PROCEED: A Pareto Optimization-based Circuit-level Evaluator for Emerging Devices

Shaodi Wang, Andrew Pan, Chi On Chui, and Puneet Gupta

Department of Electrical Engineering
University of California, Los Angeles
Los Angeles, CA 90095
E-mail: shaodiwang@g.ucla.edu

***Abstract*—Evaluation of novel devices in a circuit context is crucial to identifying and maximizing their value. We propose a new framework, PROCEED, and metrics for accurate device-circuit co-evaluation through proper optimization of digital circuit benchmarks. PROCEED assesses technology suitability over a wide operating region (MHz to GHz) by leveraging available circuit knobs (V_t assignment, power management, sizing, etc.) and improves accuracy by 3X to 115X compared to existing methods while offering orders of magnitude improvements in runtime over full physical design implementation flows. To illustrate PROCEED's capabilities, we deploy it to assess novel tunneling transistors (TFETs) compared to conventional CMOS.**

***Index Terms*—Tunneling transistor (TFET), silicon-on-insulator (SOI), circuit-level device evaluation, Pareto optimization, simulation-based optimization.**

I. INTRODUCTION

As traditional silicon devices approach their fundamental limits, it is important to explore additions or alternatives to CMOS. To do so, it is essential to systematically compare emerging devices in the context of the circuits they would be used to build. Many technology benchmarking methods have been proposed to meet this need [1]-[9]; unfortunately, as summarized in Table I, all those methods are inadequate due to neglect of various essential circuit features, any one of which can dramatically alter the benchmarking conclusions. Because of the variety and complexity of modern circuits, devices and circuit designs must be carefully chosen to complement each other before assessing viability; this requires a level of flexibility in the benchmarking process that has not existed until now.

Device/circuit assessments must consider several factors to draw realistic conclusions. For instance, *effective evaluations should examine the power-delay (PD) tradeoff over several orders of magnitude* since modern circuits' performances span a wide range from KHz to GHz frequencies. For a particular circuit to be properly used, *crucial tuning knobs such as logic gate sizing or supply voltage (V_{dd}) or threshold voltage (V_t) selection must be optimized*. In addition, since circuit performance depends critically on the chosen device operating point, *benchmarks should consider the full device I-V characteristics rather than only simple device metrics* like saturation current I_{on} or off-state leakage I_{off}. A given device may not be suitable for all circuit architectures because of *variations in logic depth histogram (LDH) patterns and logical or physical structure*. *Such adaptivity and circuit topology must be considered in any assessment.* Meanwhile, as technologies scale down, *device variability from ambient process fluctuations becomes ever more important* and impacts circuit viability. Such complexities might seem to require a complete circuit design flow, but that is impractically time-consuming. Thus, alternative evaluation method must be used *which accounts for the above factors with reasonable computational run time*.

To meet these needs, we propose a new device evaluation framework, PROCEED (PaReto Optimization-based Circuit-level Evaluator for Emerging Devices), for fully circuit-aware benchmarking. It incorporates typical circuit design flow flexibilities and tunes physically adjustable device and circuit parameters to generate realistic conclusions about the combined device-circuit performance. PROCEED remedies the flaws enumerated above in several ways:

(1) We use Pareto curves to analyze PD tradeoff over a realistically wide range of power and performance.

(2) The range and number of V_t as well as range of logic gate sizes are inputs to PROCEED, and the evaluation circuit benchmarks can use one or several V_{dd} supply voltages, in accord with realistic designs.

Table 1 COMPARISON OF VARIABLES CONSIDERED IN BENCHMARK METHODOLOGIES IN THE LITERATURE

Methodologies:		Ref. [1]	Ref. [2,4]	Ref. [3-4]	Ref. [5]	Ref. [6]	Ref. [7]	Ref. [8]	Ref. [9]	PROCEED
Metrics		CV/I, CV^2, I_{on}/I_{off}	*PD* Pareto Curves, I_{on}/I_{off}	*PD* Pareto Curves	Clock Frequency	*SS*, I_{on}	Energy, Clock Frequency	CV/I, CV^2, Model Power, Delay	CPI	***PD* Pareto Curves**
Benchmark Circuit		Latch, Inverter Chain	µP	µP	µP	Device	Inverter Chain	Small Logic Elements	µP	Arbitrary Circuit (µP here)
Power management		×	×	×	×	×	×	×	×	✓
Optimization Knobs	V_{DD}, V_t	✓	✓	✓	×	×	✓	×	***V_{DD} only***	✓
	Size	×	×	×	✓	×	×	×	✓	✓
	Multiple V_{DD}, V_t	×	×	×	✓	×	×	×	×	✓
Circuit Conditions	Interconnect	✓	✓	✓	✓	×	✓	✓	✓	✓
	LDH*	×	×	×	×	×	×	×	×	✓
	Activity	✓	✓	×	✓	×	✓	✓	×	✓
Device Model	Current	I_{on}, I_{off}	I_{on}, I_{half}, I_{off}	I_{on}, I_{half}, I_{off}	I_{on}, I_{off}	TCAD	Model	Model	R_{on}, R_{off}	**Compact Model**
	Capacitance	Fixed	Fixed	Full *C-V*	Fixed	N/A	Fixed	Full *C-V*	-	**Full *C-V***

*LDH: Logic depth histogram (Using slack histogram to estimate)

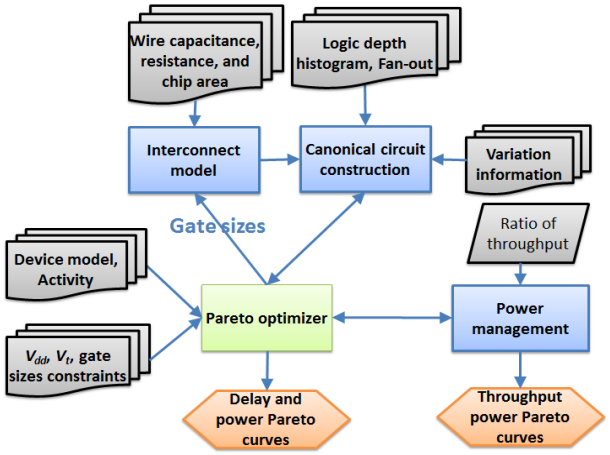

Fig. 1. Overview of PROCEED framework.

