [11] or MunEDA-GNO [12]. Nevertheless, for system level, design plans or knowledge-based tool are still around [9]. The following sections will overview the most preminent methods in analog integrated circuit sizing and the latest developments in the area.

2.2 Optimization-Based Circuit Sizing

The critical points in optimization-based approaches are the optimization techniques employed and the methods used to evaluate the circuit’s performance. Sect. 2.2.1 overviews the first and Sect. 2.2.2 the second.

2.2.1 Optimization Techniques Applied to Analog Circuit Sizing

Different optimization techniques are used for analog IC sizing and optimization, both deterministic and stochastic. In equation-based methods, as the analytic

equations that relate the circuit's performance figures to the design variables are known, is possible to use classical optimization methods. By using the circuit simulator, methods that take advantage of some properties of the models, e.g., quadric or geometric programming cannot be used, leading, to stochastic heuristic optimization techniques.

Early approaches to simulation-based automatic sizing used local optimization around a "good" solution, where simulated annealing (SA) [13] was the most common optimization technique used. SA mimics the annealing of material under slow cooling to minimize the internal energy, as the name suggests. In DELIGHT. SPICE the optimization algorithm (phase I-II-III method of feasible directions) is used to perform local optimization around a user provided starting point. Kuo-Hsuan et al. [14] and FASY [15, 16] use equations to derive an approximate solution, and then use SA to optimize the design. Likewise, Cheng et al. [17] also uses SA with a good starting point, but considers the transistor bias conditions to constrain the problem. FRIDGE [18] aims for general applicability approach by using an annealing-like optimization without any restriction to the starting point, whereas Castro-Lopez et al. [19] use SA followed by a deterministic method for fine-tuning to perform the optimization.

Other widely used class of optimization methods are genetic algorithms (GAs) [20]. The principle of the GA is to mimic natural evolution where new solution vectors are obtained from the current population by the application of the genetic operators of mutation and crossover. Barros et al. [21, 22] presents a circuit sizing optimization supported by a GA where the evaluations of the populations was made using a both the circuit simulator and a support vector machine (SVM). Alpaydin et al. [23] use hybridization of evolutionary and annealing optimization, where the circuits' performance figures are computed using a blend of equations and simulations. Given the affinity evolutionary algorithms have with parallel implementations, in Santos-Tavares [24], MAELSTROM [25] and ANACONDA [26] the time to simulate the population reduced by a parallel mechanism that shares the evaluation load among multiple computers.

A different approach to circuit sizing optimization that also employs evolutionary methods is to simultaneously generate the circuit topologies (the arrangement of the devices) and the device sizes. Koza [27], Sripramong [28], and more recently Hongying [29] proposed a design methodology able to create new topologies by exploring the immense possibilities starting from low abstraction level. Small elementary blocks are connected bottom-up to each other to form a new topology. Various fundamental entities can be applied, such as, single transistors, elementary building blocks or node connections. However, these approaches are met with great skepticism, as designers are suspicious of the generated structures, because they often differ too much from the well-known analog circuit structures.

Swarm intelligence algorithms [30] can also be found in the literature applied to analog circuit sizing. The fundament of swarm intelligence algorithms is to use many simple agents that lead an intelligent global behavior, like the one observed in many insect hives. From these methods, the most common are the ant colony

optimization (ACO), which was successfully applied in [31, 32], and particle swarm optimization (PSO) that can be found in [33–35].

Circuit sizing is in its essence a multi-objective multi constraint problem, and the designer often explores the tradeoff among contradictory performance measures, for example, minimizing power consumption while maximizing bandwidth, or maximizing gain and minimizing area of an amplifier, therefore, the usage of multi-objective optimization techniques is becoming more common. When considering multiple objectives the output is not one solution, but a set of optimal tradeoff solutions, usually referred as Pareto optimal front (POF). Given the multiple elements already present in both evolutionary and swarm intelligence algorithms, these are the natural candidates to implement such approach. In MOJITO [36] the evolutionary multi-objective methods are applied, respectively, to circuit sizing and both sizing and topology exploration, whereas in [35] the particle swarm optimization is explored in both single and multi-objective approaches. A different approach is taken by Pradhan and Vemuri in [37], where the multi-objective simulated annealing (MOSA) is used.

Instead of executing sizing on-the-fly, in some approaches the non-dominated solutions are generated using the previously mentioned multi-objective optimization methods or variations of them for the most relevant tradeoff (prior to the design task) and then, the suitable solution is selected from the already sized solutions [38–41].

2.2.2 *Circuit's Performance Evaluation*

As stated, in terms of evaluation methods, the two main subclasses are equation-based and simulation-based evaluation. As both have advantages and disadvantages, hybrid solutions considering both approaches are also common.

2.2.2.1 **Equation-Based**

In the equation-based approaches different optimization techniques are used. The optimization in OPASYN [42] is done using steepest descent, whereas in STAIC [43], a successive solution refinements technique is used. OPTIMAN [44] uses SA applied to analytical models created automatically by ISAAC [45]. DONALD [46] is an interactive design space exploration tool that assists the designer during circuit sizing by automatic analytical manipulations of the circuit equations. Maulik et al. [47, 48] define the sizing problem as a constrained nonlinear optimization problem using spice models and DC operating point constraints, solving it using sequential quadratic programming. Matsukawa et al. [49] design $\Delta\Sigma$ and pipeline analog to digital converters solving via convex optimization the equations that relate the performance of the converter to the size of the components. In ASTRX/OBLX [50]

a SA optimization is performed on a cost function defined by equations for dc operation point, and small signal asymptotic waveform evaluation (AWE). This evaluation technique is also used in DARWIN [51].

In GPCAD [52] a posynomial circuit model is optimized using geometrical programming (GP), the execution time is in the order of few seconds, but the general application of posynomial models is difficult and the time to derive the model for new circuits is still high. To reduce the long time dispended in model development several automatic techniques were proposed, in [53, 54] Gielen et al. provide a good overview on symbolic analysis applied to analog ICs. However, some design characteristics are still not easy to extract automatically in terms of analytical expressions with sufficient accuracy. Kuo-Hsuan et al. [14] revisited the posynomial modeling recently, surpassing the accuracy issue by introducing an additional step, where local optimization using SA and a circuit simulator is performed. The same strategy is applied in FASY [15, 16] where analytical expressions are solved to generate an initial solution and a simulation-based optimization is performed to fine tune the solution.

The equation-based methods' strong point is the short evaluation time, making them, like the knowledge-based approaches, extremely suited to derive first-cut designs. The main drawback is that, despite the advances in symbolic analysis, not all design characteristics can be easily captured by analytic equations, making the generalization of the method to different circuits very difficult. In addition, the approximations introduced in the equations yield low accuracy designs especially for complex circuits, requiring additional work to ensure that the circuit really meets the specifications.

2.2.2.2 Simulation-Based

With the availability of computing resources, simulation-based optimization gained ground, being the most common approach found in recent approaches. In DELIGHT.SPICE [55] the optimization algorithm (phase I-II-III method of feasible directions) is used to perform local design optimization around a user provided starting point.

FRIDGE [18] on the other hand aims for global optimality by using an annealing-like optimization without any restriction to the starting point. However, to restrict the dimensionality of the problem the user still must provide the range for the optimization variables. In MAELSTROM [25] and ANACONDA [26] the evaluation time is reduced by a parallel mechanism that shares the evaluation load among multiple computer. Given the affinity evolutionary algorithms have with parallel implementations, it was the base technique chosen in MAELSTROM, however and because the success that SA has demonstrated in many implementations, the authors' option was to use parallel re-combinative simulated annealing (PRSA). In ANACONDA the approach is similar, but instead of the PRSA it is

applied a variation of pattern search algorithms, named by the authors as stochastic pattern search.

