Analog IC Design Using Precomputed Lookup Tables: Challenges and Solutions

ABDELRAHMAN A. YOUSSEF[ID]1, **BORIS MURMANN**[ID]2, **(Fellow, IEEE),**
AND HESHAM OMRAN[ID]1
[1] Integrated Circuits Laboratory (ICL), Faculty of Engineering, Ain Shams University, Cairo 11517, Egypt
[2] Department of Electrical Engineering, Stanford University, Stanford, CA 94305, USA

Corresponding author: Hesham Omran (hesham.omran@eng.asu.edu.eg)

ABSTRACT Design productivity remains an important aspect in the analog integrated circuit design industry, as growing competition and shorter design cycles pressure the traditional flow that involves time-consuming manual iterations in a circuit simulator. This paper describes innovations within an alternative framework that uses precomputed look-up tables (LUTs) to enable fast and accurate evaluation of circuit sizing scenarios without a simulator in the loop. It lets the designer explore and understand the design space boundaries in a systematic setting, thus supporting informed decision making and architectural innovation that is difficult to attain with fully automated, black-box sizing tools. Our discussion begins with an overview of the LUT-based design paradigm and its two primary variants: inverse design (finding design parameters that meet the specifications) and forward evaluation (sweeping design parameters to search the design space). In support of the latter, the core of our work focuses on improving the accuracy and speed of LUT access, enabling millions of queries within seconds on a standard computer. Large improvements over prior art are enabled using enhanced interpolation methods, which allow for a relatively large LUT grid spacing (hence small memory footprint) and yet accurate parameter lookup. We evaluate the efficacy of the proposed methods using two classical analog circuits, a bandgap reference and a folded cascode amplifier. In the bandgap example, we observe less than 1 ppm error between the LUT-predicted temperature coefficient and circuit simulation. In the folded-cascode example, one million design points are generated in only 4 seconds, providing the designer with useful maps that delineate the reachability of certain target specifications.

INDEX TERMS Systematic analog design, precomputed lookup tables, gm/ID methodology, analog design automation, interpolation, bandgap voltage reference, folded cascode OTA.

I. INTRODUCTION

The integrated circuits industry has seen dramatic developments over the past several decades, affecting fabrication processes, design methodologies, and computer-aided design (CAD) tools. Digital IC design has witnessed substantial enhancements in the design flow and the associated CAD tools. On the other hand, no fundamental changes have been brought to the analog IC design flow. Analog design is a complicated problem that requires dealing with numerous trade-offs and a large number of design variables; thus, it is not straightforward to find a design point that meets all the required design specifications. Analog designers usually depend on their experience to tweak the design variables

on a circuit simulator, starting from an initial design point using rough hand-analysis, until fulfilling the design requirements. This tedious ad-hoc process usually includes uninformed design decisions, and leads to sub-optimal designs. In addition, this monotonous and time-consuming process must be repeated for any change in the design specifications or the process technology. Consequently, the analog part in a complex chip is often the bottleneck in design cost and time. The problem gets worse with the increasing complexity and the stringent time-to-market requirements of state-of-the-art chips.

Several approaches have been proposed in the literature to address the productivity gap of the analog IC design flow [1]–[3]. One of the early approaches was the knowledge-based approach, where expert designers transform their heuristic design procedure (plan), which is based

The associate editor coordinating the review of this manuscript and approving it for publication was Omid Kavehei[ID].

on knowledge and experience, into a computer program. One main drawback of this approach is the large discrepancies between the results of the authored design plan and the simulation results, because the designer uses simplified models in the plan. The simulation-based optimization approach alleviates this drawback by relying on a circuit simulator to solve any arbitrary circuit using accurate and sophisticated device models. This approach was the one that survived in the market, and was implemented in the commercial design tools offered by major electronic-design automation (EDA) companies, e.g., [4]. However, the simulation-based approach suffers from several limitations that hindered its wide-spread acceptance in the design community. First, this approach relies on invoking the simulator at every iteration in the optimization procedure (i.e., SPICE-in-the-loop); thus, it suffers from long execution time, especially for circuits with large number of variables (degrees of freedom, DOFs). Second, in addition to the license of the optimization tool itself, the optimizer takes several seats from the pool of expensive simulator licenses shared by the designers. Third, the designer is completely detached from the circuit, as the optimizer does not offer insights into the circuit behavior, the achievable design metrics, or the trade-offs between different specifications.

A promising approach that reconnects the designer to the design problem while boosting the productivity is the use of precomputed lookup tables (LUTs) [5], [6]. The key idea in this design paradigm is to abstract the complex device models of modern devices in the form of LUTs. These LUTs are generated by the simulator for a set of reference devices once per technology. The designer can then use these LUTs to author systematic design plans for a circuit without invoking the simulator, while achieving simulator-accurate results [5], [7], [8]. This design scenario (depicted in Fig. 1a) resembles the knowledge-based approach; however, it replaces the simplified and inaccurate large-signal models with the simulator-accurate LUTs. Thus, there is no gap between the systematic design procedure and the simulation results. However, this design scenario does not address another important drawback in the knowledge based approach, which is the requirement to solve an inverse problem. Solving the inverse problem (find sizing and bias conditions of devices given design specifications) requires high-level of expertise, considerable time and effort, and is many times impossible. Moreover, for circuits with many DOFs, the expert designer must make assumptions regarding several DOFs in order to break deadlocks and simplify the design procedure [7], [8]. Consequently, creating a design plan for a new design problem is not a straight-forward task, and the results will not be optimal.

Another LUT-based design scenario that addresses the aforementioned problem is depicted in Fig. 1b. In this scenario, the inverse problem is replaced by the direct (forward) problem, i.e., find specifications given design point (sizing and biasing of devices). Solving the forward problem is significantly easier than the inverse problem, as it requires much
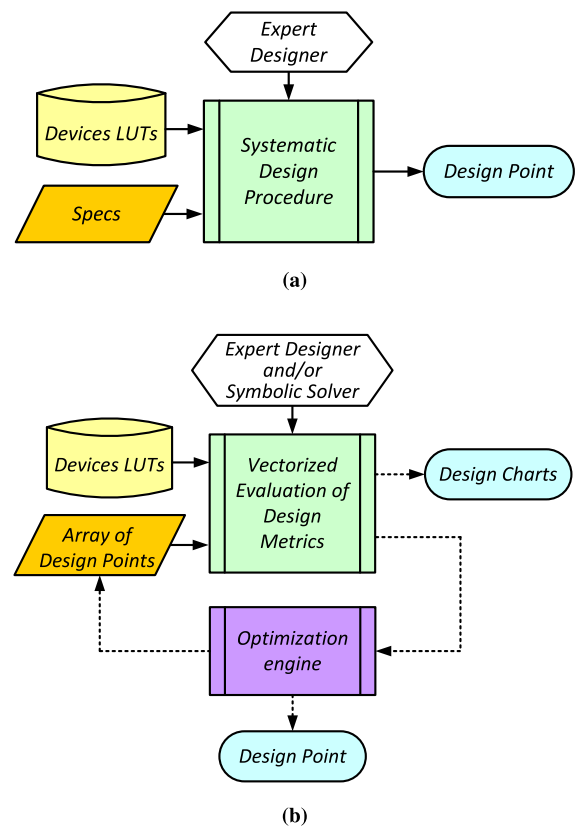
FIGURE 1. Design scenarios using precomputed LUTs: (a) Using a knowledge-based design plan to solve the inverse problem. (b) Solving the direct (forward) problem for an array of design points to search the design space.

less time, effort, and expertise. In addition, the process can be automated by using a symbolic circuit solver (e.g., [9]). Moreover, this scenario lends itself to vectorization, i.e., an array of design points can be processed simultaneously [10]. Two usage models are possible for this design scenario. First, the designer can generate design charts that show specs in the design space or illustrate the trade-offs between different specs. Second, an optimizer can be used to generate a new array of design points in a loop to search for the optimal design point that satisfies the required constraints. The later usage model resembles the simulation-based optimization approach, but it uses LUT-in-the-loop instead of SPICE-in-the-loop.

The aforementioned LUT-based design scenario can address the shortcomings of knowledge-based and optimization-based approaches, and set a new paradigm for analog IC design. The results are simulator-accurate, the execution time can be very fast, no simulator license required, design trade-offs are explored, optimal design point is targeted, and adding new design problems or circuit topologies does not require excessive effort or expertise. However, the recent implementations of the LUT-based design flow suffer from several limitations that must be addressed first [5], [6]. This papers aims at discussing the challenges

of the LUT-based design paradigm and proposing practical solutions.

The rest of the paper is organized as follows: Sec. II discusses the LUT-based analog IC design paradigm and its key challenges. Sec. III and IV propose solutions that address these challenges. Sec. V presents two design examples to illustrate the merits of the proposed solutions. Finally, the conclusions are presented in Sec. VI.