(3) To properly account for device operation at each bias, we utilize compact or lookup table-based full device models.

(4) To assess circuit topology, the full chip characteristics are considered including LDH, interconnect loads, activity factor (i.e. average gate toggle rate) and average fan-out.

(5) We analyze the circuit impact of device variability due to factors like random dopant fluctuation (RDF) and parasitic voltage drops by calculating delay for logic gates evaluated at different variation corners.

(6) For computational efficiency, we adopt scalable Pareto optimization techniques.

(7) Power gating and dynamic voltage and frequency scaling (DVFS) are modeled to assess power management and scaling.

In this paper, we describe the PROCEED framework and, as a case study, deploy it to compare a traditional technology, silicon-on-insulator (SOI), with the novel tunneling FET (TFET). The TFET is a new device concept currently drawing intense interest because of its potential for highly energy efficient operation due to its steep subthreshold switching [10]. However, comprehensive assessments of its system-level performance are still lacking; therefore we perform a microprocessor-level study of the SOI benchmark technologies and elucidate their respective strengths and disadvantages. We outline the methodology behind PROCEED in section II, and explain details of the Pareto optimization procedure in section III. We present results of our PROCEED study on TFET and SOI devices in section IV and summarize our conclusions in section V.

II. Overview of PROCEED Framework

As shown in Fig. 1, typical inputs to PROCEED include interconnect information (including average wire resistance and capacitance (*RC*) and chip size), benchmark design (i.e. design LDH and average fan-out), variability (through supply voltage

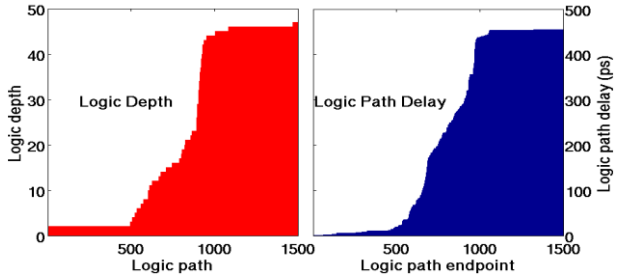

Fig. 2 Typical logic path depth distribution and logic path delay extracted from a synthesized CortexM0.

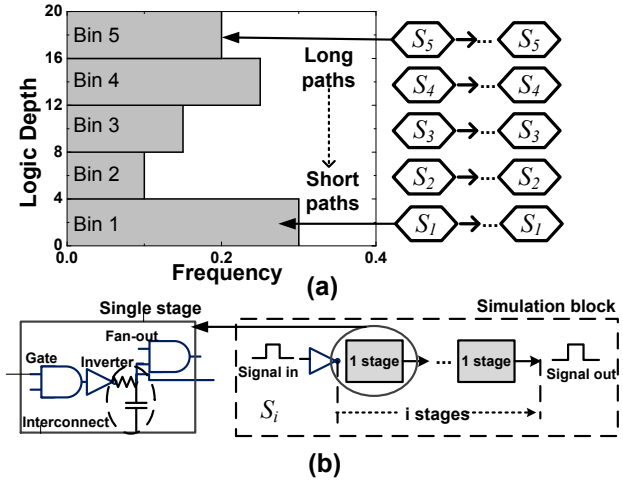

Fig. 3 Circuit schematic for simulation and optimization.

drops, threshold voltage shifts, etc.), a full device model, operating activity, and optional constraints on V_{dd}, V_t, chip area, and ratio of average to peak throughput. With input and feedback from the Pareto optimizer (through tuning parameters like V_{dd}, V_t, and gate sizes), the needed simulation blocks with interconnect loads are created in the canonical circuit construction process. Optimized results are generated in the form of the *PD* Pareto curve. Finally, power management analysis including DVFS and power gating is performed based on this Pareto curve. As presently implemented, PROCEED is capable of evaluating an arbitrary device candidate as long as it does not cause a dramatic change in circuit topology. For instance, multistate logic devices fall outside PROCEED's present scope of use because of the unconventional circuit architectures within which they must operate.

A. Canonical Circuit Construction

Full, exact optimization is an impossible job for large digital circuits. Since the goal of our approach is to predict the best performance and power tradeoffs for emerging devices, detailed circuit design is thus not our target and contributes little to evaluation. We utilize therefore only essential design information to maximize performance and determine the optimal V_{dd} , V_t, and gate sizes at a given power. A typical circuit design contains both long and short logic paths and the path delay is usually proportional to the logic depth, as shown in Fig. 2. Hence we derive the LDH by extracting endpoint slacks from benchmark designs and estimating logic paths. In Fig. 3, we show an example of the simulation blocks used to construct a specific circuit. For simplicity, we first divide logic paths into n bins based on logic depth; in Fig. 3(a), for instance, $n = 5$. More bins improve accuracy at the expense of computation time. Each bin is modeled by the corresponding simulation blocks S_i (S_1-S_5 in Fig. 3(a)), which are in turn made of i gate stages. We use the gate design for S_i to construct logic paths belonging to a given bin i. The LDH is divided such that the longest path in each bin has the same delay if all these blocks have the same delay. Fig. 3 shows an example of this, with five evenly spaced bins for logic paths from one to twenty stages such that the first bin contains one to four stage paths, the second holds paths with five to eight stages, and so forth. The delay weight W_D is the number of copies of S_i needed to construct the longest path in bin i (W_D is 4 in Fig. 3). The logic gate and interconnect used for a single stage in the simulation blocks is shown in Fig. 3(b). The gate can be NAND, NOR, or a more complicated gate like XNOR, depending on the average number of transistors per gate in a given benchmark. The gate choice can also differ from bin to bin, though in this paper's examples we will

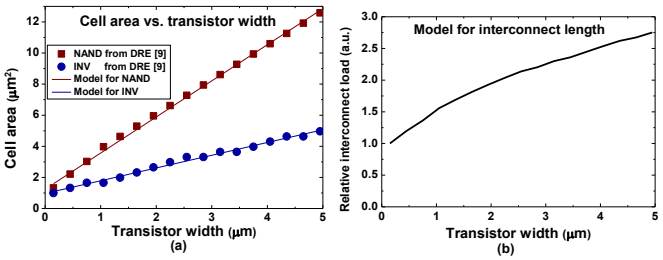

Fig. 4. (a) Cell area and (b) interconnect load as a function of transistor width. In (b), transistor width is the same in Inverter and NAND gate.

use NAND gates for all bins. An inverter or buffer is inserted after the gate to drive the fan-out (which is a replica of the chosen gate sized to average fan-out) as well as wires represented by interconnect *RC* elements. We have verified the reliability of the PROCEED results through comparison with commercial synthesis tools, as discussed in Section IV.A.

