An Efficient Algorithm for Statistical Minimization of Total Power under Timing Yield Constraints

Murari Mani[1], Anirudh Devgan[2], and Michael Orshansky[1]
[1]University of Texas, Austin, [2]Magma Design Automation

ABSTRACT
Power minimization under variability is formulated as a rigorous statistical robust optimization program with a guarantee of power and timing yields. Both power and timing metrics are treated probabilistically. Power reduction is performed by simultaneous sizing and dual threshold voltage assignment. An extremely fast run-time is achieved by casting the problem as a second-order conic problem and solving it using efficient interior-point optimization methods. When compared to the deterministic optimization, the new algorithm, on average, reduces static power by 31% and total power by 17% without the loss of parametric yield. The run time on a variety of public and industrial benchmarks is 30X faster than other known statistical power minimization algorithms.

Categories and Subject Descriptors
B.6.3 [**Design Aids**]

General Terms
Algorithms, performance, design, reliability

Keywords
Leakage, manufacturability, statistical optimization

1. INTRODUCTION
The increase in variability of several key process parameters, such as transistor gate length and threshold voltage, significantly impacts the design and optimization of low-power circuits in the nanometer regime [1]. The growth of variability can be attributed to multiple factors, including the difficulty of manufacturing control, the emergence of new systematic variation-generating mechanisms, and most importantly, the increase in fundamental atomic-scale randomness, such as the variation in the number of dopants in the transistor channel [2].

The growth of standby, or leakage, power as device geometries scale down has become an extremely urgent issue. It is projected that at the 65nm node, leakage power will account for 45% of total power of the circuit [3]. This trend can be attributed primarily to the exponential dependence of leakage current on threshold voltage of the device. This exponential dependence also causes a large spread in leakage current in the presence of process variations. It has been demonstrated that a 1.3X variation in the effective channel length could potentially lead to 20X variation in leakage current [4].

DAC 2005, June 13–17, 2005, Anaheim, California, USA.

Low power designs are especially vulnerable as low V_{th} devices exhibit larger sensitivity to variation. On the other hand, high performance parts are vulnerable as they tend to have highest leakage power leading to large yield loss in the high performance bin [11].

Post-synthesis circuit optimization techniques, such as sizing and dual-V_{th} allocation, are effective in reducing leakage, and have been widely explored in a deterministic setting [5-7]. While relying on different implementation strategies, all of these techniques essentially trade the slack of non-critical paths for power reduction by either downsizing the transistors or gates or setting them to a higher V_{th}. In the past, case-files have been used with such optimization methods to guarantee that the circuit is optimized while guaranteeing a specific yield point. The rise of uncorrelated intra-chip variability [1] results in the breakdown of the case-file based approach to handling variability in optimization as it becomes impossible to come up with a case file that will guarantee a specific yield point. This requires the introduction of fully statistical optimization techniques that can handle the variance of objective and constraint functions explicitly during optimization. Given the exponential dependence of leakage power on the highly variable transistor channel length and threshold voltage, it can be expected that the introduction of rigorous statistical optimization will significantly reduce the leakage power consumption.

While much work has been done recently to develop statistical timing analysis methods [8], very little work has been done to account for variability in circuit optimization [9-11]. In [11], an iterative TILOS-like approach of [5], selecting the transistor to modify one at a time, is extended to rely on statistical sensitivities. Because of a heuristic problem formulation, it is not apparent how to control the required yield levels by adjusting the margin of the sensitivity variables. The algorithm also has a high run-time. In [9], a heuristic way of preventing a build-up of path delays near the critical path is proposed, but is not based on rigorous statistical formulation. Several statistical sizing algorithms have also appeared but are concerned with timing rather than power-limited yield [12-14].