II. THE LUT-BASED ANALOG IC DESIGN PARADIGM

A. LUTs STRUCTURE

The lookup tables (LUTs) are tables that store the device characteristics across its different DOFs. LUTs can be created for any type of device, but this work focuses on the MOSFET as it is the ubiquitous device in analog IC design. The MOSFET is a four-terminal device; thus, its characteristics are controlled by three independent voltage differences, namely, V_{GS}, V_{DS}, and V_{SB}. The MOSFET characteristics also depend on the device sizing, i.e., the channel width (W) and length (L). For example, the drain current (I_D), which is the primary MOSFET dependent variable, can be written as a function of five independent variables (five DOFs)

$$I_D = f\left(W, L, V_{GS}, V_{DS}, V_{SB}\right). \tag{1}$$

Since, the MOSFET IV and CV characteristics are proportional to W regardless of the device operating region (i.e., bias conditions), we can rewrite (1) as

$$I_D = W \cdot f\left(L, V_{GS}, V_{DS}, V_{SB}\right). \tag{2}$$

Consequently, we can consider normalized quantities of different MOSFET parameters with respect to W, and the DOFs are reduced to four. The absolute quantities at any W can be simply computed using cross multiplication. The proportionality with W does not hold accurately for devices with small W due to narrow-width effects [5], [11]; however, narrow-width devices are not usually used in analog IC design. Thus, from a practical perspective, this small deviation can be ignored [5].

Using a circuit simulator, a nested sweep is performed for a reference device (a device with a reference W) across the four DOFs. Every parameter desired by the designer (e.g., I_D, g_m, etc.) can be saved in the form of a 4D LUT as illustrated in Fig. 2. Each LUT is a 4-D grid that stores the data of a specific parameter along the grid vectors $\{\overline{L}, \overline{V_{GS}}, \overline{V_{DS}}, \overline{V_{SB}}\}$. This procedure is repeated for every device, and the LUTs of each device can be grouped in a single structure for convenient access [12]. It should be noted that these LUTs are generated only once per technology, i.e., there is no need to invoke the circuit simulator again after LUTs generation. For detailed description on how to generate the LUTs for a given technology, the reader is referred to [5], [12].

B. DOFs IN LUT-BASED DESIGN

The role of the analog designer is to select a circuit topology that is expected to meet the design requirements, then
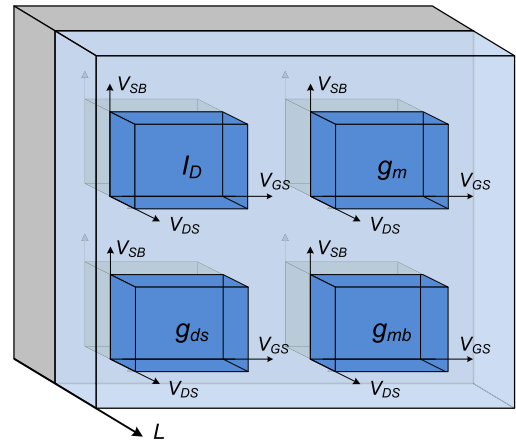

FIGURE 2. A simplified illustration showing a subset of MOSFET parameters (I_D, g_m, g_{ds}, g_{mb}) stored in 4D LUTs. All device parameters needed by the designer can be similarly stored in the LUTs.

find the biasing and sizing conditions of every device in the circuit to achieve a set of required specifications. Thus, strictly speaking, the designer must specify five DOFs ($W, L, V_{GS}, V_{DS}, V_{SB}$) for every device. However, for analog IC design the MOSFET is usually biased in saturation; thus, it basically operates as a voltage-controlled current source (VCCS). Consequently, V_{DS} is of secondary importance, and it is usually set to be above the drain-saturation voltage (V_{DSAT}) by some margin to guarantee that the device is biased in saturation. Moreover, V_{SB} is usually imposed by the circuit topology. Thus, the designer's task boils down to specifying only three DOFs (W, L, V_{GS}) for every device.

Analog circuits are usually biased by current mirrors, i.e., we set the device current (I_D) rather than the device voltage (V_{GS}). Thus, from a practical perspective, the DOFs become (W, L, I_D). In the conventional design flow, the designer (based on knowledge and experience) sweeps these DOFs (W, L, I_D) using a circuit simulator to explore the design space. However, biasing a given device using these DOFs is not straightforward because sweeping any of these three variables (W, L, I_D) changes the MOSFET inversion level (bias point); thus, the search process becomes complicated. Moreover, the search range of W is quite large, and it depends on both L and I_D. Replacing W in the DOFs with W/L does not help much because sweeping I_D still changes the inversion level, sweeping L changes the device physics (i.e., the IV characteristics) and consequently the inversion level, and the search range of W/L is still dependent on I_D.

The LUT-based design flow can address these shortcomings by replacing W with g_m/I_D, which is commonly referred to as the g_m/I_D design methodology [5], [7], [13]–[18]. The new set of DOFs ($g_m/I_D, L, I_D$) enables "orthogonal" search for the device design point. The MOSFET inversion level is solely determined by the g_m/I_D ratio, independent of L and I_D. When the designer sweeps L or I_D, the new corresponding W is retrieved from the LUTs while keeping g_m/I_D
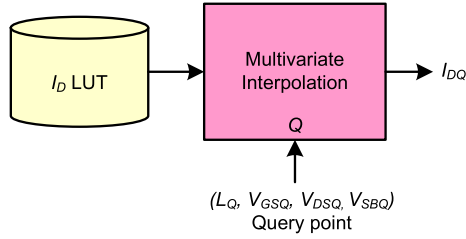
FIGURE 3. An example of a lookup operation for I_D. Multivariate linear interpolation is used in the conventional approach.

unchanged (i.e., bias point unchanged). Another benefit is that the search range of g_m/I_D is limited (typically 3 to 30 S/A), and is independent of L and I_D. The g_m/I_D methodology can be applied using design charts, but using LUTs enables automating the process [5], [7]. For deep subthreshold design, the g_m/I_D ratio saturates; thus, it can be replaced by the current density ($J_D = I_D/W$) as a single orthogonal DOF that controls the inversion level [5]. It should be noted that although V_{DS} and V_{SB} are not considered as primary designer DOFs, their effect are still taken into account as they are included in the LUTs.

C. CHALLENGES OF LUT-BASED DESIGN
The fundamental operation in the LUT-based design flow is the lookup operation, i.e., the operation of retrieving device parameters from the LUTs at a given query point. The merits of the LUT-based design paradigm can be true only if the following three requirements are satisfied:

1) the lookup operation is accurate;
2) the LUTs have reasonable size; and
3) the lookup operation is fast.

These three requirements are usually conflicting, so addressing them simultaneously is a threefold challenge.

The LUTs has inherently finite accuracy due to the finite steps used to build the 4D grids. When the query point (L_Q, V_{GSQ}, V_{DSQ}, V_{SBQ}) is off-grid, the lookup operation is basically a multivariate interpolation process as depicted in Fig. 3. Since the key advantage of the LUT-based design paradigm is being simulator-accurate, this interpolation process must maintain acceptable accuracy. The required accuracy may differ from one design to another, but in general high accuracy (e.g., error < 0.1%) is required for the design of high-precision circuits and for iterative procedures (to avoid error accumulation and/or divergence) [6], [8]. The lookup accuracy can be improved by:

1) using finer step size when building the LUTs; however, this comes at the expense of size and speed; and
2) using high-order interpolation; however, this comes at the expense of speed.

The second challenge is the LUT size, which is in direct conflict with the accuracy as previously mentioned. Halving the step size in the four dimensions results in 16-fold increase in the LUT size. Since the designer needs to save

several device parameters, the LUTs of a single device using relatively fine steps can easily jump to the GB range. The LUTs storage is not a major problem since the capacity of permanent storage is in the TB range. However, noting that the LUTs must be loaded to the RAM to perform the lookup operation, this may be a significant challenge. The problem becomes worse in modern technologies because there are many flavors for the MOSFET (e.g., high-V_T, normal-V_T, and low-V_T). Moreover, for a variation-aware design, the LUTs of every device must be extracted at several process and temperature corners. As a result, the amount of data to load in the memory can simply become impractical.

Besides accuracy and LUTs size, the performance of the lookup operation is another key requirement. This is especially important for the forward problem scenario (see Fig.1b) which involves performing a huge number of lookup operations. It is worth noting that replacing the SPICE-in-the-loop approach with LUT-in-the-loop will be attractive only if it offers substantial speedup. Using a simple interpolation procedure and small-size LUTs can help boosting the performance; however, both may come at the expense of accuracy. Therefore, using an efficient and vectorized implementation for the lookup operation is indispensable to boost the performance [10].