B. Process Variation and Voltage Drop

As devices scale to ever smaller technology nodes, device variations due to process and ambient variations are becoming more important and should not be neglected in PD evaluation. In circuit design, slow corner devices are commonly used to estimate the upper bound on delay and create a "safe" design with sufficient delay margin. We define the slow corner as a device with reduced effective V_{dd} and increased V_t due to variability and parasitic effects; these voltage shifts are inputs to PROCEED. Separate models for other variability effects may be incorporated as needed. During circuit optimization, delay is calculated using the slow corner device while power is simulated with the normal device to model the worst-case scenario.

C. Interconnect Load

We model interconnect loads using a series *RC* circuit. To construct load as a function of gate width, we use UCLADRE[1] [11] to find a relation between cell area and gate width, and then fit linear models to each cell used in PROCEED. The model accuracy is demonstrated in Fig. 4(a). We assume *R* and *C* are linear with interconnect length and chip area is linear to cell area, so the load will be proportional to the square root of the average cell area [12], and can be dynamically changed based on average gate width. Shown in Fig. 4(b) is an example of interconnect load as a function of transistor width, using a combined NAND and INV cell to estimate the cell area. The average *RC* and extracted gate width are then fed into PROCEED.

D. Pareto-Based Optimization

Following logic canonical circuit construction, all logic paths are replaced by simulation blocks (S_i) which will be optimized. However, these blocks cannot be optimized separately because they usually share a common V_{dd} and V_t, complicating the procedure. To perform the optimization we use a modified form of a general simulation-based Pareto technique [13] which is discussed in more detail in Section III. The simulation target is regarded as a black box with two optimization objectives: design power *P* and critical delay *D* (or minimum working clock period).

E. Power Management Modeling

Current technologies usually allow circuits to operate in at least three modes: normal, power saving, and sleep mode. Previous evaluation works only considered the normal mode when devices

[1]Freely available for download at
http://nanocad.ee.ucla.edu/Main/DownloadForm

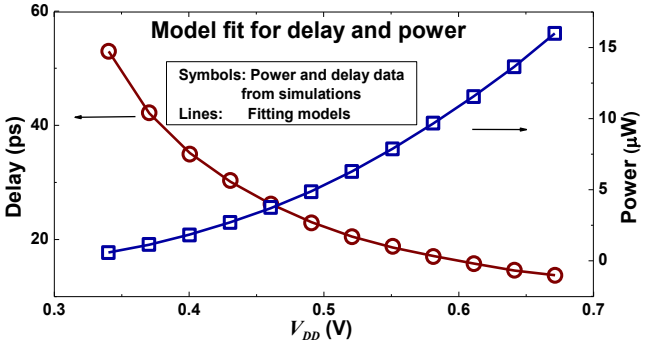

Fig. 5. Model fitting for simulation block's delay and power as a function of V_{dd}.

continuously work at peak performance. PROCEED allows devices to also operate at a second, lower supply V_{dd2} (DVFS) as well as in the off state (power gating). This allows us to evaluate device *PD* scalability as a function of V_{dd}, an important feature which, to the best of our knowledge, has been ignored in all previous evaluations.

The ratio of average to peak throughput is another input for PROCEED. To study power management, we choose all designs from the generated Pareto points which achieve the lowest power and peak throughput. From this, the optimizer selects the best choice for the second power rail and divides the time spent operating at high V_{dd1} (the original supply) and the new lower V_{dd2}. This is done as follows. Starting from the optimized design (with maximized peak throughput), we carry out circuit simulations by sweeping voltages lower than the original V_{dd1}. The original design may even have multiple supply voltages, in which case different blocks can use different V_{dd2} values. Delay and power models for every simulation block S_i as functions of V_{dd} are constructed using polynomial functions, as in Fig. 5:

$$D_{Si}(V)=\sum_{j=-2}^{5} a_{i,j}V^i,\ P_{Si}(V)=\sum_{j=-1}^{5} b_{i,j}V^i \qquad (1)$$

We have tested and found this model to be sufficiently accurate; for instance, in our experiments the relative error of the polynomial fittings is less than 2%. We then optimize for the weighted power sum $f_1P_1 + f_2P_2$, subject to

$$\begin{aligned} &D_2 >= W_D D_{Si}(V_{i2}),\ P_2 >= \sum_{i=1}^{n} W_{Pi}P_{Si}(V_{i2}) \qquad i=1,2,...,n \\ &f_1\cdot 1/D_1 + f_2\cdot 1/D_2 >= T_{Ave}, 0\le f_1+f_2\le 1 \end{aligned} \qquad (2)$$

Here $D_{1,2}$ and $P_{1,2}$ are the delay and power using $V_{dd1,2}$, W_D and W_P are the delay and power weight mapping from simulation blocks to the design, and f_1 and f_2 are the fractions of time spent operating with V_{dd1} and V_{dd2} with any remaining time assumed to be spent in the off state. Typically this step is not a feasible convex optimization problem; however, by using the fitted model of Eq. (1), an enumeration approach can solve this problem very efficiently with acceptable accuracy.

F. Activity Factor

Activity varies widely with application: in embedded sensing, for instance, factors below 1% are observed in car-park management [14], while those for systems like VigilNet exceed 50% [15]. Activity factor can therefore dramatically change evaluation results and is included as an input to PROCEED. In circuit simulations, the dynamic and leakage power are separately extracted and the total power is their weighted sum. From this the circuit can be optimized for a known activity factor.