Generality and easy-and-accurate model (the circuit netlist) are the strong points of simulation-based techniques. However, the execution time is large for complex circuits (~ 100 variables) and prohibitive at system level, and, without the proper constraints the algorithm may not converge to a good result. Some heuristic schemes exist to automate the process of defining the constraints [56]. However, automatic constraint defining mechanisms are not integrated in sizing tools and their application is somewhat circuit class specific. Being the high execution time the weaker point of this approach, some techniques had been proposed to cope with it. As stated before, Kuo-Hsuan et al. [14] and FASY [15, 16] use equation-based techniques to derive an approximate solution, and then use simulation within a SA optimization kernel to optimize the design. Cheng et al. [17] use the transistor bias conditions to constrain the problem and instead of solving the circuit by finding transistor sizes, the problem is solved by finding the bias of the transistors. The transistor sizes are then derived from the bias point using electric simulation. In MAELSTROM [25] and ANACONDA [26] the evaluation time is reduced by a parallel mechanism that share the evaluation load among multiple computers.

To handle complex circuit where the simulation is too expensive, [57] takes a hierarchical approach where equations are used for system level evaluation, while the circuit simulator is used for block level. The sizing flow is neither top-down or bottom-up, but concurrent optimization of both system and block level.

In order to increase the performance of simulator-based circuit optimizers, surrogate-based evaluation is maturing as an alternative technique to avoid the costly simulations [58, 59]. In such approaches, the global accuracy of the simulator is traded by the speed of applying the surrogate model for performance estimation. These models are automatically generated which eases design and maintenance, however tuning the model accuracy can be complicated, which makes them hard to use in a generalized form.

Alpaydin et al. [23] use a neural-fuzzy model combined with an evolutionary optimization strategy where some of the AC performance metrics are computed using an equation-based approach. De Bernardinis et al. [60] use a learning tool based in SVM to represent the performance space of analog circuits. The performance space is modeled using the knowledge acquired from a training set via circuit simulation. Wolfe et al. [61] present a performance macro-model based in a neural network. This model once constructed, is used to replace the SPICE simulation during the sizing of analog circuits, increasing the efficiency of the performance parameter estimates' computation. The training and validation data sets are constructed with discrete points, sampled over the design space. The work explores several sampling methodologies to adaptively improve model quality and applies a sizing rules methodology in order to reduce the design space and ensure the correct operation of analog circuits. Barros et al. [21, 22] present a cell-level sizing and optimization approach based on SVMs and evolutionary strategies. The SVM is used to dynamically model performance space and identify the feasible design space regions while at the same time the evolutionary techniques are looking for the

global optimum. The evaluation is still done with HSPICE to ensure accuracy, but the number of evaluation is reduced by using the SVM to prune the candidate solutions, in [62] the approach is generalized for multi-objective.

2.2.3 *Commercial Solutions*

The EDA industry closely follows the main trends in academia and R&D workgroups, focusing on the lower level of abstraction levels dealing with device sizing and layout description levels. Synopsys titan ADX (originally Magna's) takes an equation-based approach. To overcome the issues with equations, namely the non-generality and the difficulty to derive them, an automated procedure is used to extract the circuit models without user inputs. To ensure accuracy of the models, the automatic procedure uses a circuit simulator. While this automatic procedure solve the problems related to time to design and maintain the models, model accuracy is still an issue, limiting the applicability of the approach to high-performance designs. In the high-end of Cadence products (Virtuoso-GXL), automatic IC optimization is provided with 4 alternative optimization methods that can be used with the simulator. In this approach accuracy is ensured, but the limitative single-objective optimization and the hard-to-use constraint handling together with the long simulation time for real industrial designs, are the main deterrent for generalized use.

2.3 Robust Circuit Optimization

With the evolution of the fabrication processes, new challenges to analog IC design automation arisen, namely the need for circuits that are robust to the variability in both fabrication process and environment. The variability in IC design comes from the environment (voltage, temperature) variations, die-to-die variations that affect all devices identically, and intra-die variations that affect each device differently.

Process variation has been around since the early design processes, but the reiterated reduction in devices' sizes increases the effects of process variation because: transistors may be shrinking, but atoms do not. On large devices some displaced atoms do not cause a significant performance shift, but for small devices the performance shift must be accounted if the vast majority of the design are to meet specifications after fabrication [63].

The environment parameters are usually defined by a lower and upper bound, while the process variation parameters are random variables defined by some probability distribution function (commonly a Gaussian with zero mean). Environment and die-to-die variations are commonly handled using process, voltage and temperature (PVT) corners, while intra-die variations are handled using Monte Carlo (MC) analysis. However, both comprehensive PVT corners and MC

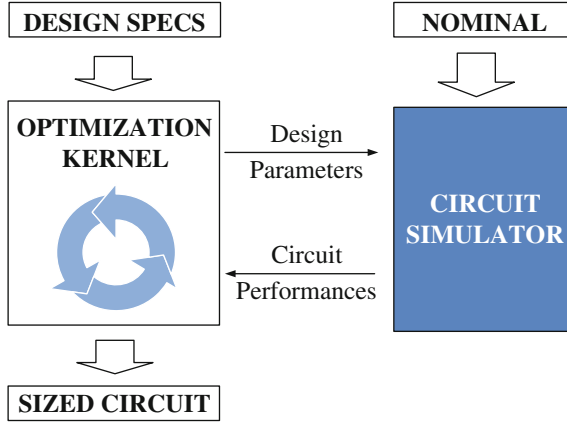


Fig. 2.2 Nominal simulation-based circuit sizing and optimization

require many additional circuit simulations and are inadequate to be used inside the optimization loop.

To circumvent the amount of simulations required, worst case parameters sets or worst case corners are often considered instead [64, 65]. Worst case corners are set of values for the environment and process parameters, which are not controlled by the designer, that lead to the worst values of circuit's performances. However, unlike digital cells where worst case corners do not depend on topology or device sizes and are provided in the models by the foundry (e.g. Slow, Fast, etc.), analog worst case sets do depend of the circuit's topology and sizes of the devices [66]. Figures 2.2 and 2.3 illustrate the differences between nominal and variation-aware circuit sizing and optimization.

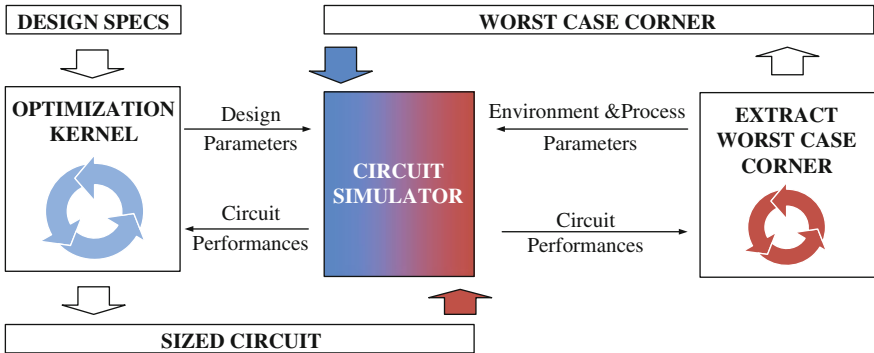


Fig. 2.3 Variation-aware automatic circuit sizing and optimization

2.3.1 Worst-Case Optimization

A simple method to derive these worst case corners, which does not account for intra-die variations, is for the designer to select the relevant worst case corners [21]. A more systematic approach is the “Fast PVT corner” described in [63] that transforms the search for the worst case PVT corner in an explicit optimization problem that minimizes the circuit performance in the PVT corner space. This worst case corner is then considered during the circuit design. As the worst case corner depends of the circuit sizing, regular updates may be needed.