III. THE LOOKUP OPERATION
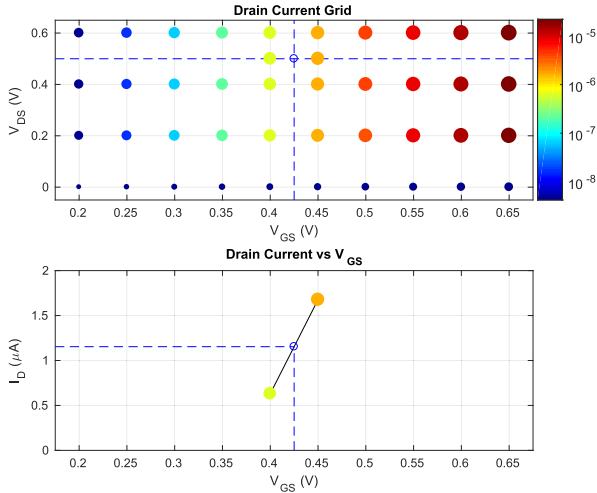### A. CONVENTIONAL LOOKUP
The LUT-based design methodology relies on interpolation to implement the lookup operation. Linear interpolation is one of the simplest methods to estimate values that lie between known data points, which we will refer to as "knots". Simply, the interpolant is formed by joining each two consecutive knots using a straight line as shown in Fig. 4. Each straight line is completely defined by specifying two unknowns; thus, only two points are needed to compute the interpolant, and the interpolant is unique. Linear interpolation has low computational complexity; however, it has poor accuracy, especially when the grid step is large. Moreover, the interpolant is not differentiable at the knots.
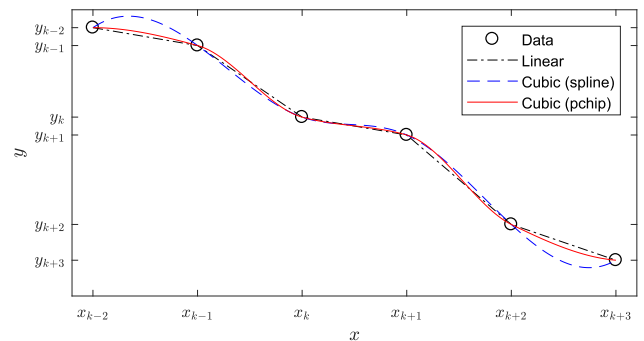
FIGURE 4. An example of linear, spline, and pchip interpolation.

The work in [5] and [12], which we will refer to as the conventional approach, uses linear interpolation for the lookup operation. A simplified illustration for the conventional lookup operation is shown in Fig. 5. The 2D grid

FIGURE 5. A simplified example for multivariate linear interpolation in conventional lookup. The procedure can be similarly applied to 4D data.

represents the drain current (I_D) along two dimensions: V_{GS} and V_{DS}. The value of I_D is represented by the color and size of every point. The grid is formed by using V_{GS} and V_{DS} vectors with a step of $50\,mV$ and $200\,mV$, respectively. To get I_D at the query point ($V_{GSQ} = 0.425\,V$, $V_{DSQ} = 0.5\,V$), first, linear interpolation is applied along V_{DS} to get I_D at the points that have V_{GS} values around V_{GSQ} ($0.4\,V < V_{GSQ} < 0.45\,V$). Then another linear interpolation is applied to get the value of I_{DQ} corresponding to V_{GSQ}. The same multivariate interpolation method can be applied on 4D LUTs. In addition to the poor accuracy of the linear interpolatin, the work in [5] and [12] does not support vectorization, i.e., it cannot handle a set of scattered query points simultaneously.

B. MODIFIED LOOKUP

In order to address the accuracy shortcoming in linear interpolation, piecewise cubic interpolating polynomials can be used to interpolate the data points. Each two adjacent knots are joined with a distinct cubic polynomial [19]. A cubic polynomial has four unkowns; thus, it is defined by four constraints. Consequently, function values at the two knots are not sufficient to define a unique cubic polynomial. Therefore, two additional points are used to estimate the first derivative values (i.e., slopes) at the knots. In order to guarantee continuity in the first derivative, the cubic Hermite interpolant between the two knots (x_k, y_k) and (x_{k+1}, y_{k+1}) is defined as [19]

$$P_k(x) = \frac{3hs^2 - 2s^3}{h^3}y_{k+1} + \frac{h^3 - 3hs^2 + 2s^3}{h^3}y_k$$
$$+ \frac{s^2(s-h)}{h^2}d_{k+1} + \frac{s(s-h)^2}{h^2}d_k. \quad (3)$$

where,

$$x_k < x < x_{k+1}$$
$$h = x_{k+1} - x_k, \quad s = x - x_k$$
$$d_k = P'_k(x_k), \quad d_{k+1} = P'_k(x_{k+1}).$$

Several methods can be used to estimate the slopes (d_k and d_{k+1}), i.e., there is no unique cubic interpolant. The most common methods are spline interpolation and shape-preserving pchip as shown in Fig. 4. In cubic spline interpolation, the slopes are calculated such that the interpolant has continuity in both first and second derivatives at each knot. Cubic splines have better smoothness than any other cubic interpolant, and they give the best results when interpolating smooth data. However, this approach has high computational complexity as it requires solving a system of equations to get the slope values. In addition, the monotonicity of the interpolant is not guaranteed even if the knots are monotonic (see Fig. 4). On the other hand, the pchip approach estimates the slopes to generate shape-preserving interpolants, i.e., monotonic interpolants for monotonic data [19], [20]. These interpolants have less smoothness than splines since they have continuity in the first derivatives only. However, the monotonic behavior is desirable in many applications. Moreover, it has low computational complexity as it does not require solving a system of equations to compute the slopes.

Since V_{GS} is the primary variable controlling the MOSFET behavior, the work in [6], which we will refer to as the modified approach, proposed that using cubic (pchip) interpolation in the V_{GS} dimension only was sufficient to perform accurate lookup operation for the design of high-precision circuits. The modified lookup operation is a two-step interpolation process, as shown in Fig. 6. First, it applies linear interpolation on all dimensions except V_{GS}, then 1D monotonic pchip is applied versus V_{GS} only. This is further illustrated in Fig. 7. The modified lookup operation applies linear interpolation along V_{DS} to the points that have V_{GS} values surrounding V_{GSQ} ($0.35\,V < 0.4\,V < V_{GSQ} < 0.45\,V < 0.5\,V$). Then the resultant values (points lying on the horizontal dashed line) are joined by a pchip interpolant versus V_{GS}. The monotonic pchip uses two-sided formula to estimate the slopes of the interior points [19], [20], which is why the drain current I_D values are evaluated at the outer points ($V_{GS} = 0.35\,V$ and $0.5\,V$). Once the slopes are determined, the interpolant is computed as given by (3), and the value of the drain current (I_{DQ}) corresponding to V_{GSQ} can be evaluated. It should be noted that if the query point is next to the endpoints of the grid vector, a one-sided formula is used to estimate the slope [19], [20]. One sided formulas usually results in worse estimations, so the interpolation error will be larger.
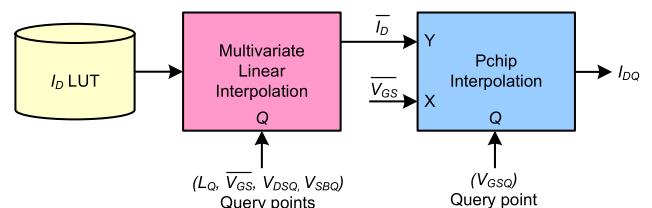
FIGURE 6. The modified lookup operation. Pchip is used for the V_{GS}-axis only.
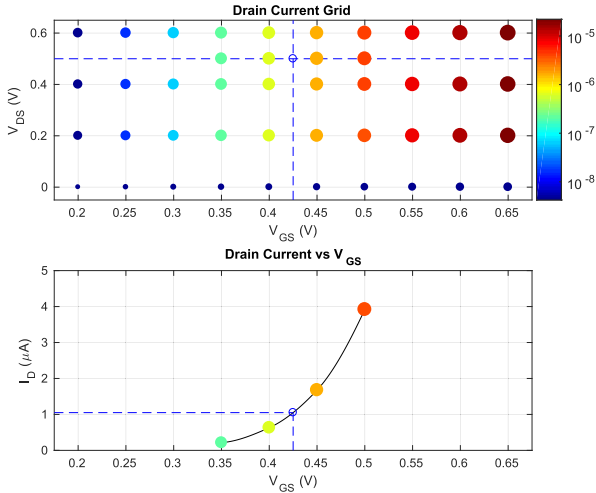
FIGURE 7. A simplified example for the modified lookup operation. The procedure can be similarly applied to 4D data.