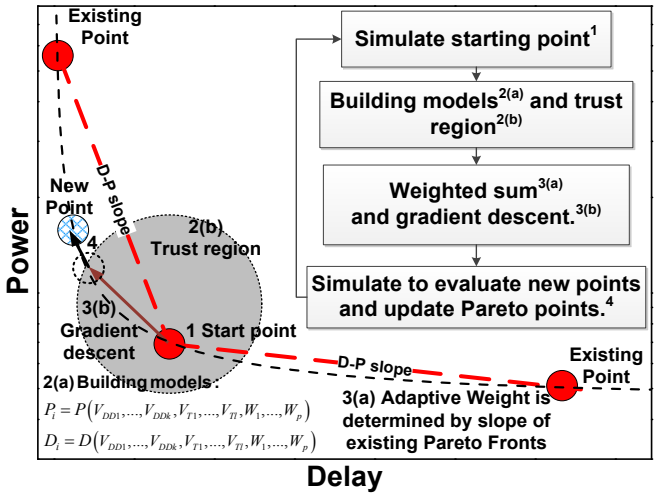

Fig. 6 Optimizer overview. Adaptive weight is chosen by slope of existing fronts. Based on starting point, meta-modeling is built and gradient descent is used to find potential points. Simulate potential points to get new Pareto points.

G. *Multiple V_{dd} and V_t*

In modern circuit designs, multiple V_{dd} and V_t values are used. In our scheme, transistors in each simulation block S_i must be assigned the same voltages, so to optimize a design with integer m different V_{dd} or V_t biases, the number of simulation blocks must be greater than m. In addition, our optimization is an iterative process whereby Pareto points are updated and improved based on previous iterations. Therefore, if the same V_{dd} or V_t is shared by multiple simulation blocks, this assignment cannot be changed during the optimization. A full optimization for multiple V_{dd} and V_t is implemented by considering designs with all sets of reasonable voltage assignments in parallel. For example, if we have five simulation blocks S_1-S_5 and two available threshold voltages, then for i from 1 to 4, blocks S_1 to S_i use the high V_t and S_{i+1} to S_5 use the low V_t. This comprises the set of useful voltage assignments, since simulation blocks with longer logic paths require higher performance (lower V_t).

III. Pareto Optimization

Fig. 6 presents an overview of our Pareto optimization process. PROCEED treats circuit simulations as a black box and uses models to optimize tuning parameters based on the simulation results. Gradient descent is utilized to find minimal objectives in the trust region. Final simulations are performed on designs outputted by the model-based optimization. The vector of tuning parameters $\boldsymbol{X}$ for optimization is represented as:

$$\begin{aligned}\boldsymbol{X} &= (x_1, x_2, ..., x_m) = (\boldsymbol{y_1}, \boldsymbol{y_2}, ..., \boldsymbol{y_n}) \\ \boldsymbol{y_i} &= (V_{dd,i}, V_{t,i}, W_{i1}, W_{i2}, ..., W_{t,2i}), i = 1,...n\end{aligned} \tag{3}$$

where $V_{dd,i}$ and $V_{t,i}$ are the supply and threshold voltages for simulation block S_i, W_{ij} are sizes for gates and inverters in S_i, x_j are the variables of $\boldsymbol{X}$, and $\boldsymbol{y_i}$ are vectors of the tuning parameter variables for S_i. The optimization entails the following steps:

(1) *Picking a starting point*: each iteration of the optimization process uses a starting set of variables $\boldsymbol{X_0}$ around which to explore. For the first iteration, any reasonable $\boldsymbol{X_0}$ may be inputted. The choice of the initial point may affect runtime but not final accuracy, since "bad points" will gradually be eliminated by the optimization process and converge to the true answer. Subsequently $\boldsymbol{X_0}$ is determined from already existing Pareto points by computing the Euclidean distance between all neighboring points in delay/power coordinates, as shown in Fig. 6. The point with the largest total distance from its two neighbors is chosen as $\boldsymbol{X_0}$ since it lies in the sparse region, which is usually suboptimal.

(2) *Building a local model around $\boldsymbol{X_0}$*: To accelerate the optimization process, second-order delay and power models are constructed based on simulation results. The delay and power models D_{Si} and P_{Si} for each block S_i are calculated separately and then combined to reduce the number of simulations, as determined by the size of the Hessian matrix (proportional to the number of variables squared). D_{Si} and P_{Si} are represented by the gradient vector $\boldsymbol{G_{Di}}$ and Hessian matrix $\boldsymbol{H_D}$ as

$$\begin{aligned}D_{Si}(\boldsymbol{y_{i,0}} + \boldsymbol{\Delta y_i}) &= D_{Si,0} + \boldsymbol{G_{Di}^T \Delta y_i} + \frac{1}{2}\boldsymbol{\Delta y_i^T H_{Di} \Delta y_i} \\ P_{Si}(\boldsymbol{y_{i,0}} + \boldsymbol{\Delta y_i}) &= P_{Si,0} + \boldsymbol{G_{Pi}^T \Delta y_i} + \frac{1}{2}\boldsymbol{\Delta y_i^T H_{Pi} \Delta y_i}\end{aligned} \tag{4}$$

This second-order model is a local estimation near the starting point. To guarantee validity, an adaptive trust region is applied as shown in Fig. 6, limiting the model range inside the region

$$\boldsymbol{X_0 - \lambda(r) < X < X_0 + \lambda(r)} \tag{5}$$

where r is the radius of this "trust region" and $\boldsymbol{\lambda}$ is the range of the tuning parameters $\boldsymbol{X}$ and is a linear function of r.