Another approach is to find the worst case parameters sets accounting for intra-die variations. One of such method is the “Sigma driven corners” [63]. The algorithm to extract the parameters that correspond to a parametric yield of $n \times \sigma$ for a given performance target is outlined in Algorithm 2.1. First a MC sampling is done (~ 100 points for 3σ), then the p.d.f. of the output is estimated using kernel density estimates. The percentile of the p.d.f. is then used to derive the $n \times \sigma$ performance boundary. The last step is to find the process and environment parameters that cause that performance values.

Algorithm 2.1 Sigma driven corner extraction [63]

1.	Simulate 100 MC samples
2.	Estimate p.d.f. of the circuit’s performance
3.	Use the p.d.f. to determine performance bound
4.	Find the corresponding process and environment parameters Starting from the closest point in the sampling

Linear Worst Case distance design [67], assumes that in circuits already designed for nominal conditions, i.e., typical values for temperature and voltage and no process variation, most of the circuits’ performance figures can be accurately linearized. It takes a deterministic circuit optimization approach, where the worst case environment and process parameters are iteratively updated using a gradient estimation. Then, at each iteration, the design variables are determined solving the sizing optimization problem considering the latest worst case environment and process parameter set. To increase the accuracy of the estimates, in [68] a quadratic model replaces the linear model. Using the worst case distances concepts, in [69] the bias-driven performance equations are derived to include variability for flexible thin film transistors. A linear-solver is then used to optimize the bias point and look up tables, which were obtained before the sizing task using the circuit simulator to account for all transistor’s non-idealities, are used to obtain devices’ sizes from their bias.

The above solutions provide a way to handle variability, but they work only over the yield considering each to the circuit’s performances separately, and are not adequate for handling the statistical yield (% of fabricated ICs that meet all specifications). On the other side, they provide a direct relation between the variability of each performance with respect to the process and environment variability, that is

extremely useful for the designers to understand which performances figures are critical under variability.

A different approach is to handle yield considering full MC for the relevant solutions and speed the process by avoid useless simulations of less promising solutions. In [70], multiple levels of evaluation accuracy are considered simultaneously. For the initial evaluations only DC simulations are considered, in the next levels other simulations are considered for nominal conditions, followed by PVT corners, and finally, full MC.

This scheme allows global optimization where tentative designs are initially evaluated using simplified (and faster) simulations, and as those tentative solutions get closer to the solution more accurate evaluations are performed. To further accelerate the optimization, a surface response model is generated on the fly, and is used to generate new tentative solutions. A similar approach to efficiently use the computational resources without compromising accuracy is taken in [71], where evolutionary ordinal optimization is used. In ordinal optimization the exact values for yield are not necessary, as long as tentative solutions are well sorted in terms of performance. In this manner, the number of simulations executed to estimate the yield is allocated efficiently to avoid unnecessary yield estimations of bad tentative solutions and, accurately estimate the yield of the best candidates as the execution progresses.

Despite the merit of the previously discussed approaches in handling variability, they are still impractical for high-sigma design. High-sigma design take the amount of simulations required to another level, a 6σ design (approx. 2 error per billion samples) requires billions of MC simulations for verification [63]. Specific methods for high-sigma design have been proposed that reduce the amount of simulations to practical levels [63, 72–74]. Yet, while extremely relevant for bit-cell and other kind of highly redundant blocks that appear thousands of times in an IC, the relevance of high-sigma design is reduced for analog IC design, as most analog cells are not repeated hundreds of times in an IC. Eventually for some stricter applications, 4-sigma design (1 error per 15 thousand samples [63]) might make sense.

In addition to environment and fabrication variability, effects like well proximity effects (WPE), shallow trench isolation (STI) stress effect, electromigration, hot carrier injection (HCI), negative bias temperature instability (NBTI), positive bias temperature instability (PBTI), among others, lead to aging effects that are noticeable in the latest fabrication processes, where the behavior of the devices and consequently the performance of the circuit degrades over time. With respect to this issue, methods to simulate considering aging have been proposed, e.g., in [75] MC circuit simulations (plus a surface model) are used to derive the failure distribution through the lifetime of the circuit, and, in [76], quadratic worst case distances are extended to account both for variation and aging effects.

2.3.2 Commercial Solutions

The growth of the importance of variation effects, created a gap between what traditional tools supported and what the analog designer needed. Taking that opportunity, new commercial EDA solutions emerged to efficiently tackle the variation-aware design. For example, fast PVT corners, sigma-driven corners and high-sigma via sampling ordering are provided as commercial solutions by Solido Design Solutions [77], whereas, MunEDA [11] delivers worst case distances using linearized and quadratic models, and importance sampling, among other solutions.

Much has been achieved in this domain, still, there is a space for innovation in variation-aware circuit sizing, first and foremost, variation-aware design still require an agonizingly high number of circuit simulations to yield accurate results.

2.4 Layout-Aware Sizing

Another source of analog IC performance degradation are the parasitic devices formed in the silicon implementation of the circuit [78]. Hence, the performance of a circuit needs to be guaranteed post-layout in the presence of these layout parasitics, which prevent the circuit from realizing the estimated performance.

In the traditional flow, the layout generation task is only triggered when the sizing task is complete. However, to achieve post-layout successful designs that meet all specifications, time-consuming and non-systematic iterations between these electrical and physical design phases are required. Without them, performance overdesign results in wasted power and area, and if underestimated, the circuits' post-layout performance can be compromised [79]. Furthermore, even the evaluation of the circuit area or aspect ratio is practically impossible from the netlist alone, which is especially critical for designs with geometric constraints, e.g. an upper limit in the layout width, causing expensive redesigns. Thus, to address post-layout performance degradation and geometric requirements earlier in the design flow, the, so called, layout-aware or layout-driven design approaches include layout effects during the sizing loop, as illustrated in Fig. 2.4. Table 2.1 summarizes some state-of-the-art layout-aware sizing tools and their different ways of extracting layout-parasitics, that are presented next.

Without actually generating the layout in real-time, in [37] device parasitic effects are modeled by linear regression from a Pareto optimal surface, obtained by sampling the design space and using a procedural generator to produce the layout for each point, where each solution is aware of its specific layout induced effect. In [80] the layout is also produced by a parameterized generator, where layout parasitics and devices sizes are passed to the precompiled symbolic performance model that estimates the circuit performance for each sizing solution, attempting to avoid circuit simulation. In [81] the parasitic effects are modeled in the circuit equations, a

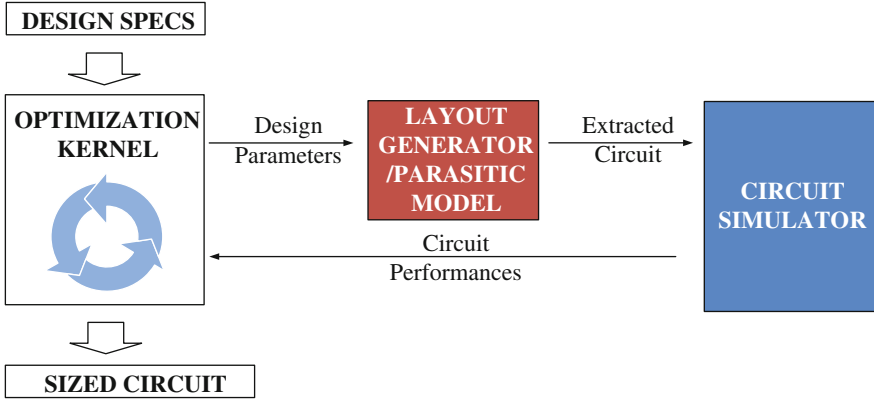


Fig. 2.4 Layout-aware automatic circuit sizing and optimization

template based layout generator is used to derive the parasitic equations that are solve together with the performance equations using non-linear-programming.