The modified lookup operation has better accuracy but slightly lower speed compared to the conventional lookup operation. Both implementations are not vectorized, so they are inefficient for high-dimensional design space exploration.
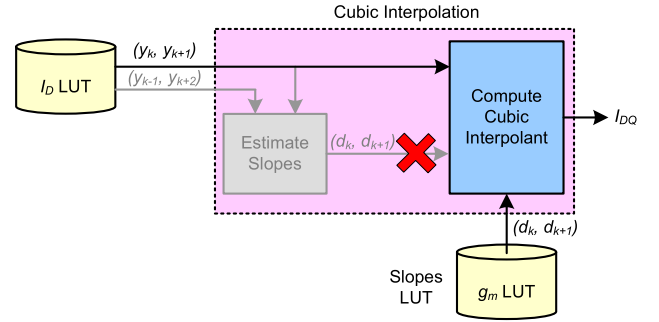
C. PROPOSED LOOKUP

The proposed lookup method aims at simultaneously addressing the three challenges discussed in Sec. II-C (accuracy, size, and speed). A key idea in the proposed method is to make use of the MOSFET parameters stored in the LUTs structure to enhance the interpolation process. For example, if it is required to lookup I_D, the modified lookup approach used pchip interpolation along the V_{GS} dimension to estimate the slopes. However, the slope values (d_k and d_{k+1}) used to define the interpolant in (3) need not be estimated using the pchip approach. Instead, the slopes can be extracted from the LUTs structure itself. The slope is defined by

$$\frac{\partial I_D}{\partial V_{GS}} = g_m. \tag{4}$$

Since the g_m LUT is already stored in the LUTs structure, slope values at the knots can be directly extracted from it, as depicted in Fig. 8. Since the g_m LUT is simulator-accurate, the slope values provided from the g_m LUT will provide much better accuracy compared to the mathematical estimations (e.g., spline and pchip). The improved accuracy is also achieved at the endpoints because the drawback of using a one-sided formula does not exist. In addition to the improved accuracy, the proposed approach enables higher speed and smaller LUT size. First, the slope estimation step is skipped, which improves the performance. Second, for a given accuracy, a larger grid step (i.e., smaller LUT size) can be used compared to the conventional and modified approaches, which reduces the LUTs size and also improves the performance.

The proposed lookup operation is illustrated in Fig. 9, which uses the same grid and the same query point as in Fig. 5

FIGURE 8. Simplified illustration for the proposed lookup operation. The slopes are retrieved from the LUTs themselves rather than being estimated.
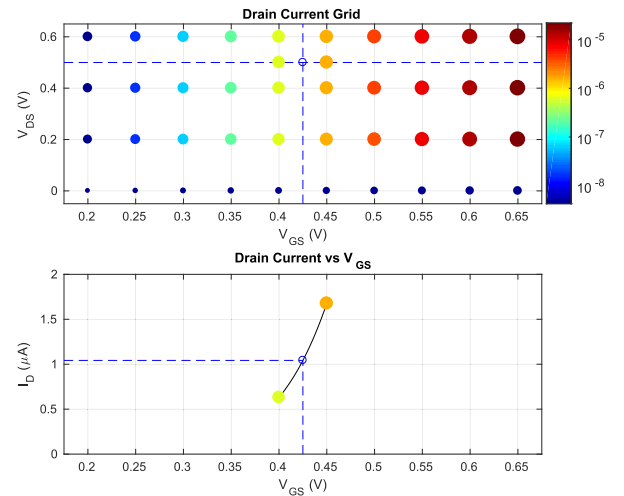
FIGURE 9. A simplified example for the proposed lookup operation. The procedure can be similarly applied to 4D data.

and Fig. 7. Linear interpolation along V_{DS} is applied at $V_{GS} = 0.4\,V$ and $0.45\,V$. The same linear interpolation is applied on the g_m grid. Then, the unique cubic interpolant of I_D versus V_{GS} is defined using (3), and the value of I_{DQ} corresponding to V_{GSQ} can be evaluated. Only two knots are needed in this case, since the slopes at these knots are provided from the g_m LUT. The same idea can be applied to interpolation versus V_{DS} and V_{SB} if needed, as the slopes (g_{ds} and g_{mb}, respectively) are also stored in the LUTs structure.

The accuracy of the proposed lookup operation can be further improved by taking into consideration the behavior of the MOSFET *IV* characteristics. In strong inversion (SI), I_D depends on V_{GS}^α, where α typically ranges from 1 to 2. Thus, using the cubic interpolant can faithfully mimic the actual *IV* characteristics. However, in weak inversion (WI), I_D has exponential dependence on V_{GS}. Thus, the cubic interpolant will not be able to follow the exponential trend, and the interpolation error will increase. In order to achieve consistent accuracy across all operating regions, the interpolation can be applied to $\ln I_D$ rather than I_D. The slopes at the knots will
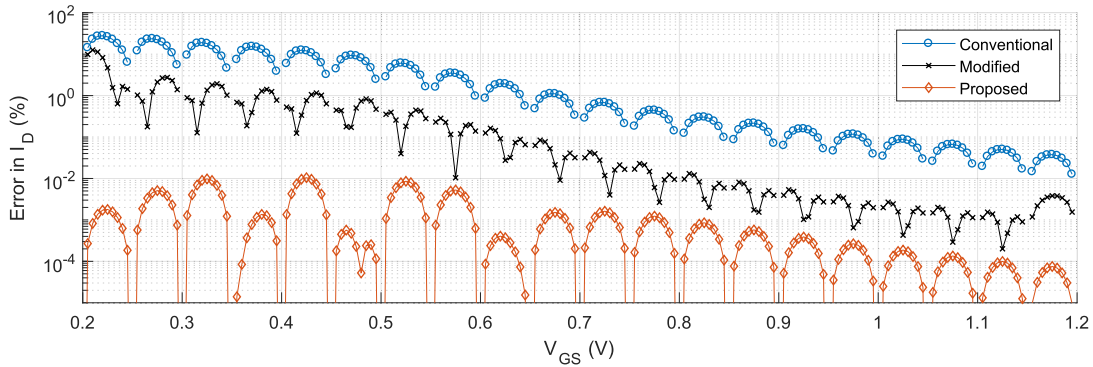
FIGURE 10. Lookup operation accuracy: percent error in I_D vs V_{GS}.

then be given by

$$\frac{\partial(\ln I_D)}{\partial V_{GS}} = \frac{g_m}{I_D}, \tag{5}$$

which can also be retrieved from the LUTs.

The previous discussion mainly considers I_D, which is the most important MOSFET parameter, especially when solving iterative procedures [6]. The second most important MOSFET parameter is the transconductance (g_m). Since the derivative of g_m is not available in the LUTs, the previous approach cannot be directly used. However, noting that the interpolant of I_D has been already obtained with improved accuracy using the proposed approach, the interpolant of g_m can be defined as the derivative of the interpolant of I_D, i.e., the derivative of (3)

$$P'_k(x) = \frac{6hs - 6s^2}{h^3} y_{k+1} + \frac{6s^2 - 6hs}{h^3} y_k$$
$$+ \frac{3s^2 - 2hs}{h^2} d_{k+1} + \frac{3s^2 - 4hs + h^2}{h^2} d_k. \tag{6}$$

This means that g_m uses a parabolic interpolant (i.e., second order polynomial). However, since the cubic interpolant of I_D is already accurate, this quadratic interpolant of g_m can give overall better accuracy than the cubic pchip used in the modified lookup approach, in addition to a performance advantage.

The proposed solutions indirectly provide some improvement in the performance of the lookup operation. However, significant additional performance enhancement can be achieved by using an efficient and vectorized implementation. The conventional and modified lookup approaches do not support vectorized operations. Moreover, every time the lookup operation is invoked it performs redundant and slow checks and preprocessing on the input arguments, in addition to creating a new gridded interpolant object for every call. In order to drastically enhance the performance, the proposed approach eliminates these redundant operations, and creates and stores the gridded interpolant objects themselves, rather than the raw 4D arrays. This is well aligned with the precomputing spirit promoted in the proposed work. Moreover, it has negligible impact on the LUTs size. Consequently, the time

of the lookup operation is substantially reduced. In addition, the proposed implementation supports performing the lookup operation on an array of scattered query points, which allows fast exploration of the design space.

D. RESULTS AND DISCUSSION

In order to compare the proposed lookup approach with the conventional and modified approaches, two LUTs structures were generated: coarse LUTs with V_{GS} step of $50\,mV$ and fine LUTs with V_{GS} step of $5\,mV$. LUTs data throughout this paper are generated from a $180\,nm$ CMOS technology; however, the precomputed LUTs design paradigm can be similarly applied to more advanced technologies [5]. The lookup operation is applied to the coarse LUTs to estimate the MOSFET parameters at every $5\,mV$ change in V_{GS}. The interpolation errors can be calculated by calculating the difference between the coarse LUTs estimated values and the fine LUTs simulator-accurate values.