In this paper we describe a new rigorous statistical algorithm for total power minimization. To our knowledge, this is the first attempt to solve the statistical leakage minimization problem using a theoretically rigorous formulation. It is also amenable to a *highly efficient computational implementation*. A two phase approach based on optimal delay budgeting and slack utilization, akin to [7], is used. The delay budgeting phase is formulated as a robust version of the power-weighted linear program that assigns slacks based on power-delay sensitivities of gates. We explicitly incorporate the notion of variability in delay and power due to process variations into the optimization, by setting an uncertain

robust linear program. The variance of delay and power, assumed to be due to channel length and threshold voltage variation, is mapped to the variance of the sensitivity vector. The statistical (robust) linear program is cast into a second order conic program that can be solved efficiently. The slack assignment is inter-leaved with the configuration selection which optimally redistributes slack to the gates in the circuit to minimize total power savings. Across the public and industrial benchmarks, the leakage and total power, when compared to a deterministic solution under the same timing performance, are on average, 31% and 17% lower respectively. The robust LP is solved using efficient interior-point methods, which are far superior to general non-linear solvers. As a result, the algorithm has extremely good run-time, providing a 30X speed-up compared to another known statistical leakage reduction algorithm [11].

The rest of the paper is organized as follows. In section 2, we present the power and delay models and introduce the deterministic slack assignment problem. The statistical optimization flow is described in detail in section 3. Section 4 presents the experimental results of running the algorithm on the benchmark circuits and section 5 presents the conclusion.

2. POWER MINIMIZATION BY LINEAR PROGRAMMING

The deterministic algorithm for power minimization is a two-phase iterative relaxation scheme. The input to the first phase is a circuit sized for maximum slack using a transistor (gate) sizing algorithm, such as TILOS [15], with all the devices set to low V_{th}. This circuit has the highest possible power consumption of any circuit realization. The available slack is then optimally distributed to the gates based on the power-delay sensitivities: that is, the slack is allocated in a way that maximizes the power reduction. The second phase consists of a local search among gate configurations in the library, such that slack assigned to gates in previous phase is utilized for power reduction.

The idea of using power-delay sensitivity of a circuit as an optimization criterion is itself well known [16]. A linear measure of gate's power-delay sensitivity is power reduction per unit of added delay:

(1) $$s = \partial P / \partial D$$

The power reduction for an added delay $d(i)$ is then given by $s(i)d(i)$. For example, a node driving a node with large fan-out will have a higher sensitivity than a small fan-out node. Thus, a unit of added slack to a node with a higher sensitivity will lead to the greater power reduction. We rely on extending this concept to efficient optimization based on large-scale linear programming by converting a power minimization problem into a power-weighted slack redistribution. Let a gate configuration be any valid assignment of sizes and threshold voltages to transistors in a gate in the library. For any fixed load, a set of Pareto points in the power-delay space can be identified among all the possible configurations (Figure 1). A power optimal solution will contain only the Pareto-optimal gate configurations. The trade-offs between delay and both leakage and dynamic power can be captured in tables, parameterized by the capacitive load. For each of the Pareto-optimal gate configurations, the decrease in power consumption (ΔP) and the change in delay (ΔD) are calculated. For example, we may compute the sensitivity of changing the gate from all transistors having low V_{th} to the configuration where all transistors have high V_{th}.

Using this framework, a linear program can be formulated to distribute slack to gates with the objective of maximizing total power reduction while satisfying the delay constraints on the circuit:

(2) $$\begin{aligned} & max \sum s_i d_i \\ & s.t.\ \ AT_i \geq AT_j + d_i^0 + d_i, \text{ for } \forall j \in FI(i) \\ & AT_o \leq RAT, \ \text{ for } \forall o \in PO,\ d_i \leq \delta d \end{aligned}$$

Here AT_i is the arrival time at node i, RAT is the required arrival time at the primary output, d_i^0 is the delay of the gate i in the circuit configuration obtained by sizing for maximum slack and d_i is the additional slack assigned.

The algorithm is constructed as an iterative-relaxation method. At its core is an interleaved sequence of (i) optimal slack-redistribution using LP, and (ii) the local search over the gate configuration space to identify a configuration that will absorb the assigned slack. Selection of optimal configurations is done independently for each gate. It has been shown that when the configuration space is continuous, and delay is a monotonic and separable function, such a procedure is optimal for small increments of slack assignments δd [17]. Also, the sensitivity vector is accurate within a narrow range of delay, requiring moving towards the solution under small slack increments. Even though the configuration space generated by V_{th} assignments is discrete, the ability to size transistors in a continuous manner permits treating as continuous. This ensures that a configuration exactly utilizing the slack allotted in the slack assignment phase can be found.

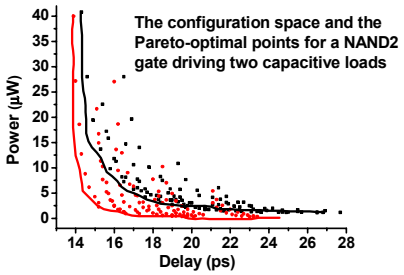

Figure 1: An example of a configuration space.