In [83] the procedural generator is used during the execution of a design plan, where part of the parasitic effects are estimated using analytical models, and other part is accounted by setting the BSIM's [90] parameters accordingly. Unlike previous approaches, in [82] the procedural generator is used to generate the layout but inside the optimization loop. Likewise in [19, 85] both parasitic-aware and geometrically constrained sizing are handled. Since the predefined layout template is supported by a slicing tree and the block placement is fixed, area and shape optimization are obtained by finding the number of fingers of MOS transistors that yield desired geometric features. In addition, layout parasitic effects are extracted and used during the circuit's evaluation in the optimization loop. This extraction can be done either using analytical equations and layout sampling or using complete 3D geometric extraction models. In [86], a layout description script is also used in-loop in a simulation-based scheme, where the evolutionary strategies algorithm and simulated annealing are used to perform layout-aware optimization of one candidate from the Pareto front.

Finally, in [84] the designer knowledge is embedded in a set of placement and routing matching, symmetry and proximity constraints, as well as geometric and electrical performance constraints. Then, a deterministic nonlinear optimization algorithm is used to determine the device sizes. In each iteration, the floorplan is exhaustively explored by enumeration [91], and then the best placement, selected based on some performance criteria, is routed and considered for post-layout simulation.

Layout-aware/Layout-inclusive sizing tools target a design process that avoids time consuming iterations, by bringing layout-related data into the sizing process. However, both complete automatic layout generation and accurate parasitic extraction are still time consuming and hard to setup operations [79]. The time cost appears either in the processing time, when using custom automatic layout

Table 2.1 Overview of layout-aware sizing tools

Work	Circuit synthesizer	Performance evaluation	Layout generator		Layout data included	
			Placer	Router	Geom.	Parasitics
Vancorenland et al. [82]	Optimization-based	Fitted functions			Eq.	ID/2D critical nets only
Ranjan et al. [79]	Optimization-based	Symbolic models			No	Area interconnect extractor
Pradhan [37]	MOO	Symbolic models			No	Bulk, device and interconnect
Youssef et al. [83]	Design plan to iteratively solve model equations	BSIM4 equations			Yes	Models of devices' stress effects
Habal et al. [84]	Nonlinear deterministic optimizer	Circuit simulator	Enum. of all floorplans	Cadence chip assembly router®	Yes	Commercial extractor
Lopez [19, 85]	MOO	Circuit simulator	Coded slicing-tree	Template-based	Yes	Analytical and commercial extractor
Liao et al. [81]	Nonlinear programming	Bias driven equations	Template based/assisted placement	Channel router	Yes	ID/2D modeling of the critical nets
Berkol et al. [86]	Evolutionary strategies and SA	Circuit simulator	Layout description script		Yes	Commercial extractor
AIDA [87–89]	MOO	Simulator (Spectre®, Eldo® or HSPICE®)	Multiple B*-trees	Automatic electromigration-aware WT and global routing in-loop	Yes	2D/2.5D modeling of the devices and routing

generator plus layout extraction inside the sizing optimization loop [84], or in the design time of circuit specific template-based [19, 85] or procedural generators [37, 79, 82, 83] with parasitic estimation.

2.5 Summary of the Automatic Circuit Sizing Approaches

In this survey, the most significant automatic analog IC sizing tools were presented to provide a better understanding of its advantages and shortcomings. The variety of existing tools and techniques covering several aspects of analog design are summarized in Table 2.2, highlighting the benefits and pitfalls of the main methods to automate analog IC sizing, and the sizing tools surveyed are summarized in Table 2.3.

The following paragraphs frame the most relevant aspects of the simulation-based multi-objective optimization approach developed in this work, faced with the techniques found in the surveyed state-of-the-art methodologies.

Table 2.2 Summary of advantages and shortcomings of the techniques applied to circuit sizing tools

Knowledge-based	Optimization-based		
	Equations	Simulator	Models
IDAC [2] OASYS [3] BLADES [4] CAMP [5] ISAID [6, 7]	OPASYN [42] STAIC [43] Kuo-Hsuan [14] OPTIMAN [44] DONALD [46] ASTRX/OBLX [50] DARWIN [51] GPCAD [52] Doboli [92] Matsukawa [49]	Kuo-Hsuan [14] FASY [15, 16] ASTRX/OBLX [50] DARWIN [51] DELIGHT.SPICE [55] Cheng [17] FRIDGE [18] MAELSTROM [25] ANACONDA [26] Castro-Lopez [19]	Alpaydin [23] De Bernardinis [60] Wolfe [61] Barros [21, 22]
(+) Fast execution time (+) Expert knowledge	(+) Fast execution time (+) Use of expert knowledge (+) Automatic symbolic analysis	(+) Models are easy to develop (++) Accurate and flexible	(+) can be accurate (+) can be flexible
(-) Expert knowledge is difficult to capture (-) Not optimal	(-) Difficult derivation of some equations (-) Simplifications lead to lack of accuracy	(-) Long execution time (-) Limited to cell-level	(-) Accuracy is hard to predict

Table 2.3 Summary of circuit sizing tools

Tool/author	Circuits	Design plan/optm.	Evaluation	1	2	3	Time setup/exec.
IDAC [2]	Analog cells	Design plan plus SA post-optimization	Equations	⊗	✗	✓ ⁶	months/few sec
DELIGHT.SPICE [55]	Analog cells	Feasible directions optimization	Circuit simulator	✓	✗	✗	moderate/18 h
OASYS [3]	Amplifier	Design plan (includes backtracking features)	Equations	✓	✓ ⁴	✗	6 months/3 s
BLADES [4]	Amplifier	Expert system for analog design	Equations	⊗	✓ ⁴	✗	Long/20 min
OPASYN [42]	Amplifier	Steepest descent	Equations	✓	✓ ⁴	✓ ⁶	2 weeks/5 min
CAMP [5]	Amplifier	Expert system, flexible architecture	Circuit simulator	⊗	✓ ⁵	✓ ⁶	⊗/⊗
OPTIMAN [45]	Amplifier	SA	Analytical models ISAAC [72]	✗	✗	✗	⊗/1 min
SEAS [100]	Amplifier	SA	Equations	✗	✓ ⁵	✗	⊗/⊗
DONALD [46]	Amplifier	Equation solver (Newton Raphson variant)	Equations	⊗	✗	✗	⊗/⊗
Chang [101]	ADC	Top-Down constraint driven	Behavior models	⊗	✓ ⁵	✓ ⁶	⊗/4-89 h
STAIC [43]	Amplifier	2 step optimization	Equations	⊗	✗	✓ ⁶	long/2 min
MINLP [102]	Amplifier	Branch and bound	Equations and BSIM models	⊗	✓ ⁵	✗	6 months/1 min
Maulik et al. [47, 48]	Amplifier	Sequential quadratic programming	Equations and BSIM models	⊗	✗	✗	6 months/1 min
FRIDGE [18]	Amplifier	SA	Circuit simulator	✗	✗	✗	1 h/45 min
DARWIN [51]	Amplifier	GA	Small signal, Analytical expressions	✗	✓ ⁵	✗	⊗/⊗
ISAID [6, 7]	Amplifier	Qualitative reasoning + post optimization	Equations and qualitative reasoning	⊗	✗	✗	⊗/⊗

(continued)

Table 2.3 (continued)