(3) *Model-based optimization*: In this step, four metrics are used in optimization: D, P, $W_{dl}\times D + W_{pl}\times P$, and $W_{dr}\times D + W_{pr}\times P$. Minimization of D and P yields the fastest and lowest power designs in the local region, while the weighted sums of delay and power are used to populate the phase space by finding two Pareto points between the starting point and its neighbors. Since the problem may not be convex, gradient descent with the logarithmic barrier method [16] is used to find these optimal points. The model's region of validity lies in the intersection of the trust region and the inputted bounds for the tuning parameters. The objective function is performed as follows:

$$\begin{aligned}&Minimize\ W_D D(\boldsymbol{X}) + W_P P(\boldsymbol{X}) - t\left(\sum_{j=1}^{m}\log(-x_j + x_{j,u}) - \sum_{j=1}^{m}\log(x_j - x_{j,l})\right) \\ &D(\boldsymbol{X}) = D(\boldsymbol{X_0}) + \boldsymbol{G_D}(\boldsymbol{X_0})^T(\boldsymbol{X} - \boldsymbol{X_0}) + (\boldsymbol{X} - \boldsymbol{X_0})^T \boldsymbol{H_D}(\boldsymbol{X_0})(\boldsymbol{X} - \boldsymbol{X_0}) \\ &P(\boldsymbol{X}) = P(\boldsymbol{X_0}) + \boldsymbol{G_P}(\boldsymbol{X_0})^T(\boldsymbol{X} - \boldsymbol{X_0}) + (\boldsymbol{X} - \boldsymbol{X_0})^T \boldsymbol{H_P}(\boldsymbol{X_0})(\boldsymbol{X} - \boldsymbol{X_0})\end{aligned} \tag{6}$$

where $x_{j,l}$ and $x_{j,u}$ are the upper and lower bounds for variable x_j, and D and P are delay and power for the entire design, respectively. The weights for delay and power are defined as follows:

$$\begin{aligned}W_{dl(r)} &= (P_{l(r)} - P_0) / \sqrt{(P_{l(r)} - P_0)^2 + (D_{l(r)} - D_0)^2} \\ W_{pl(r)} &= (D_0 - D_{l(r)}) / \sqrt{(P_{l(r)} - P_0)^2 + (D_{l(r)} - D_0)^2}\end{aligned} \tag{7}$$

where (D_0, P_0) is the starting point and (D_l, P_l) and (D_r, P_r) are the left and right neighbor points, respectively. The solid points in Fig. 6 are examples of such points. The direction vectors (W_{dl}, W_{pl}) and (W_{dr}, W_{pr}) of the weighted sum of objectives are calculated so as to be perpendicular to the connecting lines between the starting point and its neighbors, as illustrated by the dashed line in Fig. 6. D and P are given by

$$D(\boldsymbol{X}) = W_D \cdot \max\left((D_{S1}(\boldsymbol{y_1}), D_{S2}(\boldsymbol{y_2}), ..., D_{Sn}(\boldsymbol{y_n}))\right), P = \sum_{i=1}^{n} W_i \cdot P_{Si} \tag{8}$$

where W_D is the delay weight discussed in Section II.A and $\boldsymbol{W_i}$ is the number of S_i used in the canonical circuit construction. Because the maximizing function does not have a continuous derivative, we use higher order norms to estimate the maximum, so the elements of gradient vector and Hessian matrix for delay are derived as follows:

$$\begin{aligned}&D(\boldsymbol{X}) \approx \|\boldsymbol{D}\|_K, \boldsymbol{D} = (D_{S1}(\boldsymbol{y_1}), D_{S2}(\boldsymbol{y_2}), ..., D_{Sn}(\boldsymbol{y_n})) \\ &G_{D,j}(\boldsymbol{X}) = \frac{\partial D(\boldsymbol{X})}{\partial x_j} \approx \frac{\partial \|\boldsymbol{D}\|_K}{\partial x_j}, H_{D,jk}(\boldsymbol{X}) = \frac{\partial^2 D(\boldsymbol{X})}{\partial x_j \partial x_k} \approx \frac{\partial^2 \|\boldsymbol{D}\|_K}{\partial x_j \partial x_k}\end{aligned} \tag{9}$$

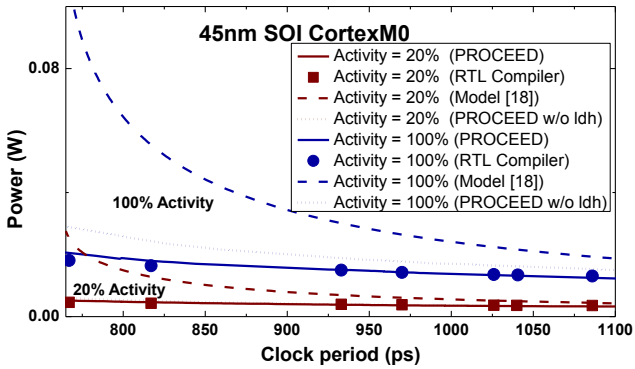

Fig. 7. Comparison between commercial synthesis tool, Model [4], and PROCEED. V_{dd} and V_t are constants and only size is a variable.

where K is the order of the norm. Higher K results in more accurate results (we use K = 100 in our simulations). Similarly, the elements of the gradient vector and Hessian matrix for power are given as

$$G_{P,j} = \frac{\partial P(\boldsymbol{X})}{\partial x_j} = \sum_{i=1}^{n} W_i \cdot \frac{\partial P_{Si}(\boldsymbol{y}_i)}{\partial x_j}, \mathrm{H}_{P,jk} \frac{\partial^2 P(\boldsymbol{X})}{\partial x_j \partial x_k} = \sum_{i=1}^{n} W_i \cdot \frac{\partial^2 P_{Si}(\boldsymbol{y}_i)}{\partial x_j \partial x_k} \quad (10)$$

(4) *Addition of new Pareto points*: To correct for model errors, circuit simulations are performed to evaluate D and P for all remaining potential Pareto points found by the optimization. In Fig. 6, this process is illustrated by the shift of the hatched point to the dotted circle. Finally, points not on the Pareto frontier (such that at least one other point with both lower delay and power exists) are filtered out.

(5) *Iteration termination*: For each iteration, when choosing the starting point for each step, the radius of trust region around this point is decreased by a factor of p (p > 1).Two termination conditions are applied: 1) existence of a sufficient Pareto point density in the region of interest, defined by the largest gap between any two neighboring points being smaller than a given criterion. This condition is usually used for devices with large operating regions (i.e. suitable for both high speed and low power applications). 2) Reduction of the radius of trust below a given criteria. This usually occurs due to limitations on the device operating region or device model discontinuities.