Deterministic power minimization
0: Size the circuit for maximum slack under all low Vth
1: Compute sensitivities for each gate
2: Solve linear program to optimally allocate slack
3: Find gate configurations to minimize power for given slack
4: Check timing
5: If circuit meets timing go to Step 1

Figure 2: The pseudo-code for the deterministic power minimization algorithm based on linear programming.

3. STATISTICAL DELAY AND POWER MODELS

In this work we are concerned with handling two primary sources of variability: effective channel length (L_{eff}) and gate-length independent variation of threshold voltage (V_{th}). From a physical point of view, this later source of variability will be primarily due

to random dopant fluctuation. These parameters have significant impact on timing (L_{eff}) and leakage power (V_{th}). An additive statistical model that decomposes the variability, of both L_{eff} and V_{th}, into the intra-chip and chip-to-chip variability components is used. For gate length,

$$L_{eff} = L_0 + \Delta L_{inter} + \Delta L_{intra}$$

Both L_{eff} and V_{th} are assumed to be Gaussian random variables, which is in agreement with empirical data [1]. The relative magnitudes of the intra- and inter-chip components can be controlled by adjusting their variances (${\sigma_L}^2 = {\sigma_{inter}}^2 + {\sigma_{intra}}^2$)

In keeping with the deterministic optimization algorithm, the statistical optimization will rely on the power-delay sensitivity vector. The impact of variability on delay and power is captured by statistically characterizing a standard cell library, in which two V_{th} levels and several discrete gate sizes are assumed to form the cell configuration space. The variance and covariance of the power-delay sensitivity coefficients are characterized statistically using Monte-Carlo simulation for all the cells in the library. The characterization provides the numerical values of the vector of mean sensitivities, $\overline{s}$, and the covariance matrix Σ of *s*. In order to formulate the statistical optimization problem rigorously, however, we need to establish some theoretical properties of the random sensitivity vector. We assume that a first-order Taylor expansion of the gate delay function is adequate:

$$d_g \cong d_g(L_o, V_{tho}) + (\partial d_g / \partial L)\Delta L + (\partial d_g / \partial V_{th})\Delta V_{th}$$

Under this model, the delay is Gaussian. Dynamic power consumption is very weakly dependent on the variation of V_{th} and L_{eff}, thus we ignore it. It was shown in [18], that an empirical leakage power model $P_{leak} = c_o e^{-c_1 L - c_2 V_{th}}$ with constants c_0, c_1 and c_2 can be used to accurately describe the variation in leakage power. Under this model, the leakage power, and hence, total power P is a log-normal random variable. The sensitivity coefficient can be written as $s = \partial P / \partial d$. Then, by chain rule *s* can be written as

$$s = \partial P / \partial d$$
$$= -c_0 e^{-c_1 L - c_2 V_{th}} \left(\frac{1}{c_1 (\partial d_g / \partial L)} + \frac{1}{c_2 (\partial d_g / \partial V_{th})} \right)$$

It can be seen from the above equation that, because *L* and V_{th} are normal random variables, *s* follows a log-normal distribution.

In the presence of non-zero inter-chip variability and spatial intra-chip variability, the sensitivity coefficients are correlated. Because the optimization is easier to set up when the sensitivities are uncorrelated, Principal Component Analysis (PCA) is used to transform the original vector of sensitivities into one with a diagonal covariance matrix. This transformation handles both sources of cell correlation. Given the covariance matrix Σ of the vector of sensitivities *s*, PCA obtains the vector of principal components s'. Then, the sensitivities are expressed in terms of their uncorrelated principal components [19]:

$$s = \overline{s} + As'$$

where, $\overline{s}$ is the vector of mean sensitivities and the matrix A is the eigenvector matrix of Σ.

4. STATISTICAL POWER OPTIMIZATION

In this section, a rigorous statistical equivalent for the power minimization strategy is described. To handle variability of process parameters, the problem is reformulated as a robust linear program and solved using efficient interior-point convex methods.