The relative error of I_D is plotted versus V_{GS} in Fig. 10. Several observations can be made from this figure. First, all lines have nulls every $50\,mV$ because this is the coarse LUTs V_{GS} step. Second, the proposed approach achieves orders of magnitude smaller error compared to both conventional and modified approaches. Third, due to the use of the logarithmic transformation, the error reduction is significantly larger in WI and the error variation of the proposed approach across operating regions is much less than the conventional and modified approaches. Fourth, achieving an error less than $0.01\,\%$ using coarse LUTs of $50\,mV$ step enables significant reduction of LUTs size without sacrificing accuracy.

The relative error of g_m is plotted versus V_{GS} in Fig. 11. The proposed approach achieves orders of magnitude error reduction compared to the conventional approach. The lookup error in WI is more than one order of magnitude better than the modified approach, although the modified approach uses cubic interpolation. This does not hold in SI due to the logarithmic transformation applied in all regions. However, the proposed approach provides overall better results because the error is consistent across all operating regions, in addition to being sufficiently low. On the other hand, the error of the
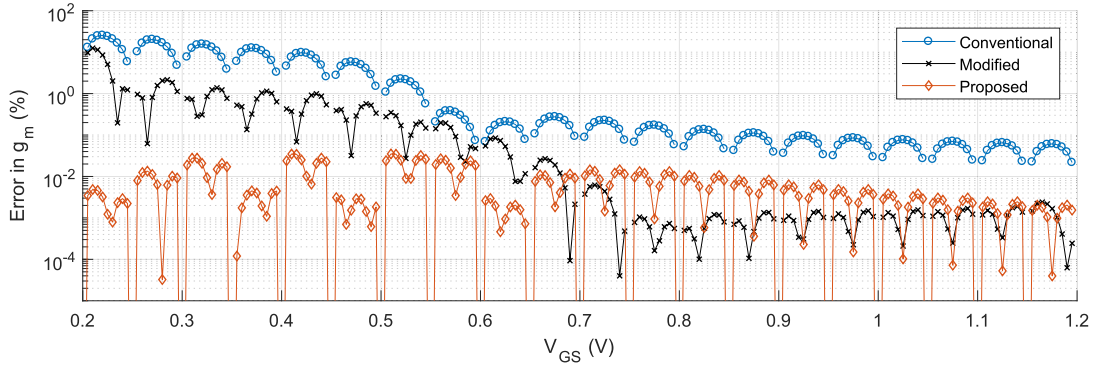
FIGURE 11. Lookup operation accuracy: percent error in g_m vs V_{GS}.
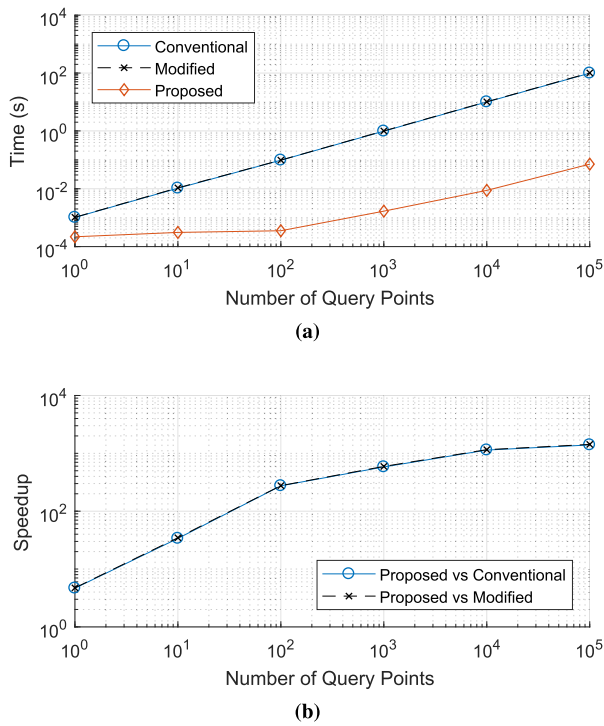

(a)

(b)

FIGURE 12. Performance of the lookup operation: (a) Comparison of lookup time vs number of query points. (b) Speedup of the proposed approach vs the conventional and modified approaches.

FIGURE 13. Conventional V_{GS} lookup operation using two-step interpolation.

becomes more than three orders of magnitude. This enables very fast exploration of high-dimensional design spaces as will be seen in Sec. V-B.

IV. THE V_{GS} LOOKUP OPERATION

A. CONVENTIONAL V_{GS} LOOKUP

The LUT-based design scenario uses g_m/I_D or J_D as a knob to set the device bias point as explained in Sec. II-B. Moreover, design procedures often include scenarios where the bias current (I_D) is known and it is required to get the bias voltage (V_{GS}) [6]. However, the normal lookup operation shown in Fig. 3 does not allow g_m/I_D or J_D in the query point. Therefore, another lookup operation is needed to look up V_{GS} given ($L, g_m/I_D, V_{DS}, V_{SB}$) or ($L, J_D, V_{DS}, V_{SB}$).

The conventional implementation of this V_{GS} lookup operation uses a two-step interpolation process that includes axes swapping as depicted in Fig. 13 [5], [12]. First, it computes J_D (or g_m/I_D), for all values in the V_{GS} grid vector. Next, the axes are swapped, i.e., J_D becomes the grid vector (X-axis) and V_{GS} is treated as the dependent variable (Y-axis), and 1D pchip interpolation is performed to get the required V_{GSQ} value corresponding to J_{DQ} in the query point. This is different from the modified lookup operation in Fig. 6, which does not involve swapping the axes. For J_D (or g_m/I_D) to be a valid grid vector, it must be strictly monotonic; thus, it may have to be trimmed before performing the 1D interpolation [5]. The modified lookup approach in [6] uses the same conventional approach for V_{GS} lookup operations.

The V_{GS} lookup operation is illustrated in Fig. 14, where it is required to get V_{GS} value corresponding to the query

modified approach varies by four orders of magnitude and is unacceptable in WI.

The performance of the proposed lookup is compared to the conventional and modified approaches in Fig. 12. All functions are implemented in MATLAB and the test is performed on the same machine (Core-i7-6500U CPU and 12 GB RAM) for a fair comparison. Since the execution time may fluctuate from one run to another, a large number of runs is performed using random query points, then the results are averaged out. Fig. 12 shows that there is a considerable speedup even for a single query point by virtue of removing the redundant checks and preprocessing. For a large number of query points, the vectorization kicks in, and the speedup
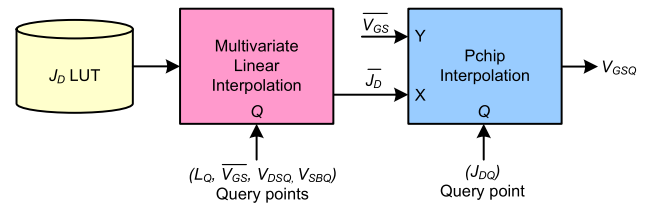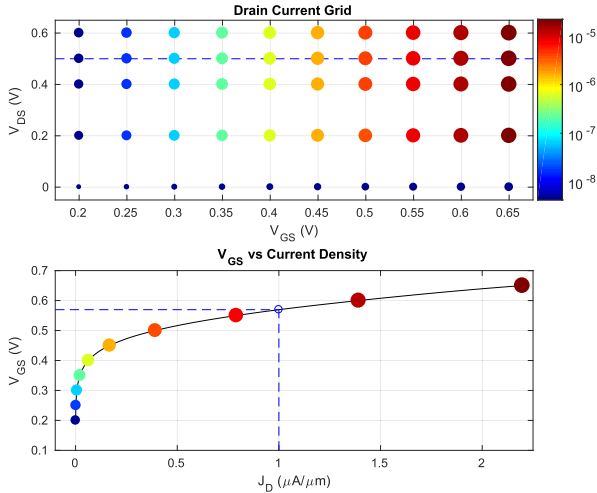
FIGURE 14. An example of the conventional V_{GS} lookup operation.

FIGURE 15. Proposed LUTs structure. Two new LUTs are added for fast and vectorized V_{GS} lookup operation.

point $J_{DQ} = 1\,\mu A/\mu m$ at $V_{DSQ} = 0.5\,V$. First, linear interpolation is applied to get all J_D values corresponding to the query vector formed using the grid vector of V_{GS} (points on the horizontal dashed line). Then the resultant points are plotted with J_D on X-axis and the corresponding V_{GS} values on Y-axis, and pchip interpolation is applied to get V_{GSQ} corresponding to J_{DQ}.

This V_{GS} lookup operation unnecessarily uses pchip interpolation to estimate the slopes for the cubic interpolant. In addition, the two-step interpolation process is slow, and cannot be vectorized for a set of scattered query points. Thus, it will represent a bottleneck in any LUT-based design procedure, even with the performance enhancement achieved for the lookup operation presented in Sec. III-D.