Tool/author	Circuits	Design plan/optm.	Evaluation	1	2	3	Time setup/exec.
SD-OPT [103]	$\Sigma\Delta$ -modulator	SA	Equations and behavioral simulation	✗	✗	✗	Long/1, 5 week
FASY [15, 16]	Amplifier	SA + Gradient	Circuit simulator	✗	✓ ⁴	✗	⊘/6 h
ASTRX/OBLX [50]	Analog Cells	SA	AWE Equations	✗	✗	✗	few days/few sec
Koza [27]	Analog Cells	GA	Circuit simulator	✗	✓ ⁵	✗	⊘/⊘
GPCAD [52]	Amplifier	Geometric programming	Posynomial models	✗	✗	✗	⊘/fast
MAELSTROM [25]	Amplifier	GA + SA	Circuit simulator	✗	✗	✗	⊘/3,6 h
ANACONDA [26]	Amplifier	Stochastic pattern search	Circuit simulator	✗	✗	✗	⊘/10 h
Sripamong [28]	Amplifier	GA	Circuit simulator	✗	✓ ⁵	✗	⊘/3 days
Alpaydin [23]	Amplifier	Evolutionary strategies + SA	Fuzzy + NN trained with Circuit simulator	✓	✗	✗	⊘/45 min
Shou-Jin [104]	Passive filters	GA	Equations	✗	✓ ⁵	✗	⊘/⊘
Barros [21, 22]	Analog cells	GA	Circuit simulator and feasibility SVM models	✓	✗	✗	⊘/20 min
Castro-Lopez [19]	Amplifier	SA + Powell's method	Circuit simulator	⊘	✗	✓	⊘/25 min
Santos-Tavares [24]	Amplifier	Parallel GA	Circuit simulator	⊘	✗	✗	⊘/⊘
MOJITO [36]	Amplifier	NSGA-II	Circuit simulator	✓	✓ ⁵	✗	⊘/<7 days
Pradhan [37]	Amplifier, Filter	Multi-objective SA	Layout aware MNA models	✗	✗	✗	⊘/16 min
Matsukawa [49]	ADC	Convex Optimization	Convex functions	✓	✓ ⁵	✗	⊘/⊘
Cheng [17]	Amplifier	SA	Equations	✗	✗	✓	⊘/<1 h
Hongying [29]	Amplifier	GA with VDE	Circuit simulator	✗	✓ ⁵	✗	⊘/⊘
Fakhfakh [35]	LNA current conveyor	Multi-Objective PSO	Equations	✗	✗	✗	⊘/⊘

(continued)

Table 2.3 (continued)

Tool/author	Circuits	Design plan/optm.	Evaluation	1	2	3	Time setup/exec.
Kuo-Hsuan [14]	RFDA	Convex optimization Fine Tuning	Posynomial models Circuit simulator	✗	✗	✗	⊗/1 h
Habal [84]	Tunable OTA	Deterministic non-linear optimizer	Circuit simulator	✗	✗	✓	⊗/2–11 h
Kamisetty [33]	Oscillator	PSO	Equations	✗	✗	✗	⊗/⊗
Benhala [32]	Current Conveyors	ACO	Equations	✗	✗	✗	⊗/<1 min
Roca [40]	Amplifiers	NSGA-II	Circuit simulator	✗	✗	✗	⊗/1–2 h
Gupta and Gosh [31]	Amplifiers	ACO	Circuit simulator	✗	✗	✗	⊗/1–3 h
Kumar and Duraishwamy [34]	Amplifier	PSO	Circuit simulator	✗	✗	✗	⊗/⊗
AIDA-C [62, 93, 94]	OPAMPs	NSGA-II	Circuit simulator and SVM	✓	✗	✗	1 h/several hours
AIDA-GM [96–98]	LNA OPAMPs	NSGA-II with gradient models	Circuit simulator	✓	✗	✗	1 h/several hours
Liao et al. [81]	OPAMP	Nonlinear programming	Bias driven equations	✓	✗	✓	⊗/<1 s
Afacan et al. [57]	Active Filter	Evolutionary strategies	Equations and Simulator	✗	✗	✗	⊗/<few minutes
Chen et al. [69]	OPAMPs	Convex Solver	Bias driven equations with WCD	✓	✓	✗	⊗/<1 s
AIDA [87, 95, 98]	Oscillators OPAMPs	NSGA-II; MOPSO; MOSA	Circuit simulator	✓	✗	✓	1 h/several days

✓—Supported; ✗—Not supported; ⊗—Undefiend
/ Robust design; 2 Topology; 3 Includes layout effects
^aTopology is handled before sizing; ^bTopology is handled during sizing; ^cLayout is handled after sizing

Circuit Evaluation In the tool that implements the proposed techniques, AIDA, the evaluation of the circuits is done using the circuit simulator. In equation-based systems the usage of classical optimization methods is possible, however, the accuracy of the models and the derivation of such equations strong limits its applicability. Besides the accuracy ensured in the evaluation, the generality of the approach is increased and the setup time required to introduce new circuits is decreased, as the circuit simulator is part of the traditional design flow anyway. The price for these advantages is paid with a significantly increase in the evaluation time, and also, the relation between the performance figures and the design variables becomes unknown, making the usage of classic optimization methods inappropriate.

Optimization Algorithms AIDA uses a multi-objective multi-constraint optimization based on evolutionary techniques. From the surveyed simulation-based approaches the most common stochastic algorithms were based on simulation annealing and evolutionary approaches, with some of the latest implementations considering swarm intelligence methods (PSO and ACO). Besides, analog IC design is characterized by highly constrained problems and many tradeoffs. In this context, special relevance is given to multi-objective algorithms.

2.5.1 Contributions

The contributions that were described in Chap. 1 of this book are here revisited in light of the techniques used in state-of-the-art in analog IC sizing automation.

First, in [93], a simple but effective method to model the analog IC sizing optimization in multi-objective multi-constraint optimization using NSGA-II is proposed. The model accounts for variability by considering worst case corners. In the implemented tool, the worst case corners are selected by the designer. The main advantage of this approach to handle variability is its simplicity. Besides, the usage of corner sets is a common signoff practice in the industry. However, by delegating to the designer the selection of the relevant corners, a place for human error is added to the design flow. To automatically derive the worst case corners advanced techniques like Fast PVT, sigma driven corners or WCD could be introduced in the implementation. The relevance of true multi-objective analog IC sizing and optimization is further explored in [94] where a VCO is optimized for both minimum Phase-Noise and Power consumption, instead of the traditional optimization of the figure-of-merit (FOM) that relate both objectives. In this work the obtained POE contains more than 40 different solutions with performance that beat current state-of-the-art oscillators and delivers the optimal tradeoffs to the designer.

As seen, the main drawback of using the circuit simulator is the increased evaluation time, hence an effective usage of the simulations is mandatory. To generate better tentative solutions, in [95–97] the Gradient Model, which is automatically extracted using machine learning techniques, is used to add circuit specific knowledge to the optimization. The Gradient Model, is embedded in the

implementation of the genetic operators of the optimization kernel, increasing the efficiency of the exploration of the search space.

While many algorithms have been used to approach analog IC sizing, a fair comparison between approaches was not presented in the state-of-the-art. Historically, both SA and GA have been used intensively, in this sense is natural to consider both evolutionary and annealing approaches. Given the recent experiments with swarm intelligence in this domain, and to broad the scope of the implemented framework, this class of methods were also considered. From a brief perusal of the multi-objectives implementations, the ideas like non-sorted domination or crowding distance (further described in Chap. 4 of this book) presented in NSGA-II are reused by several other methods. In terms of the swarm intelligence algorithms, both ACO and PSO have been applied to circuit sizing. Because of MOPSO is already found in the literature and the un-natural application to real valued problems of the path finding ideas of the ACO, MOPSO was considered. In [98] these stochastic approaches were experimented in for the sizing of oscillators.

In [87–89] and in Chap. 6 of this book is presented a methodology for automatic layout-aware sizing of analog ICs that uses a built-in extractor to accurately compute the impact of parasitic from both floorplan and early-stages of routing. The computational efficiency is improved by avoiding the time-consuming detailed layout generation needed by off-the-shelf tools for parasitic extraction [99]. Moreover, prior knowledge of the circuit's parasitics is not required, i.e., there is no need for circuit specific equations or models as in [79]. The applicability of the approach is demonstrated with post-layout performances attained for the circuits designed with the developed layout-aware methodology, versus, the traditional optimization-based sizing.