The essential contribution of this paper is the formulation of a rigorous statistical equivalent of the slack assignment using the notion of robust linear programming. Robust optimization is concerned with ensuring the feasibility and optimality of the solution under all permissible realizations of the coefficients of the objective and constraint functions [20]. A further contribution is an explicit incorporation of uncertainty in a formulation that is amenable to highly efficient computation.

When formulating a statistical power minimization problem, we find that an equivalent formulation of (2), which places the power weighted slack vector into the constraint set, is more convenient. Suppose that P_{max} is the initial maximum power, $\hat{P}$ is the optimal power achieved by (2) at a specific RAT, and $\hat{d}_1$ the vector of optimal allocated slacks. The following optimization problem (3) is equivalent to (2):

$$\begin{aligned} & min \sum d_i \\ & s.t. \ \sum s_i d_i \geq P_{max} - \hat{P} \\ & AT_o \leq RAT, \ \text{ for } \forall o \in PO \\ & AT_i \geq AT_j + d_i^0 + d_i, \ \text{for } \forall j \in FI(i) \end{aligned} \tag{3}$$

That is, if $\hat{d}_2$ denotes allocated slacks for (3), it can be shown that $\hat{d}_1 = \hat{d}_2$, and $P(\hat{d}_1) = P(\hat{d}_2)$ is a minimum power solution at the specified RAT. The reason is that (3) forces the LP to place a premium on the total slack and assign more slack to gates with higher sensitivity in order to meet the power constraint.

The statistical equivalent of (3) is now formulated by probabilistically treating the uncertainty of the sensitivity vector and of timing constraints:

$$\begin{aligned} & min \sum d_i \\ & s.t. \ P\left(\sum s_i d_i \geq P_{max} - P_{const} \} \right) \geq \eta \\ & P(AT_o \leq RAT) \geq \zeta \ \text{ for } \forall o \in PO \\ & AT_i \geq AT_j + d_i^0 + d_i, \ \text{for } \forall j \in FI(i) \end{aligned} \tag{4}$$

Here, the deterministic constraints have been transformed into the probabilistic constraints. These probabilistic constraints set respectively the power-limited parametric yield, η, and the timing-limited parametric yield, ζ. Based on the formulation of the model of uncertainty, they capture the uncertainty due to process parameters via the uncertainty of power and delay metrics. We now transform both probabilistic inequalities such that they can be efficiently handled by the available optimization methods. The challenge is to handle these inequalities analytically, in closed form.

We first consider timing constraints. The probabilistic timing constraints in (4) are now transformed such that the resulting expression still guarantees achieving the specified parametric yield level. Because of typically positive correlation between paths delays

$$P(D_i \leq t \mid D_j \leq t) \geq P(D_i \leq t)$$

Then, if we impose the constraint that $P(D_i \le RAT) \ge \zeta$ on every path, it ensures that the original timing constraint $P\left(AT_o \le RAT\right) \ge \zeta$ is met. This is the simplest approach. It is possible to apply a heuristic approach to adjusting the path-dependent coefficients, such that the conservatism is reduced. The probabilistic timing constraints can be represented by a *percent point function*:

$$D_i + \phi^{-1}(\zeta)\sigma_{D_i} \le RAT \tag{5}$$

where σ_{at_o} is the standard deviation of the i^{th} path at output *o*. In order to reduce the number of constraints and increase the sparsity of the constraint matrices, we further transform the path based constraints into node based constraints. In [21] it is shown that good results can be achieved by using a heuristic method of modeling the node delays with $d_i^0 + \phi^{-1}(\zeta)\sigma_{d_i^0}$, where $\sigma_{d_j^0}$ is the standard deviation of the gate delay d_i^0. This finally permits us to formulate the probabilistic timing constraint:

$$\begin{aligned} &AT_o \le RAT, \quad \text{for } \forall o \in PO \\ &AT_i \ge AT_j + d_i^0 + \phi^{-1}(\zeta)\sigma_{d_i^0} + d_i, \text{ for } \forall j \in FI(i) \end{aligned} \tag{6}$$

We now have to handle the probabilistic power constraint in (4). Letting $u = \sum s_i d_i = s^T d$, $\Delta P = P_{max} - P_{const}$, and $\eta' = 1 - \eta$, we can re-write the probabilistic constraint as $P(\ln u \le \ln \Delta P) \ge \eta'$. In section 3 we have shown that *u* can be modeled as a lognormal random variable. If $u \sim LN(m, \delta^2)$, then, $\ln u \sim N(\mu, \sigma^2)$. Now, if the mean of *u* is *m* and the standard deviation of *u* is δ, then,

$$\mu = \ln\left(m^2 / \sqrt{m^2 + \delta^2}\right), \ \sigma = \sqrt{\ln(1 + m^2/\delta^2)} \tag{7}$$