B. PROPOSED V_{GS} LOOKUP
In order to overcome the limitations of the inefficient two-step V_{GS} lookup operation, we can resort to the key technique in the proposed LUT-based design paradigm: precomputation. The two-step process depicted in Fig. 13 can be done in advance by going through all the possible query vectors in a nested loop (i.e., iterate through every L, V_{DS}, and V_{SB}). The resultant is two new LUTs added to the LUTs structure: V_{GS} vs $\overline{\ln J_D}$ and $\overline{g_m/I_D}$ grid vectors as shown in Fig. 15. It is important to use $\ln J_D$ rather than J_D as the grid vector since J_D spans several orders of magnitude. Precomputing the new V_{GS} LUTs is done only once per technology similar to the original LUTs structure. It should be noted that these two new LUTs are precomputed from the LUTs structure itself without invoking the simulator again. Since the LUTs structure already contains many LUTs for different MOSFET parameters, adding the two new LUTs will have a minor effect on its size.

Using these two new LUTs, the V_{GS} lookup operation can be treated similar to the normal lookup operation discussed in Sec. III. Moreover, the idea of using the data in the LUTs to estimate the slopes of the cubic interpolant can be similarly
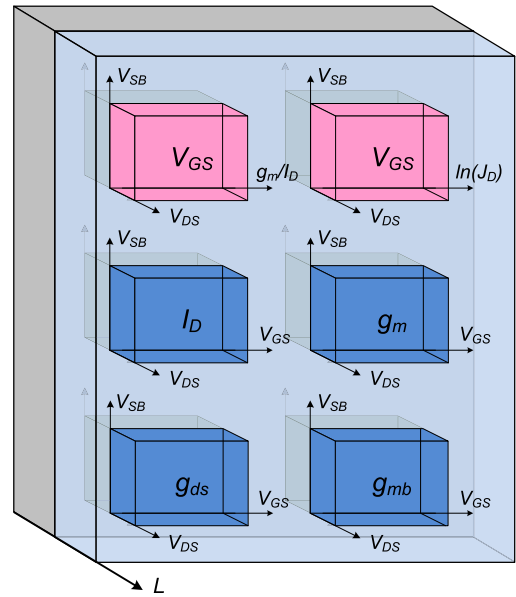
applied. To look up V_{GS} given $\ln J_D$, the slopes are given by

$$\frac{\partial V_{GS}}{\partial (\ln J_D)} = \frac{J_D \cdot W}{g_m}, \qquad (7)$$

which can be retrieved from the LUTs, where W is the width of the reference device. As previously discussed, the simulator-accurate slopes will yield more accurate interpolants. These slopes can be also used while building the V_{GS} LUTs in the precomputation step.

C. RESULTS AND DISCUSSION
Similar to Sec. III-D, a coarse LUT with $50\,mV$ V_{GS} step and a fine LUT with $5\,mV$ V_{GS} step are used to compare the accuracy of the conventional and proposed V_{GS} lookup operations. J_D values that correspond to the V_{GS} grid vector are extracted from the fine LUT. Then this J_D vector is used as query points for the V_{GS} lookup operation using the coarse LUT. The interpolation error is calculated as the difference between the interpolated V_{GS} and the fine LUT V_{GS} grid vector. The error for the conventional and proposed approaches is shown in Fig. 16. Although the conventional method uses pchip interpolation, the proposed method achieves significantly better accuracy, especially in WI where the error is reduced by up to two orders of magnitude.

The performance of the proposed V_{GS} lookup is compared to the conventional method in Fig. 17. For a single query point, the speedup is more than one order of magnitude. As the number of points increases, the speedup approaches four orders of magnitude. Due to eliminating the two-step interpolation with axes swapping (see Fig. 13), the V_{GS} lookup speedup is one order of magnitude higher than the speedup of the normal lookup operation shown in Fig 12.
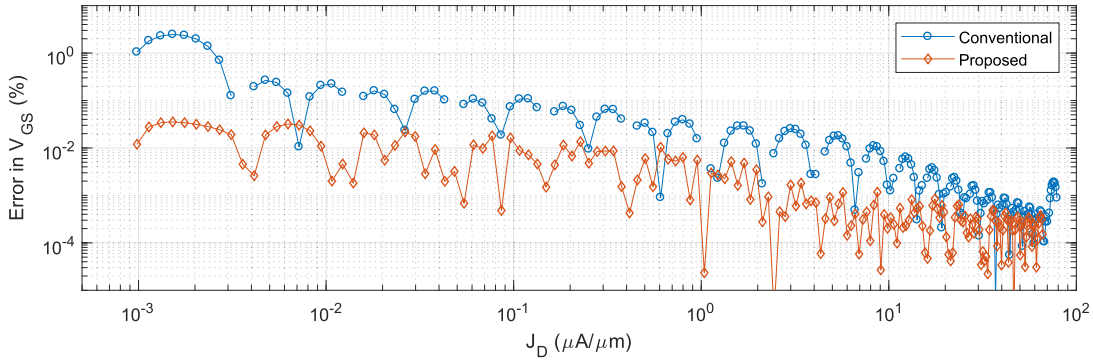
FIGURE 16. Accuracy of V_{GS} lookup operation: percent error in V_{GS} vs J_D. The modified lookup approach in [6] is the same as the conventional approach for V_{GS} lookup operations.
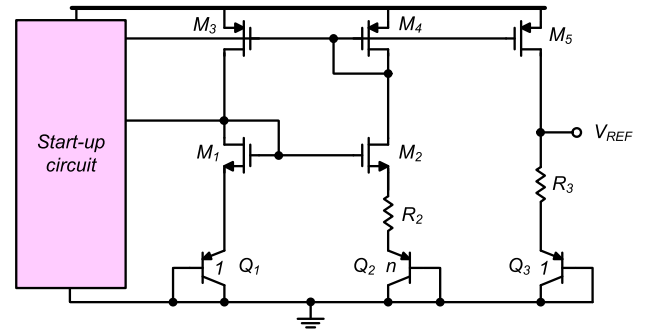

(a)

(b)
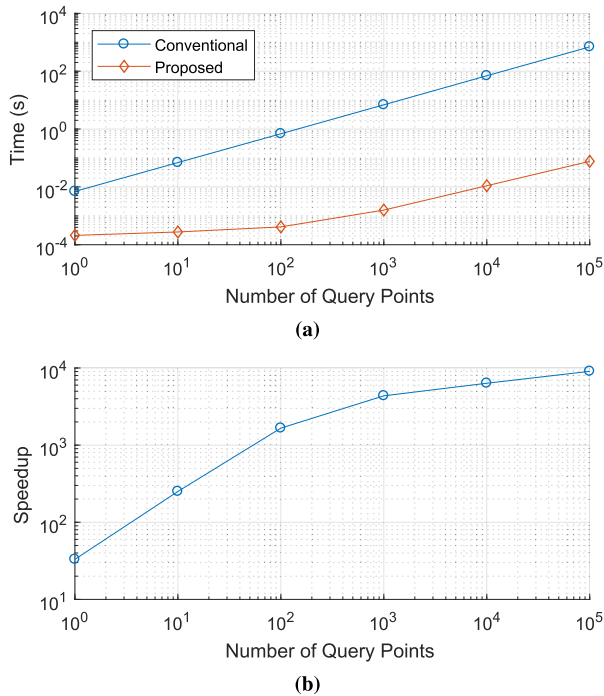
FIGURE 17. Performance of the V_{GS} lookup operation: (a) Comparison of lookup time vs number of query points. (b) Speedup of the proposed approach vs the conventional approach. The modified lookup approach in [6] is the same as the conventional approach for V_{GS} lookup operations.

FIGURE 18. Schematic of the bandgap voltage reference circuit used as a design example.

V. DESIGN EXAMPLES

In order to appreciate the importance of the solutions proposed in this work, it is important to put them in the context of practical design examples. Two design examples are discussed in this section. The first example is a bandgap voltage reference circuit, which benefits mainly from the improved accuracy and the reduced LUTs size. The second example is a folded cascode OTA, which shows how the vectorized and fast lookup implementation can be used to explore the design trade-offs in multi-dimensional design problems.

A. BANDGAP VOLTAGE REFERENCE

The bandgap voltage reference is a precision circuit that is sensitive to small errors; thus, it is a good example to show the effect of lookup errors. The design procedure of the bandgap circuit shown in Fig. 18 is presented in detail in [6]. One of the key metrics of this circuit is the dependence of the output reference voltage V_{REF} on temperature, which is characterized by the temperature coefficient (TC). For a temperature range from T_{MIN} to T_{MAX}, TC in ppm is defined as

$$TC\ (ppm) = \frac{10^6}{V_{REF,T_{NOM}}} \cdot \frac{V_{REF,MAX} - V_{REF,MIN}}{T_{MAX} - T_{MIN}}. \quad (8)$$

where T_{NOM} is the nominal temperature.