The PROCEED runtime is of order $O(r \times m^2)+O(r)$, where r is the resolution constraint (number of points in a unit Pareto curve), m is the total number of tuning parameters, $O(r \times m^2)$ is the complexity of the simulations for gradient and Hessian matrix calculation, and $O(r)$ is the complexity of simulating potential Pareto points. In our experiments, runtimes are mainly dominated by the resolution constraint; however, for large m, the $O(r \times m^2)$ term will dominate. The average PROCEED runtime to generate a full Pareto curve over three orders of magnitude in performance is about 4 hours on a single CPU. We use MATLAB in the optimization process and HSPICE for circuit simulations.

IV. Experiment Results

To illustrate PROCEED's capabilities, we compare it with existing evaluation methods and use it to assess SOI and silicon TFET devices at the 45 nm node. Because of their use of interband tunneling, TFETs are capable of very low leakage and extremely steep subthreshold swing, making them well-suited for low voltage operation [8]. Currently, however, nonidealities in experimental devices and low on-current limit their performance. We examine the viability of currently achievable TFETs using a device compact model [17]-[18] calibrated against TCAD simulations and experimental SOI devices [19]. While this does not represent the best possible TFET, which may require a different channel material or device structure, it have the advantages of being experimentally validated and structurally comparable to conventional SOI devices and represents a realistic lower bound. 45 nm SOI MOSFETs are modeled using commercial characteristics and compact model. Unless otherwise specified, all circuit results are generated with one V_{dd} and two V_t. To easily compare devices, we will refer to the Pareto crossover, defined as the delay above which the optimized novel device (here, the TFET) consumes less power than the established technology (SOI); lower Pareto crossover means the novel device is more promising for a given case.

A. Framework Evaluation

To validate our PROCEED framework, we use the widely employed evaluation model of Ref. [4] (hereafter Model [4]), and a commercial synthesis tool to evaluate the PD Pareto curve for a CortexM0 microprocessor with a commercial 45 nm SOI library and model. The information needed for PROCEED and Model [4] (LDH, average fan-out and interconnect load) is extracted from a synthesized, placed, and routed netlist at a clock period of 933 ps. Only one constant V_{dd} and one constant V_t are used, as Model [4] does not support multiple voltages and the commercial library has only constant supply and threshold voltages. As shown in Fig. 7, the PROCEED predictions are in much better agreement with the comprehensive optimized results from the RTL compiler compared to Model [4], which is frequently used for device evaluation [2]-[3]. The operating range for comparison is chosen by the synthesis results with the commercial library with one V_{dd} and V_t. We note that using the compiler for evaluation purposes is completely impracticable, since generating a Pareto curve from kHz to GHz speeds necessitates libraries with V_{dd} and V_t varying from 0.5V to 1.2V and 0.1V to 0.5V respectively. However, the generation and optimization of these libraries would consume months of runtime, whereas we completed the same study in hours using PROCEED. Meanwhile, the computationally simple Model [4] takes seconds to complete such Pareto curves but grossly overestimates power for two reasons: the neglect of LDH in assuming all gates have the same (large) size used for the critical path, and the use of analytical *PD* models rather than circuit simulations using full device characteristics. The dotted line is the Pareto curve generated by PROCEED while neglecting LDH, illustrating the accuracy improvement contributed by the two foregoing points. We further note that Model [4] cannot account for adaptivity, variability, or multiple V_{dd} and V_t effects. By benchmarking to the RTL results in Fig. 7, we observe *PROCEED improves accuracy by 3X to 115X compared to the current standard Model [4].*

B. Impact of Multiple V_{dd}, V_t, and Gate Sizing

More tuning parameters create a larger phase space for design optimization, as illustrated in Fig. 8 for a 45 nm SOI CortexM0 topology. As more LDH bin divisions are introduced, power is increasingly optimized because of a greater range of gate sizes with which to construct the design. Similarly, the introduction of additional supplies and threshold voltages substantially improves performance. The result does not account for the overhead consumed by the voltage shifter used in multiple V_{dd} design. Overall, however, we observe that *the evaluated optimal power at a given delay may change by over 50% as gate size tuning and multiple V_{dd} and V_t are introduced, demonstrating the necessity of including these effects in any quantitative comparison.*

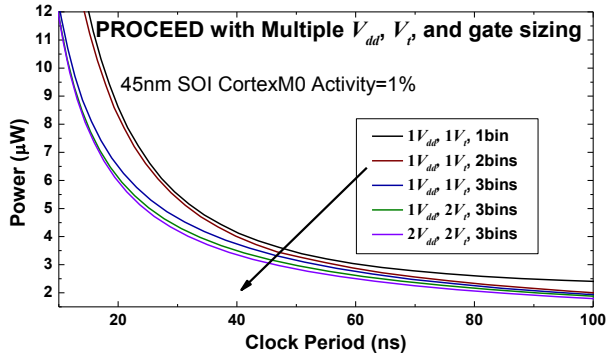

Fig. 8. 45nm SOI CortexM0 power-clock period as tuning parameters are increased.

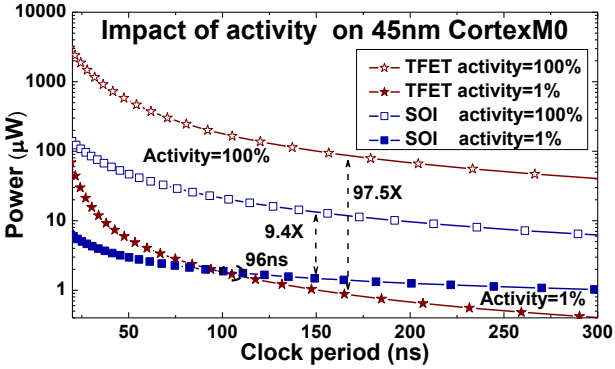

Fig. 10. Activity impact on 45nm Si SOI and Si TFET power-clock period.