2.6 Conclusions

Although much has been accomplished in automatic design of analog circuits, the fact is that automatic custom generators usable in industrial design environment are just starting to gain ground. As seen, the evolution of the technology keeps creating new challenges/opportunities for analog EDA researchers. In this survey, the most significant analog IC sizing automation tools were presented and analyzed to provide a better understanding of its advantages and shortcomings. The different approaches were classified in terms of the techniques used and the most significant aspects observed were the setup and the execution time, as well as the accuracy in the evaluation of the solutions. The trends in optimization methods were also surveyed, showing a predominance of the multi-objective approaches.

The presented study was used to help and frame the definition of the techniques and strategies used in the development of AIDA, which is described in the next chapters.

References

1. Barros MFM, Guilherme JMC, Horta NCG (2010) Analog circuits and systems optimization based on evolutionary computation techniques. Springer, Berlin
2. Degrauwe MGR et al (1987) IDAC: an interactive design tool for analog CMOS circuits. *IEEE J Solid-State Circuits* 22(6):1106–1116
3. Harjani R, Rutenbar RA, Carley LR (1989) OASYS: a framework for analog circuit synthesis. *IEEE Trans Comput Aided Des Integr Circuits Syst* 8(12):1247–1266
4. El-Turky F, Perry EE (1989) BLADES: an artificial intelligence approach to analog circuit design. *IEEE Trans Comput Aided Des Integr Circuits Syst* 8(6):680–692
5. Sheu BJ, Lee JC (1990) Flexible architecture approach to knowledge-based analogue IC design. *IEE Proc G—Circuits Dev Syst* 137(4):266–274
6. Makris CA, Toumazou C (1995) Analog IC design automation. II. Automated circuit correction by qualitative reasoning. *IEEE Trans Computer-Aided Design Integr Circuits Syst* 14(2):239–254
7. Toumazou C, Makris CA (1995) Analog IC design automation. I. Automated circuit generation: new concepts and methods. *IEEE Trans Comput Aided Des Integr Circuits Syst* 14(2):218–238
8. Nagel LW (1975) SPICE2: a computer program to simulate semiconductor circuits. Thesis, EECS Department, University of California, Berkeley
9. Gielen GGE, Rutenbar RA (2000) Computer-aided design of analog and mixed-signal integrated circuits. *Proc IEEE* 88(12):1825–1852
10. Cadence (2014) Cadence. <http://www.cadence.com/>. Accessed 17 Oct 2014
11. Synopsys (2014) Synopsys. <http://www.synopsys.com/>. Accessed 17 Oct 2014
12. MunEDA (2015) MunEDA. <http://www.muneda.com/>. Accessed 12 Apr 2015
13. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. *Science* 220(4598):671–680
14. Kuo-Hsuan M, Po-Cheng P, Hung-Ming C (2011) Integrated hierarchical synthesis of analog/RF circuits with accurate performance mapping. In: International symposium on quality electronic design, Santa Clara, 2011
15. Torralba A, Chavez J, Franquelo LG (1996) FASY: a fuzzy-logic based tool for analog synthesis. *IEEE Trans Comput Aided Des Integr Circuits Syst* 15(7):705–715
16. Torralba AJ, Chavez J, Franquelo LG (1996) Fuzzy-logic-based analog design tools. *IEEE Micro* 16(4):60–68
17. Cheng-Wu L, Pin-Dai S, Ya-Ting S, Soon-Jyh C (2009) A bias-driven approach for automated design of operational amplifiers. In: International symposium on VLSI design, automation and test, Hsinchu, 2009
18. Medeiro F, Fernandez FV, Dominguez-Castro R, Rodriguez-Vazquez A (1994) A statistical optimization-based approach for automated sizing of analog cells. In *IEEE/ACM international conference on computer-aided design (ICCAD)*, 1994
19. Castro-Lopez R, Guerra O, Roca E, Fernandez FV (2008) An integrated layout-synthesis approach for analog ICs. *IEEE Trans Comput Aided Des Integr Circuits Syst* 27(7):1179–1189
20. Goldberg David E (1994) Genetic and evolutionary algorithms come of age. *Commun ACM* 37(3):113–119
21. Barros M, Guilherme J, Horta N (2010) Analog circuits optimization based on evolutionary computation techniques. *Integr VLSI J* 43(1):136–155
22. Barros M, Guilherme J, Horta N (2007) GA-SVM feasibility model and optimization kernel applied to analog IC design automation. In: *ACM Great Lakes symposium on VLSI*, Stresa-Lago Maggiore, 2007
23. Alpaydin G, Balkir S, Dunder G (2003) An evolutionary approach to automatic synthesis of high-performance analog integrated circuits. *IEEE Trans Evol Comput* 7(3):240–252

24. Santos-Tavares R, Paulino N, Higino J, Goes J, Oliveira JP (2008) Optimization of multi-stage amplifiers in deep-submicron CMOS using a distributed/parallel genetic algorithm. In: International symposium on circuits and systems, Seattle, 2008
25. Krasnicki M, Phelps R, Rutenbar RA, Carley LR (1999) MAELSTROM: efficient simulation-based synthesis for custom analog cells. In: Design automation conference, New Orleans, 1999
26. Phelps R, Krasnicki M, Rutenbar RA, Carley LR, Hellums JR (2000) Anaconda: simulation-based synthesis of analog circuits via stochastic pattern search. *IEEE Trans Comput Aided Des Integr Circuits Syst* 19(6):703–717
27. Koza JR, III Bennett FH, Andre D, Keane MA, Dunlap F (1997) Automated synthesis of analog electrical circuits by means of genetic programming. *IEEE Trans Evol Comput* 1 (2):109–128
28. Sripramong T, Toumazou C (2002) The invention of CMOS amplifiers using genetic programming and current-flow analysis. *IEEE Trans Comput Aided Des Integr Circuits Syst* 21(11):1237–1252
29. Hongying Y, Jingsong H (2010) Evolutionary design of operational amplifier using variable-length differential evolution algorithm. In: International conference on computer application and system modeling, Taiyuan Shanxi, 2010
30. Chu S-C, Huang H-C, Roddick JF, Pan J-S (2011) Overview of algorithms for swarm intelligence. In: Jedrzejowicz P, Nguyen NT, Hoang K (eds) Computational collective intelligence. Technologies and applications, Third International Conference, ICCCI 2011, Gdynia, Poland, September 21–23, 2011, Proceedings, Part I. Lecture Notes in Computer Science, vol 6922. Springer, Heidelberg, pp 28–41
31. Gupta H, Ghosh B (2012) Analog circuits design using ant colony optimization. *Int J Electron Comput Commun Technol* 2(3):9–21
32. Benhala B, Ahaitouf A, Fakhfakh M, Mechaqrane A (2012) New adaptation of the ACO algorithm for the analog circuits design optimization. *Int J Comput Sci Issues* 9(3):360–367
33. Kamisetty S, Garg J, Tripathi JN, Mukherjee J (2011) Optimization of analog RF circuit parameters using randomness in particle swarm optimization. In: World Congress on information and communication technologies, Mumbai, 11–14 Dec 2011
34. Kumar PP, Duraiswamy K (2012) An optimized device sizing of analog circuits using particle swarm optimization. *J Comput Sci* 8(6):930–935
35. Fakhfakh Mourad, Cooren Yann, Sallem Amin, Loulou Mourad, Siarry Patrick (2010) Analog circuit design optimization through the particle swarm optimization technique. *Analog Integr Circ Sig Process* 63(1):71–82
36. McConaghy T, Palmers P, Steyaert M, Gielen G (2011) Trustworthy genetic programming-based synthesis of analog circuit topologies using hierarchical domain-specific building blocks. *IEEE Trans Evol Comput* 15(4):557–570
37. Pradhan A, Vemuri R (2009) Efficient synthesis of a uniformly spread layout aware pareto surface for analog circuits. In: International conference on VLSI Design, New Delhi, 2009
38. Deniz E, Dundar G (2010) Hierarchical performance estimation of analog blocks using Pareto Fronts. In: 2010 Conference on Ph.D. research in microelectronics and electronics (PRIME), Berlin, 18–21 July 2010
39. Castro-Lopez R, Roca E, Fernandez FV (2009) Multimode Pareto fronts for design of reconfigurable analogue circuits. *Electron Lett* 45(2):95–96
40. Roca Elisenda, Velasco-Jiménez Manuel, Castro-López Rafael, Fernández Francisco V (2012) Context-dependent transformation of pareto-optimal performance fronts of operational amplifiers. *Analog Integr Circ Sig Process* 73(1):65–76
41. Gielen G, McConaghy T, Eeckelaert T (2005) Performance space modeling for hierarchical synthesis of analog integrated circuits. In: 42nd design automation conference, 13–17 June 2005
42. Koh HY, Sequin CH, Gray PR (1990) OPASYN: a compiler for CMOS operational amplifiers. *IEEE Trans Comput Aided Des Integr Circuits Syst* 9(2):113–125