The translation-invariance property of a normal distribution can be used to express $P(\ln u \le \ln \Delta P) \ge \eta'$ as

$$P\left(\frac{\ln u - \mu}{\sigma} \le \frac{\ln \Delta P - \mu}{\sigma}\right) \ge \eta' \tag{8}$$

Since $(\ln u - \mu)/\sigma \sim N(0,1)$, letting $\phi(\cdot)$ be the *cdf* of $N(0,1)$, $P(\ln u \le \ln \Delta P) \ge \eta'$ is $\phi((\ln \Delta P - \mu)/\sigma) \ge \eta'$, and finally:

$$\mu + \phi^{-1}(\eta')\sigma \le \ln(\Delta P)$$

Using the above relationships between m and μ, and σ and s, we can finally express the probabilistic constraints as

$$\ln\left(m^2 / \sqrt{m^2 + \delta^2}\right) + \phi^{-1}(\eta')\sqrt{\ln(1 + m^2/\delta^2)} \le \ln \Delta P \tag{9}$$

The advantage of our formulation is the ability to take into account uncertainty of the constraint function explicitly. Indeed, the mean of *u* is $m = E(\underset{\sim}{s}^T d) = \overline{\underset{\sim}{s}}^T d$, and the variance is $\delta^2 = d^T \Sigma d$, where Σ is the covariance matrix of the vector of sensitivities *s*. Using the above non-linear probabilistic constraint, however, would require solving a non-linear optimization problem which is computationally expensive. However, we can reformulate this problem as a second-order conic program (SOCP) that can be solved efficiently. In general, an SOCP consists of minimizing a linear function over the convex set described by the intersection of an affine space with one or more second-order cones.

From (9) we can define:

$$f_0(m, \delta, k) = \ln(m^2 / \sqrt{m^2 + \delta^2}) + \phi^{-1}(\eta')\sqrt{\ln(1 + m^2/\delta^2)}$$

To formulate (4) as an SOCP, we need a percent point function which is linear in m and δ. Letting $k = \phi^{-1}(\eta')$, a least square of fit of f_0 onto the linear function f of the form can be performed:

$$f(m, \delta, k) = (a_1 + a_2 k)m + (a_3 + a_4 k)\delta$$

were $a,\ b,\ c,\ e$ are the fitting coefficients. This fit is justified as the *rms* error is ~5%. The constraint (9) can now be re-written as:

$$(a_1 + a_2 k)\overline{s}^T d + (a_3 + a_4 k)\sqrt{d^T \Sigma d} \le \ln \Delta P \tag{10}$$

Using (9), we can formulate the SOCP as:

$$\begin{aligned} &min \sum d_i \\ &s.t.\ (a_1 + a_2 k)\overline{s}^T d + (a_3 + a_4 k)\sqrt{d^T \Sigma d} \le \ln \Delta P \\ &AT_i \ge AT_j + d_i^0 + \phi^{-1}(\zeta)\sigma_{d_i^0} + d_i,\ AT_o \le RAT \end{aligned} \tag{11}$$

The optimization problem (11) has a special structure that can be exploited to result in very fast optimization. The reason is that the constraints in (11) are second-order conic functions that can be efficiently optimized by the interior point methods [20]. Because the second-order conic programs are convex [22], they guarantee a globally optimal solution. The reliance on interior-point methods means that the computational complexity of solving this non-linear program is close to $O(N^{1.3})$ in the size of the circuit. The second phase of the power minimization algorithm is $O(kN)$, where *k* is the number of alternatives in the gate configuration space. Thus, the overall complexity of our statistical power minimization algorithm is close to linear.