C. *Impact of Benchmarks on Evaluation – SOI vs. TFET*

To show the impact of benchmark selection, we compare the performance of two microprocessors, CortexM0 and MIPS, using SOI and TFET devices and two supply rails and two threshold voltages. We choose these benchmarks because, as shown in Fig. 9(a), they have a similar number of critical path stages (56 in CortexM0 vs. 62 in MIPS) and total gates (8990 vs. 9248), but the CortexM0 has a more evenly distributed LDH. The power consumption in MIPS is dominated by short paths, which means it will be more accommodating of slow devices compared to the CortexM0. Accordingly, in Fig. 9(b), both SOI and TFET achieve better power efficiency in MIPS designs because the second V_{dd} and V_t can be optimized to save power along the short paths. The crossover points where the Pareto curves for different devices intersect define their advantageous operating regions; a device changes from being less power efficient on one side of the crossover to being more efficient on the other side. If multiple crossovers are found, then the Pareto curve can be divided into several regions (high performance, low power, etc.) such that in each one, there is only a single crossover point. This allows us to demarcate the (possibly multiple) favorable operating ranges for each device. The Pareto crossover occurs at 73 ns and 106 ns for MIPS and CortexM0, respectively, showing that TFETs are more acceptable for applications like MIPS which tolerate slower devices. However, TFET drive currents must be increased if they are to be usable at higher clock rates. Previous evaluations, like those in Table 1, which ignore LDH, are not able to distinguish between benchmarks in this way. *These results show how the choice of circuit topology strongly impacts the suitability of emerging devices.*

D. *Impact of Activity Factor – SOI vs. TFET*

We next examine how activity factor affects SOI- and TFET-based CortexM0 processors in Fig. 10. As activity reduces

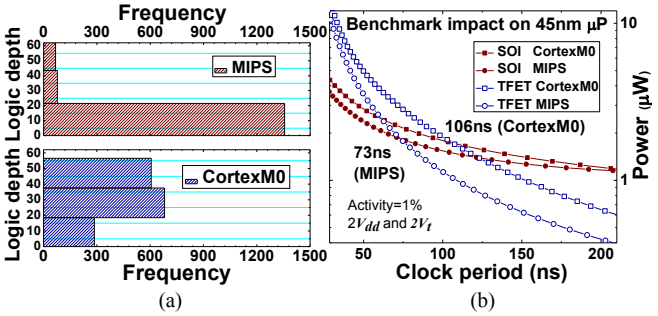

Fig. 9. (a) LDH of MIPS and CortexM0. (b) Power and delay curves for MIPS and CortexM0 designed with TFET and SOI respectively. Activity is 1% and two V_{dd} and two V_t are applied.

from 100% to 1%, TFET circuit power scales in lockstep by 97.6X due to low device leakage. However, the corresponding SOI designs only see power reduction of 9.4X because of its higher off-current. We see that *TFETs change from being completely impracticable at 100% activity to being superior to SOI beyond the 96 ns delay point at 1% activity; thus activity factor, and hence system use contexts, can drastically alter the device evaluation and must be considered.*

E. *Power Management Modeling*

The results of the previous subsections make clear that there is no panacea device and that device-circuit evaluation must be done with specific applications and operating windows in mind. DVFS and power gating are crucial ingredients for such usage-mindful evaluation. In Fig. 11, we show PROCEED- generated Pareto curves at different ratios of average to peak throughputs for SOI and TFET CortexM0 using DVFS and power gating. Power is reduced by operating at the lower supply rail or turned off by power gating; the achievable power reduction differs with device and operating region. The peak throughput crossover point for TFETs shifts from 10.9M to 21.5M operations per second as the ratio of average to peak throughput reduces from 100% to 10%; *the relative performance of TFETs effectively doubles as throughput requirements become less aggressive, emphasizing the importance of incorporating power management into device benchmarking.*

F. *Variation-Aware Evaluation*

To illustrate how variability might impact conclusions drawn using nominal devices, we show in Fig. 12 how the SOI and TFET Pareto curves are changed when slow corner devices are used. We define the slow corner as a device with 10% effective voltage

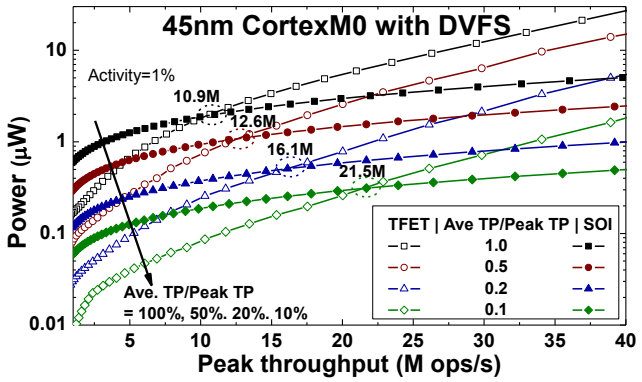

Fig. 11. 45nm SOI and TFET CortexM0 microprocessors with power management. The ratio of average to peak throughputs are 10%, 20%, 50% and 100%. Curves with ratios of 100% are designs outputted from Pareto optimizer.

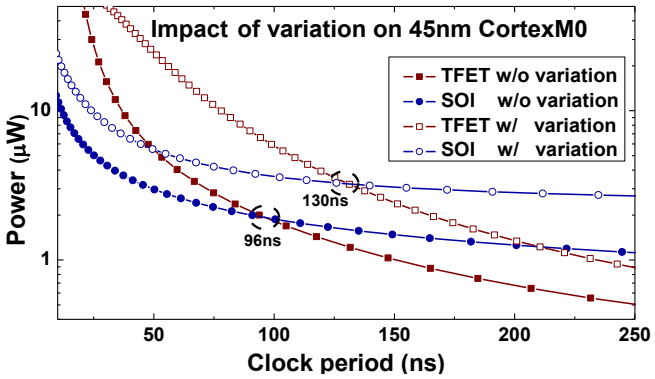

Fig. 12. Variation-aware evaluations of 45nm technologies. Assumed voltage drop is 90% and V_t shift is 50mV.