43. Harvey JP, Elmasry MI, Leung B (1992) STAIC: an interactive framework for synthesizing CMOS and BiCMOS analog circuits. *IEEE Trans Comput Aided Des Integr Circuits Syst* 11 (11):1402–1417
44. Gielen GGE, Walscharts HCC, Sansen WMC (1990) Analog circuit design optimization based on symbolic simulation and simulated annealing. *IEEE J Solid-State Circuits* 25 (3):707–713
45. Gielen GGE, Walscharts HCC, Sansen WMC (1990) ISAAC: a symbolic simulator for analog integrated circuits. *IEEE J Solid-State Circuits* 25(3):707–713
46. Swings K, Sansen W (1991) DONALD: a workbench for interactive design space exploration and sizing of analog circuits. In: *Proceedings of the European conference on design automation (EDAC)*, Amsterdam, 25–28 Feb 1991
47. Maulik PC, Carley LR, Allstot DJ (1993) Sizing of cell-level analog circuits using constrained optimization techniques. *IEEE J Solid-State Circuits* 28(3):223–241
48. Maulik PC, Carley LR, Rutenbar RA (1995) Integer programming based topology selection of cell-level analog circuits. *IEEE Trans Comput Aided Des Integr Circuits Syst* 14 (4):401–412
49. Matsukawa K et al (2009) Design methods for pipeline and delta-sigma A-to-D converters with convex optimization. In: *2009 Asia and South Pacific design automation conference*, Yokohama, 19–22 Jan 2009
50. Ochotta ES, Rutenbar RA, Carley LR (1996) Synthesis of high-performance analog circuits in ASTRX/OBLX. *IEEE Trans Comput Aided Des Integr Circuits Syst* 15(3):273–294
51. Kruiskamp W, Leenaerts D (1995) DARWIN: CMOS opamp synthesis by means of a genetic algorithm. In: *Proceedings of the 32nd annual ACM/IEEE design automation conference DAC'95*, San Francisco, CA, 1995
52. del Mar Hershenson M, Boyd SP, Lee TH (1998) GPCAD: a tool for CMOS op-amp synthesis. In: *IEEE/ACM international conference on computer-aided design*, San Jose, 1998
53. Gielen G, Wambacq P, Sansen WM (1994) Symbolic analysis methods and applications for analog circuits: a tutorial overview. *Proc IEEE* 82(2):680–692
54. Daems W, Gielen G, Sansen W (2003) Simulation-based generation of posynomial performance models for the sizing of analog integrated circuits. *IEEE Trans Comput Aided Des Integr Circuits Syst* 22(5):517–534
55. Nye W, Riley DC, Sangiovanni-Vincentelli A, Tits AL (1988) DELIGHT.SPICE: an optimization-based system for the design of integrated circuits. *IEEE Trans Comput Aided Des Integr Circuits Syst* 7(4):501–519
56. Massier T, Graeb H, Schlichtmann U (2008) The sizing rules method for CMOS and bipolar analog integrated circuit synthesis. *IEEE Trans Comput Aided Des Integr Circuits Syst* 27 (12):2209–2222
57. Afacan E, Ay S, Fernandez FV, Dundar G, Bascakaya F (2014) Model based hierarchical optimization strategies for analog design automation. In: *2014 Design, automation and test in Europe conference and exhibition (DATE)*, Dresden, 24–28 March 2014
58. Yelten MB, Ting Zhu, Koziel S, Franzon PD, Steer MB (2012) Demystifying surrogate modeling for circuits and systems. *IEEE Circuits Syst Mag* 12(1):45–63
59. Liu B, Zhang Q, Gielen G (2013) A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems. *IEEE Trans Evol Comput* 18 (9):180–192
60. De Bernardinis F, Jordan MI, Sangiovanni Vincentelli A (2004) Support vector machines for analog circuit performance representation. In: *Proceedings of the 40th annual Design Automation Conference (DAC'03)*, Anaheim CA, 2–6 June 2003
61. Wolfe GA (2004) Performance macro-modeling techniques for fast analog circuit synthesis. Dissertation, University of Cincinnati
62. Lourenço N, Martins R, Barros M, Horta N (2013) Analog circuit design based on robust POFs using an enhanced MOEA with SVM models. In: Fakhfakh M, Tlelo-Cuautle E, Castro-Lopez R (eds) *Analog/RF and mixed-signal circuit systematic design*. Springer. *Lecture Notes in Electrical Engineering*, vol 233. Springer, Heidelberg, pp 149–167