5. IMPLEMENTATION AND RESULTS

The algorithm was implemented in C as a pre-processing module to interface with a commercial conic solver available as part of MOSEK [23]. The benchmark circuits were synthesized to a cell library that was characterized for a 70 nm process using Berkeley Predictive Technology Model [24]. Gates have discrete sizes, ranging from 1x to 8x of minimum size. It is assumed that granularity of V_{th} allocation is at the NMOS/PMOS stack level. For NMOS (PMOS) transistors, the high threshold voltage is 0.20V (-0.20V) and the low threshold voltage is 0.10V (-0.10V). Different levels of variability in L_{eff} were explored ranging from 3% to 8% of σ/μ. It is assumed that σ_{Vth} of a gate is inversely proportional to its size, and gate-length independent V_{th} variation is due to random dopant placement. Pelgrom's model [25] is used to describe σ_{Vth} dependence on transistor size. The assumed magnitude of V_{th} variability is $\sigma/\mu = 7\%$. The mean and covariance matrix of cell sensitivities were computed for all gate configurations using SPICE. Principal component analysis was used to orthogonalize the covariance matrix of cell sensitivity coefficients. The performance and run-time behavior of the optimization algorithm is validated on the public ISCAS'85 benchmark circuits and several industrial blocks. All comparisons are done for the same arrival time at the primary output. This can be achieved by performing the deterministic power optimization using statistical timing constraints. Deterministic optimization under the 'worst-case' conditions is assumed to result in 100% yield. Across the benchmarks results indicate that the savings of, on average, 33% in leakage power without the loss of timing or power yield can achieved by statistical optimization as opposed to the deterministic approach, Table 1. The level of L_{eff} variability is assumed to be $\sigma/\mu = 8\%$. In the table, *n* is the number of gates

Table 1: Power savings obtained by deterministic and statistical optimizations at different yield levels.

		Timing yield ζ = 99.9%, Power yield η = 99.9%						Timing yield ζ = 84%, Power yield η = 99.9%						
	n	Deterministic Optimization		Statistical Optimization		Savings in Power (%)		Deterministic Optimization		Statistical Optimization		Savings in Power (%)		Run Time (s)
		Static	Total	Static	Total	Static	Total	Static	Total	Static	Total	Static	Total	
sc_ivlogic	40	29	140	19	111	35.2	20.8	19	113	12	97	33.3	14.8	9
sc_inc12	78	45	218	28	176	37.7	19.4	32	192	21	149	35.0	22.0	10
sc_edcs1	258	186	747	127	632	32.1	15.4	126	683	87	583	30.7	14.6	30
c432	261	157	858	107	696	32.3	18.9	112	783	75	620	32.8	20.8	31
c499	641	457	1290	305	1066	33.4	17.3	302	1054	213	894	29.6	15.2	52
c880	615	713	1217	492	1018	31.0	16.3	461	847	331	728	28.2	14.1	47
c1355	685	531	1501	343	1216	35.5	19.0	379	1240	244	994	35.6	19.8	56
c1908	1238	899	2559	611	2112	32.1	17.5	673	2284	503	1945	25.2	14.9	122
c2670	2041	1468	4814	1055	4113	28.1	14.6	1112	3926	813	3382	26.9	13.9	153
c3540	2582	1181	5549	809	4765	31.5	14.1	856	4498	602	3943	29.7	12.3	171
c5315	3753	2984	5411	1960	4493	34.3	17.0	2096	3769	1456	3222	30.5	14.5	241
c6288	2704	1178	5744	778	4691	34.0	18.3	746	4130	529	3429	29.1	17.0	273
Average savings						33.1	17.4					30.5	16.2	

in the circuit, and Static and Total refer to static and total power in μW respectively. Table 1 also documents the run-time behavior of the statistical optimization algorithm. For the largest benchmark the run-time is of the order of a few (~4) minutes. It compares very favorably with existing approaches, yielding a 30X speedup compared to [11]. It is pertinent to mention that the speedup is obtained due to the special structure of the SOCP program that is not available to the general non-linear solvers enabling the optimization problem to be solved extremely efficiently.