reduction and 50 mV V_t shift; total power is simulated using the nominal device, while delay is evaluated with the slow corner. We observe that the TFET is more sensitive to variability effects than SOI, as the Pareto crossover shifts from 96 ns to 130 ns. This is due to the TFET's steep subthreshold swing around the crossover, leading a high sensitivity of drive current to voltage [20]-[21] *This suggests that TFETs need to show substantial* ***nominal device*** *advantages in order to buffer this sensitivity and demonstrates that even a simple consideration of variability is important in device evaluation and selection.*

V. Conclusion

The proposed circuit-device co-evaluation framework [2] accounts for circuit topology, adaptivity, variability and use context using efficient Pareto optimization heuristic. Previous device evaluation frameworks ignore one or more crucial factors like multiple supply and threshold voltages, power management, logic depth, variability, etc., which can easily lead to misleading results. For instance, we find that including power management in our evaluation can effectively double the usable operating range for TFETs, and that choice of activity factor can dictate whether TFETs are acceptable at all in a given application. These observations are made possible by PROCEED's scope and computational efficiency in studying several orders of magnitude in possible device/circuit performance, and demonstrate the power and flexibility of our new methodology.

VI. Acknowledgement

We would like to acknowledge the generous support of IMPACT UC Discovery Grant (http://www.impact.berkeley.edu/) in accomplishing this work.

References

[1] L. Wei, S. Oh, and H.-S. P. Wong, "Performance benchmarks for Si, III–V, TFET, and carbon nanotube FET-re-thinking the technology assessment methodology for complementary logic applications," *Proc. IEDM*, pp.16.2.1-4, 2010.

[2] L. Wei and D. A. Antoniadis, "CMOS device design and optimization from a perspective of circuit-level energy-delay optimization," *Proc. IEDM*, pp.15.3.1-4, 2011.

[3] P. M. Solomon, D. J. Frank, and S. O. Koswatta, "Compact model and performance estimation for tunneling nanowire FET," *Proc. DRC*, pp.197-198, 2011.

[4] D. J. Frank, W. Haensch, G. Shahidi, and O. H. Dokumaci, "Optimizing CMOS technology for maximum performance," *IBM J. Research and Dev.*, vol. 50, no. 4/5, pp.419-431, Jul.-Sep. 2006.

[5] D. Sylvester and K. Keutzer, "System-Level Performance Modeling with BACPAC – Berkeley Advanced Chip Performance Calculator," *Proc. SLIP*, pp. 109-114, 1999.

[6] M. Luisier, M. Lundstrom, D. A. Antoniadis, and J. Bokor, "Ultimate device scaling: Intrinsic performance comparisons of carbon-based, InGaAs, and Si field-effect transistors for 5 nm gate length," *Proc. IEDM*, pp.11.2.1-4, 2011.

[7] H. Kam, T.-J. King-Liu, E. Alon, and M. Horowitz, "Circuit-level requirements for MOSFET-replacement devices," *Proc. IEDM*, pp.1.1. 15-17, 2008.

[8] C. Augustine, A. Raychowdhury, Y. Gao, M. Lundstrom, and K. Roy, "PETE: A device/circuit analysis framework for evaluation and comparison of charge based emerging devices," *Proc. ISQED*, pp.80-85, 2009.

[9] C. Pan and A. Naeemi, "System-Level Optimization and Benchmarking of Graphene PN Junction Logic System Based on Empirical CPI Model," *IEEE Proc. Intl. Conf. IC Design & Technology,* May. 2012.

[10] A. M. Ionescu and H. Reil, "Tunnel field-effect transistors as energy-efficient electronic switches," *Nature*, vol. 479, no. 7373 pp. 329-337, 2011.

[11] R. S. Ghaida and P. Gupta, "DRE: a Framework for Early Co-Evaluation of Design Rules, Technology Choices, and Layout Methodologies," *IEEE Trans. CAD*, vol. 31, no. 9, pp. 1379-1392, Sep. 2012.

[12] J. A. Davis, V. K. De, and J. D. Meindl, "A Stochastic Wire-Length Distribution for Gigascale Integration (GSI)—Part II: Applications to Clock Frequency, Power Dissipation, and Chip Size Estimation," *IEEE TED*, vol. 45, no. 3, pp. 590-597, Mar. 1998.

[13] J.-H. Ryu, S. Kim, and H. Wan, "Pareto front approximation with adaptive weighted sum method in multiobjective simulation optimization," *Proc. Winter Simulation Conference (WSC)*, pp.623-633, 2009.

[14] J. P. Benson *et al*., "Car-park management using wireless sensor networks." *Proc. Conf. Local Computer Networks*, pp. 588-595, 2006.

[15] T. He *et al*., "Achieving Real-time Target Tracking Using Wireless Sensor Networks," *Proc. RTAS Symp.*, pp. 37-48, 2006.

[16] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, 2004.

[17] A. Pan and C. O. Chui, "A Quasi-Analytical Model for Double-Gate Tunneling Field-Effect Transistors," *IEEE EDL,* vol. 33, no. 10, pp. 1468-1470, Oct. 2012.

[18] A. Pan, S. Chen, and C. O. Chui, "Electrostatic Modeling and Insights Regarding Multigate Lateral Tunneling Transistors," *IEEE TED*, vol. 60, no. 9, pp. 2712-2720, Sept. 2013.

[19] K. Jeon *et al*., "Si Tunnel Transistors with a Novel Silicided Source and 46 mV/dec Swing," *2010 VLSI Symp.*, p. 121-122, 2010.

[20] G. Leung and C. O. Chui, "Stochastic Variability in Silicon Double-Gate Lateral Tunnel Field-Effect Transistors," *IEEE Trans. Electron Devices*, vol. 60, no. 1, pp. 84-91, 2013.

[21] G. Leung and C. O. Chui, "Interactions between Line Edge Roughness and Random Dopant Fluctuation in Non-Planar Field-Effect Transistor Variability," *IEEE Trans. Electron Devices*, vol. 60, no. 10, pp. 3277-3284, 2013.

[2] PROCEED will be made publicly available as open-source software.