The circuit is synthesized using LUTs with $25\,mV$ V_{GS} step, and the synthesized circuit is simulated using Cadence Spectre to compare synthesis results against simulations. The synthesis procedure involves both normal lookup operations and V_{GS} lookup operations. The lookup approaches discussed in Sec. III and Sec. IV are compared in Fig. 19. The results show V_{REF} vs temperature at $\rho_N = \rho_P = 20\,S/A$, where ρ_N and ρ_P are the g_m/I_D ratio of the NMOS and PMOS transistors, respectively. In addition, Fig. 19 shows the contours of TC in the $\rho_N - \rho_P$ space in order to study the effect of the transistor operating region on the results. As evident from Fig. 19a, the conventional approach fails to provide accurate results, and the difference between synthesis and simulation is quite large. The modified approach provides better results as shown in Fig. 19b; however, the contour plot shows that the error is significant when the devices are

(a)

(b)

(c)
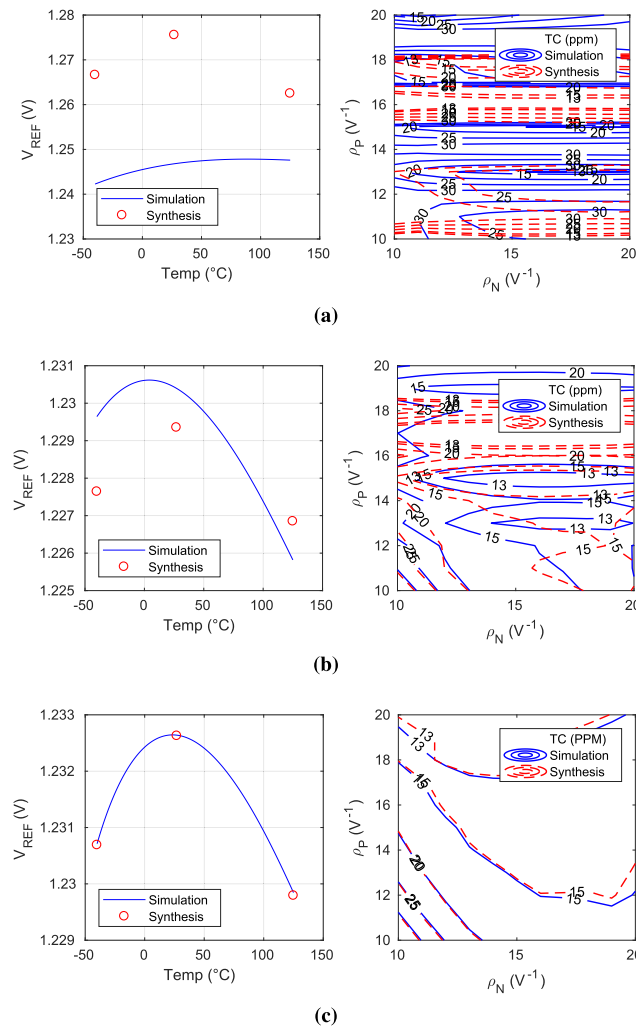
FIGURE 19. Comparison of synthesis and simulation results for the bandgap reference circuit showing V_{REF} vs temperature and *TC* in the $\rho_N - \rho_P$ space, where ρ_N and ρ_P are the g_m/I_D ratio of the NMOS and PMOS transistors, respectively. (a) The conventional lookup approach. (b) The modified lookup approach. (c) The proposed lookup approach.

biased in WI. Fig. 19c shows that the proposed approach achieves impressing matching between synthesis and simulation results across all operating regions with less than 1 ppm error. Achieving this level of accuracy using $25\,mV\ V_{GS}$ step enables easily generating tables at different temperature and process corners, which is essential in the design of this type of circuits. In addition to PVT variations, process mismatch can be also taken into account using the LUTs as shown in [6].

B. FOLDED CASCODE OTA

The folded cascode is one of the most popular OTA topologies due to its flexible input range. Fig. 20 shows a schematic of a fully differential folded cascode OTA. Due to symmetry, the half-circuit principle can be applied, and the designer needs to specify the DOFs of five transistors only. According to the discussion in Sec. II-B, each transistor has three DOFs ($L, \rho\ =\ g_m/I_D, I_D$). However, assuming the bias
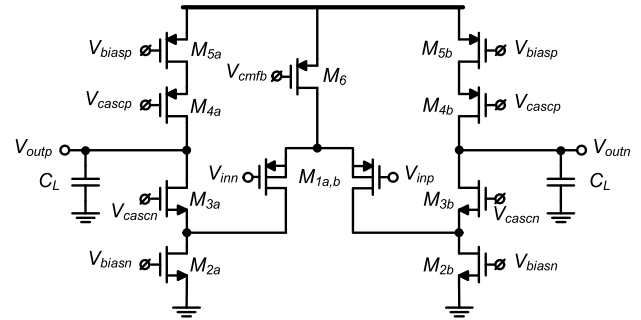

FIGURE 20. Schematic of the folded cascode OTA used as a design example.

current is evenly split between the common-source and the common-gate stages [21], the bias current of all transistors boils down to a single DOF, and the total number of DOFs is reduced to 11.

In order to explore the design space, a set of 10^6 design points is generated. For every design point, the two DOFs of every transistor (L and ρ) are randomly selected from a uniform distribution. L is selected in the interval ($0.2\,\mu m$, $2\,\mu m$) and ρ is selected in the interval ($5\,S/A$, $25\,S/A$). This allows exploring short/long channel behavior and WI/SI biasing. The total bias current (I_B) is randomly selected from a uniform logarithmic distribution in the interval ($0.1\,mA$, $10\,mA$). Using randomly generated design points gives a more uniform exploration for the design space compared to a grid search. The load capacitance (C_L) is considered an external constraint, and is set to $2\,pF$ in this design example. The design metrics are computed using vectorized evaluation of symbolic expressions according to the scenario in Fig. 1b. It should be noted that this does not sacrifice the accuracy because all MOSFET parameters substituted in the symbolic equations are extracted from the LUTs.

Remarkably, synthesizing the 10^6 design points and computing the design metrics takes $4\,s$ only. Generating this large dataset using the conventional lookup method requires more than three orders of magnitude more time. On the other hand, it will require impractical time using simulation-based techniques. This huge amount of data gives the designer endless possibilities to gain insights and examine design trade-offs using design charts. As an example, Fig. 21a shows the design points in the DC gain (A_{vo}) vs gain-bandwidth product (GBW) space. Moreover, phase margin (PM) and fan-out (FO) constraints are applied to the design, where FO is defined as the ratio of the load capacitance to the input capacitance. The chart in Fig. 21a tells what range of gain is achievable? What range of GBW is achievable? How PM and FO affect the achievable gain or GBW? Is putting more current going to improve or hurt the GBW given the gain is kept constant? All these questions cannot be answered using the simulation-based optimization approach.

Fig. 21b shows another design chart which explores the power-speed trade-off of the OTA, and how this trade-off is
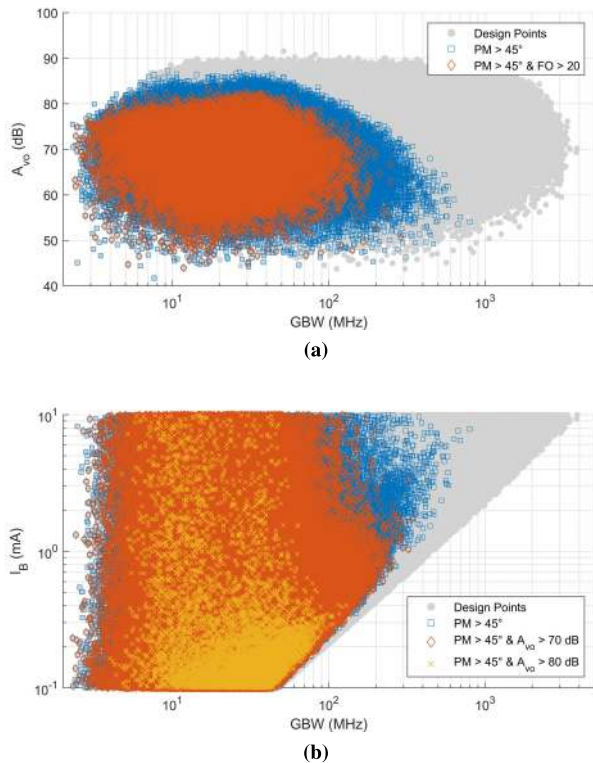
FIGURE 21. Design charts of folded cascode OTA. (a) DC gain vs *GBW* with *PM* and *FO* constraints. (b) Bias current vs *GBW* with *PM* and DC gain constraints.

affected by *PM* and gain constraints. The maximum achievable *GBW* is reduced by one order of magnitude due to the gain specification regardless of the current consumption. Moreover, for a given *GBW*, *PM*, and gain, the bias current of an unoptimized design may be two orders of magnitude inferior to an optimized design. Trade-offs with other design metrics such as noise and area can be similarly explored. Moreover, as in Fig. 1b, the designer can use a local or global optimizer to minimize an objective, e.g., bias current, while satisfying a set of constraints.