The fundamental reason for the reduction in power enabled by statistical optimization is the ability of the statistical algorithm to explicitly account for the variance of constraint and objective functions. This can be attributed to the fact that the statistical optimization allots slack more efficiently. One manifestation of the superiority of statistical optimization is the fact that it can assign more transistors to a high V_{th}. For example for the C432 benchmark optimized for a target delay of 0.55ns for 99.9% timing and power yields, the number of transistors set to high V_{th} by the statistical algorithm is 20% more than the corresponding number for the deterministic algorithm. As a result, the spread of the leakage distribution is reduced and the mean is shifted towards lower values. Figure 3 shows the *pdf* of static power obtained by a Monte Carlo simulation of the circuit configurations produced by the statistical and deterministic optimizations. Both the mean and variance of static power for the deterministically optimized circuit are greater, which implies that the static power savings increase at higher percentiles. The superiority of statistical optimization over the deterministic optimization is illustrated in Figure 4. Under the same power and timing yield constraints ($\zeta = \eta$ = 99.9%), statistical optimization produces uniformly better power-delay curves. The improvement strongly depends on the underlying structure of physical process variation. As the amount of uncorrelated variability increases, i.e. the intra-chip component grows in comparison with the chip-to-chip component, the power savings enabled by statistical optimization increase. The power savings at the 95th percentile are 23%, and those at 99th percentile are 27% respectively.

The ability to directly control the level of parametric power and timing limited yield permits choosing a 'sweet spot' in the power-delay space. Figures 5-6 show a set of power-delay curves for one of the benchmarks, c432. Figure 5 plots the total power vs. delay at the output obtained by running the statistical optimization for various timing yield levels (ζ), with the power yield set at 99.9%. It can be observed that at tight timing constraints the difference in power optimized for different yield levels is significant. Figure 6 confirms that optimizing the circuit for a lower power yield will lead to higher total power consumption and longer delay. For the same yield, the trade-off between power and arrival time is much more marked at tighter timing constraints. The raw magnitude of variability of physical parameters is clearly important to assessing the effectiveness of statistical optimization. If the variance of L_{eff} is reduced to $\sigma/\mu = 3\%$, the savings are smaller. Still, about 15% of savings in total power can be achieved at tighter timing constraints.

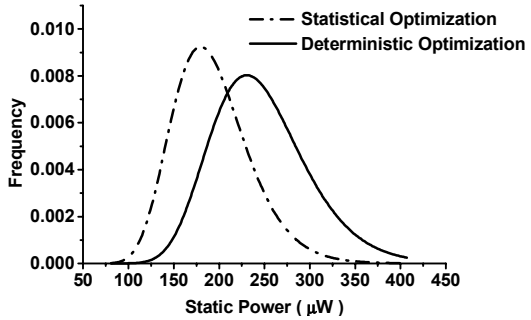

Figure 3. PDFs of static (leakage) power produced by a Monte-Carlo simulation of the benchmark circuit (C432) optimized by the deterministic and statistical algorithms.

6. CONCLUSION

In this paper we have presented a novel statistical algorithm for total power minimization that is based on a rigorous analytical formulation. We demonstrate that across the benchmarks our algorithm achieves significant reduction in static and total power. The algorithm also exhibits run-time that is substantially better than other known statistical algorithms.

7. ACKNOWLEDGEMENTS

This research was supported by SRC, GSRC, NSF, SUN, and University of Texas.

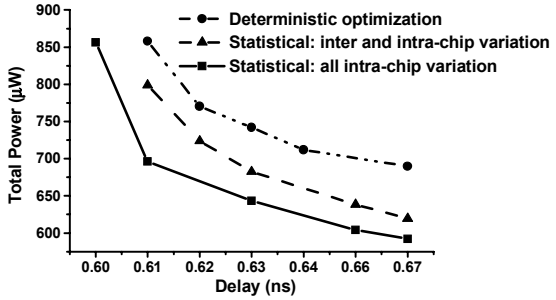

Figure 4: Power-delay curves for 99.9% timing and power yield. Statistical optimization does uniformly better. For the case of mixed inter- and intra-chip variability, an equal breakdown is assumed.

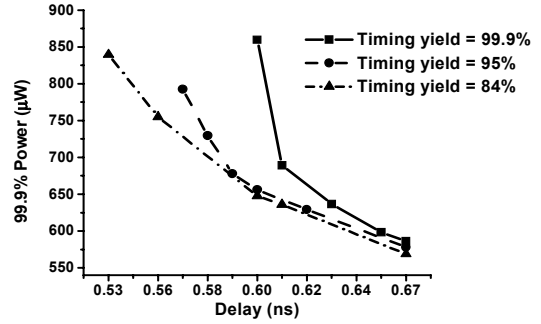

Figure 5: Power-delay curves at different timing yield levels for the C432 benchmark. At larger delay, the power penalty for higher yield is smaller.