63. McConaghy T, Breen K, Dyck J, Gupta A (2013) Variation-aware design of custom integrated circuits: a hands-on field guide. Springer-Verlag, New York
64. Debyser G, Gielen G (1998) Efficient analog circuit synthesis with simultaneous yield and robustness optimization. In: 1998 IEEE/ACM international conference on computer-aided design (ICCAD 98), San Jose, CA, 8–12 Nov 1998
65. Dharchoudhury A, Kang SM (1995) Worst-case analysis and optimization of VLSI circuit performances. IEEE Trans Comput Aided Des Integr Circuits Syst 14(4):481–492
66. Antreich KJ, Graeb HE, Wieser CU (1994) Circuit analysis and optimization driven by worst-case distances. IEEE Trans Comput Aided Des Integr Circuits Syst 13(1):57–71
67. Schwencker R, Schenkel F, Pronath M, Graeb H (2002) Analog circuit sizing using adaptive worst-case parameter sets. In: Design, automation and test in Europe conference and exhibition, Paris, 4–8 March 2002
68. Graeb HE (2007) Analog design centering and sizing. Springer, Netherlands
69. Chen YL, Wu WR, Liu CNJ, Li JCM (2014) Simultaneous optimization of analog circuits with reliability and variability for applications on flexible electronics. IEEE Trans Comput Aided Des Integr Circuits Syst 33(1):24–35
70. McConaghy T, Gielen GGE (2009) Globally reliable variation-aware sizing of analog integrated circuits via response surfaces and structural homotopy. IEEE Trans Comput Aided Des Integr Circuits Syst 28(11):1627–1640
71. Liu Bo, Fernandez FV, Bo Liu FV, Gielen GGE (2011) Efficient and accurate statistical analog yield optimization and variation-aware circuit sizing based on computational intelligence techniques. IEEE Trans Comput Aided Des Integr Circuits Syst 30(6):793–805
72. Singhee A, Rutenbar RA (2009) Statistical blockade: very fast statistical simulation and modeling of rare circuit events and its application to memory design. IEEE Trans Comput Aided Des Integr Circuits Syst 28(8):1176–1189
73. Kanj R, Joshi R, Nassif S (2006) Mixture importance sampling and its application to the analysis of SRAM designs in the presence of rare failure events. In: 43rd ACM/IEEE design automation conference, San Francisco, CA, 2006
74. Sun S, Li X, Liu H, Luo K, Gu B (2015) Fast statistical analysis of rare circuit failure events via scaled-sigma sampling for high-dimensional variation space. IEEE Trans Comput Aided Des Integr Circuits Syst 34(7):1096–1109
75. Maricaud E, Gielen G (2011) Computer-aided analog circuit design for reliability in nanometer CMOS. IEEE J Emerg Selected Topics Circuits Syst 1(1):50–58
76. Pan X, Graeb H (2010) Reliability analysis of analog circuits using quadratic lifetime worst-case distance prediction. In: 2010 IEEE custom integrated circuits conference (CICC), San Jose CA, 19–22 Sept 2010
77. Solido Design Automation (2015) Solido design automation <http://www.solidodesign.com/>. Accessed 12 April 2015
78. Hastings A (2005) The art of analog layout, 2nd edn. Prentice Hall, Upper Saddle River
79. Graeb HE (ed) (2010) Analog layout synthesis. Springer, Berlin
80. Ranjan M et al (2004) Fast, layout-inclusive analog circuit synthesis using pre-compiled parasitic-aware symbolic performance models. In: Design, automation and test in europe conference and exhibition, Paris, 2004
81. Liao YC, Chen YL, Cai XT, Liu CN, Chen TC (2013) LASER: layout-aware analog synthesis environment on laker. In: Proceedings of the 23rd ACM international conference on Great lakes symposium on VLSI, Paris, 2–4 May 2013
82. Vancorenland P, Van der Plas G, Steyaert M, Gielen G, Sansen W (2001) A layout-aware synthesis methodology for RF circuits. In: IEEE/ACM international conference computer-aided design, 2001
83. Youssef S, Javid F, Dupuis D, Iskander R, Louerat M-M (2011) A python-based layout-aware analog design methodology for nanometric technologies. In: 2011 IEEE 6th international design and test workshop (IDT), Beirut, 11–14 Dec 2011
84. Habal H, Graeb H (2011) Constraint-based layout-driven sizing of analog circuits. IEEE Trans Comput Aided Des Integr Circuits Syst 30(8):1089–1102

85. Toro-Frias A, Castro-Lopez R, Fernandez F (2012) An automated layout-aware design flow. In: International conference on synthesis, modeling, analysis and simulation methods and applications to circuit design (SMACD), Seville, 19–21 September 2012
86. Berkol G, Unutulmaz A, Afacan E, Dundar G, Fernandez FV, Pusane AE, Baskaya F (2015) A Two-step layout-in-the-loop design automation tool. In IEEE 13th international new circuits and systems conference (NEWCAS), Grenoble, 7–10 June 2015
87. Lourenço N, Canelas A, Póvoa R, Martins R, Horta N (2014) Floorplan-aware analog IC sizing and optimization based on topological constraints. *Integration, the VLSI Journal*, Elsevier 48:183–197
88. Lourenco N, Martins R, Horta N (2015) Layout-aware sizing of analog ICs using floorplan and routing estimates for parasitic extraction. In: 2015 Design, automation and test in europe conference and exhibition (DATE), Grenoble, 9–13 March 2015
89. Martins R, Lourenco N, Canelas A, Povoá R, Horta N (2015) AIDA: Robust layout-aware synthesis of analog ICs including sizing and layout generation. In: International conference on synthesis, modeling, analysis and simulation methods and applications to circuit design (SMACD), Istanbul, 7–9 Sept 2015
90. Paydavosi N et al BSIM4v4.8.0 MOSFET Model Manual. http://www-device.eecs.berkeley.edu/bsim/Files/BSIM4/BSIM480/BSIM480_Manual.pdf. Accessed 12 April 2016
91. Strasser M, Eick M, Grab H, Schlichtmann U, Johannes FM (2008) Deterministic analog circuit placement using hierarchically bounded enumeration and enhanced shape functions. In IEEE/ACM international conference computer-aided design, San Jose, 2008
92. Doboli A, Dhanwada N, Nunez-Aldana A, Vemuri R (2004) A two-layer library-based approach to synthesis of analog systems from VHDL-AMS specifications. *ACM Trans Des Autom Electron Syst* 9(2):238–271
93. Lourenço N, Horta N (2012) GENOM-POF: multi-objective evolutionary synthesis of analog ICs with corners validation. In: Genetic and evolutionary computation conference, Philadelphia, 2012
94. Póvoa R, Lourenço R, Lourenço N, Canelas A, Martins R, Horta N (2014) LC-VCO automatic synthesis using multi-objective evolutionary techniques. In: 2014 IEEE international symposium on circuits and systems (ISCAS), Melbourne VIC, 2014
95. Rocha F, Martins R, Lourenço N, Horta N (2014) Electronic design automation of analog ics combining gradient models with multi-objective evolutionary algorithms. Springer International Publishing
96. Rocha F, Lourenço N, Póvoa R, Martins R, Horta N (2014) A new metaheuristic combining gradient models with NSGA-II to enhance analog IC synthesis. In: 2013 IEEE congress on evolutionary computation, Cancun, 2013
97. Rocha F, Martins R, Lourenço N, Horta N (2013) Enhancing a layout-aware synthesis methodology for analog ICs by embedding statistical knowledge into the evolutionary optimization Kernel. In: Camarinha-Matos L, Tomic S, Graça P (eds) Technological innovation for the internet of things. 4th IFIP WG 5.5/SOCOLNET Doctoral Conference on computing, electrical and industrial systems, DoCEIS 2013, Costa de Caparica, Portugal, April 15–17, 2013. Proceedings, IFIP advances in information and communication technology, vol 394. Springer Berlin Heidelberg
98. Póvoa R, Lourenço R, Lourenço N, Canelas A, Martins R, Horta N (2015) Synthesis of LC-oscillators using rival multi-objective/multi-constraint optimization kernels. In: Fakhfakh M, Tlelo-Cuautle E, Fino M (eds) Performance optimization techniques in analog, mixed-signal, and radio-frequency circuit design. IGI Global, Hershey PA, pp 1–27
99. Martins R, Lourenço N, Canelas A, Horta N (2014) Electromigration-aware analog router with multilayer multiport terminal structures. *Integration, VLSI J* 47(4):532–547
100. Ning ZQ, Mouthaan T, Wallinga H (1991) SEAS: a simulated evolution approach for analog circuit synthesis. In: Custom integrated circuits conference, 1991
101. Chang H et al (1992) A top-down, constraint-driven design methodology for analog integrated circuits. In: Custom integrated circuits conference, 1992

102. Maulik PC, Carley LR, Rutenbar RA (1995) Integer programming based topology selection of cell-level analog circuits. *IEEE Trans Comput Aided Des Integr Circuits Syst* 14(4):401–412
103. Medeiro F, Perez-Verdu B, Rodriguez-Vazquez A, Huertas JL (1995) A vertically integrated tool for automated design of $\Sigma\Delta$ modulators. *IEEE J Solid-State Circuits* 30(7):762–772
104. Chang SJ, Hou HS, Su YK (2006) Automated passive filter synthesis using a novel tree representation and genetic programming. *IEEE Trans Evol Comput* 10(1):93–100

Automatic Analog IC Sizing and Optimization

Constrained with PVT Corners and Layout Effects

Lourenço, N.; VAT PT502854200, I. de T.; Gomes Horta,
N.C.

2017, XXVII, 182 p. 112 illus., 90 illus. in color.,

Hardcover

ISBN: 978-3-319-42036-3