VI. CONCLUSION

This paper discussed advantages and challenges of the LUT-based design flow for analog circuits. Specifically, it identified the need for fast and accurate table lookup to enable rapid exploration of the circuit's performance space using direct (forward) evaluation of its underlying design equations. The presented solution uses enhanced interpolation methods that facilitate large LUT grid spacing while maintaining highly accurate lookup. The latter is validated using a highly sensitive bandgap design that achieves ppm-level accuracy between the synthesized and SPICE-simulated designs. In addition to being accurate, the lookup method developed in this work is also fast, enabling millions of queries within seconds. Such functionality can be used to search a circuit's design space, providing rapid feedback to the designer for feasibility studies, topology

selection and guidance for the invention of new circuits. It is conceivable to couple our fast evaluation method to an advanced optimization tool to navigate optimal sizing conditions and topology changes without a simulator in the loop.

REFERENCES

[1] S. Pandit, C. Mandal, and A. Patra, *Nano-Scale CMOS Analog Circuits: Models and CAD Techniques for High-Level Design*. Boca Raton, FL, USA: CRC Press, 2014.

[2] M. F. Barros, J. M. Guilherme, and N. C. G. Horta, *Analog Circuits and Systems Optimization Based on Evolutionary Computation Techniques*. Berlin, Germany: Springer, 2010.

[3] R. A. Rutenbar, G. G. E. Gielen, and B. A. Antao, Eds., *Computer-Aided Design of Analog Integrated Circuits and Systems*. Hoboken, NJ, USA: Wiley, 2002.

[4] Cadence Design Systems. *Virtuoso Analog Design Environment GXL*. Accessed: Jun. 1, 2020. [Online]. Available: https://www.cadence.com

[5] P. Jespers and B. Murmann, *Systematic Design of Analog CMOS Circuits Using Pre-Computed Lookup Tables*. Cambridge, U.K.: Cambridge Univ. Press, 2017.

[6] H. Omran, M. H. Amer, and A. M. Mansour, "Systematic design of bandgap voltage reference using precomputed lookup tables," *IEEE Access*, vol. 7, pp. 100131–100142, 2019.

[7] M. N. Sabry, H. Omran, and M. Dessouky, "Systematic design and optimization of operational transconductance amplifier using g_m/I_D design methodology," *Microelectron. J.*, vol. 75, pp. 87–96, May 2018.

[8] M. N. Sabry, I. Nashaat, and H. Omran, "Automated design and optimization flow for fully-differential switched capacitor amplifiers using recycling folded cascode OTA," *Microelectron. J.*, vol. 101, Jul. 2020, Art. no. 104814.

[9] A. Montagne. *SLiCAP: Symbolic Linear Circuit Analysis Program*. Accessed: Jun. 1, 2020. [Online]. Available: https://www.analog-electronics.eu/slicap

[10] The MathWorks. *MATLAB Documentation*. Accessed: Jun. 1, 2020. [Online]. Available: https://www.mathworks.com/help/matlab/matlab_prog/vectorization.html

[11] J. Ou and P. M. Ferreira, "Implications of small geometry effects on g_m/I_D based design methodology for analog circuits," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 66, no. 1, pp. 81–85, Jan. 2019.

[12] B. Murmann. g_m/I_D *Starter Kit*. Accessed: Jun. 1, 2020. [Online]. Available: https://web.stanford.edu/~murmann/gmid

[13] F. Silveira, D. Flandre, and P. G. A. Jespers, "A g_m/I_D based methodology for the design of CMOS analog circuits and its application to the synthesis of a silicon-on-insulator micropower OTA," *IEEE J. Solid-State Circuits*, vol. 31, no. 9, pp. 1314–1319, Sep. 1996.

[14] R. Fiorelli, E. J. Peralias, and F. Silveira, "LC-VCO design optimization methodology based on the g_m/I_D ratio for nanometer CMOS technologies," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 7, pp. 1822–1831, Jul. 2011.

[15] F. T. Gebreyohannes, J. Porte, M. Louërat, and H. Aboushady, "A g_m/I_D methodology based data-driven search algorithm for the design of multi-stage multi-path feed-forward-compensated amplifiers targeting high speed continuous-time $\Sigma\Delta$-modulators," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, early access, Jan. 7, 2020, doi: 10.1109/TCAD.2020.2966998.

[16] G. Piccinni, C. Talarico, G. Avitabile, and G. Coviello, "Innovative strategy for mixer design optimization based on g_m/I_D methodology," *Electronics*, vol. 8, no. 9, p. 954, Aug. 2019.

[17] J. Ou and P. M. Ferreira, "A g_m/I_D-based noise optimization for CMOS folded-cascode operational amplifier," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 61, no. 10, pp. 783–787, Oct. 2014.

[18] P. Jespers, *The g_m/I_D Methodology, a Sizing Tool for Low-Voltage Analog CMOS Circuits: The Semi-Empirical and Compact Model Approaches*. Boston, MA, USA: Springer, 2010.

[19] C. B. Moler, *Numerical Computing With MATLAB*, vol. 87. Philadelphia, PA, USA: SIAM, 2008.

[20] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM J. Numer. Anal.*, vol. 17, no. 2, pp. 238–246, Apr. 1980.

[21] H. Omran, "Optimum split ratio for folded cascode OTA bias current: A qualitative and quantitative study," in *Proc. 31st Int. Conf. Microelectron. (ICM)*, Dec. 2019, pp. 223–226.

ABDELRAHMAN A. YOUSSEF received the B.Sc. degree (Hons.) in electrical engineering from Ain Shams University, Cairo, Egypt, in 2019. He is currently a Research Assistant with the Integrated Circuits Laboratory (ICL), Ain Shams University. His research interests include design of analog and mixed-signal integrated circuits and design automation.

BORIS MURMANN (Fellow, IEEE) received the Dipl.Ing. degree in communications engineering from the Fachhochschule Dieburg, Dieburg, Germany, in 1994, the M.S. degree in electrical engineering from Santa Clara University, Santa Clara, CA, USA, in 1999, and the Ph.D. degree in electrical engineering from the University of California at Berkeley, Berkeley, CA, USA, in 2003. From 1994 to 1997, he was with Neutron Mikrolektronik GmbH, Hanau, Germany, where he was involved with the development of low-power and the smart-power application-specified integrated circuits (ASICs) in automotive CMOS technology. Since 2004, he has been with the Department of Electrical Engineering, Stanford University, Stanford, CA, USA, where he is currently a Full Professor. His current research interests include mixed-signal integrated circuit design, with a special emphasis on data converters, sensor interfaces, and circuits for embedded machine learning. He was a co-recipient of the Best Student Paper Award from the Very Large-Scale Integration (VLSI) Circuits Symposium, in 2008. He was a recipient of the Best Invited Paper Award from the IEEE Custom Integrated Circuits Conference (CICC), in 2008, the Agilent Early Career Professor Award, in 2009, and the Friedrich Wilhelm Bessel Research Award, in 2012. He served as the Data Converter Subcommittee Chair and the 2017 Program Chair for the IEEE International Solid-State Circuits Conference (ISSCC). He served as an Associate Editor for the IEEE JOURNAL OF SOLID-STATE CIRCUITS.

HESHAM OMRAN received the B.Sc. (Hons.) and M.Sc. degrees in electrical engineering from Ain Shams University, Cairo, Egypt, in 2007 and 2010, respectively, and the Ph.D. degree in electrical engineering from the King Abdullah University of Science and Technology (KAUST), Saudi Arabia, in 2015. From 2008 to 2011, he was a Research and a Teaching Assistant with the Integrated Circuits Laboratory (ICL), Ain Shams University, and a Design Engineer with Si-Ware Systems (SWS), Cairo, where he worked on the circuit and system design of the first miniaturized FT-IR MEMS spectrometer (NeoSpectra). From 2011 to 2016, he was a Researcher with the Sensors Laboratory, KAUST. He held Internships with the Bosch Research and Technology Center, Sunnyvale, CA, USA, and Mentor Graphics, Cairo. In 2016, he joined ICL, Ain Shams University, as an Assistant Professor. He has published more than 30 papers in international journals and conferences. His research interests include design of analog and mixed-signal integrated circuits, especially in analog and mixed-signal design automation.

. . .