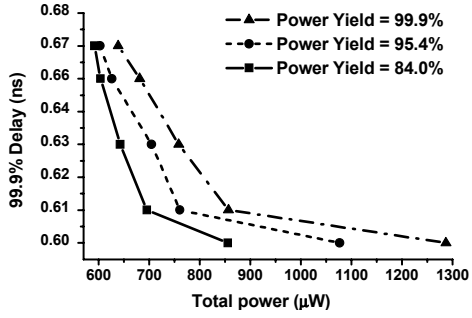

Figure 6: Power-delay curves at different power limited yields.

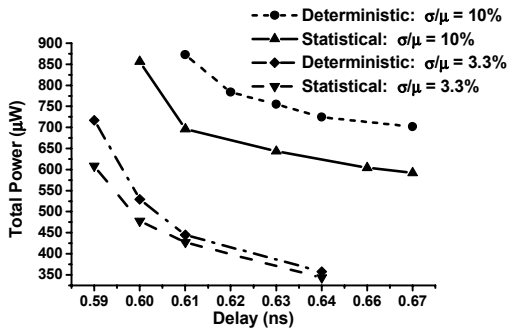

Figure 7: Power-delay curves for different levels of variability.

8. REFERENCES

[1] C. Visweswariah, "Death, taxes and failing chips," *Proc. of DAC* 2003, pp. 343-347.

[2] Y. Taur *et al.*, "CMOS scaling into the nanometer regime," *Proc. of the IEEE*, no. 4, 1997, pp. 486-504.

[3] R. Brodersen *et al.*, "Methods for True Power Minimization," *in Proc. of ICCAD*, 2002, pp. 35-40.

[4] S. Borkar *et al.*, "Parameter variation and impact on Circuits and Microarchitecture," *Proc. of D*AC, 2003, pp. 338-342.

[5] S. Sirichotiyakul *et al.*, "Stand-by power minimization through simultaneous threshold voltage selection and circuit sizing," *Proc. of D*AC, 1999, pp. 436-441.

[6] Q. Wang, and S. Vrudhula, "Static power optimization of deep submicron CMOS circuit for dual V_{th} technology," *Proc. of ICCAD*, 1998, pp. 490-496.

[7] D. Nguyen *et al., "*Minimization of dynamic and static power through joint assignment of threshold voltages and sizing optimization," *Proc. of ISLPED,* 2003, pp. 158 – 163.

[8] H. Chang and S. Sapatnekar, "Statistical timing analysis considering spatial correlations using a single PERT-like traversal," *Proc. of ICCAD, 2003*, pp. 621 – 625.

[9] X. Bai *et al.*, "Uncertainty aware circuit optimization," *Proc. of DAC*, 2002, pp. 58 – 63.

[10] S. Raj *et al.*, "A methodology to Improve Timing Yield," *Proc. of DAC*, 2004, pp. 448-453.

[11] A. Srivastava, D. Sylvester, and D. Blaauw, *"*Statistical optimization of leakage power considering process variations using dual-V_{th} and sizing," *Proc. of DAC*, June 7-11, 2004 pp. 773 – 778.

[12] E. Jacobs and M. Berkelaar, "Gate sizing using a statistical delay model," *Proc. of DAC*, 2000, pp. 283-290.

[13] P. Seung *et al.*, "Novel sizing algorithm for yield improvement under process variation in nanometer technology", *Proc. of DAC,* 2004, June 7-11, 2004, pp. 454 – 459.

[14] M. Mani and M. Orshansky, "A new statistical optimization algorithm for gate sizing," *Proc. of ICCD*, 2004, pp. 272 – 277.

[15] J. Fishburn and A. Dunlop, "TILOS: A Posynomial Programming Approach to Transistor Sizing," *Proc. of ICCAD*, 1985, pp. 326-328.

[16] D. Markovic *et al.*, "Methods for true energy-performance optimization," *J. of Solid-State Circuits*, 2004, pp. 1282- 1293.

[17] V. Sundararajan *et al.*, "Fast and Exact Transistor sizing Based on Iterative Relaxation," *IEEE Trans. on CAD*, vol. 21, 2002, pp.568-581.

[18] J. Kao *et al.*, "Subthreshold Leakage Modeling and Reduction Techniques," *Proc. of ICCAD*, 2002, pp. 141-149.

[19] C. Chatfield, *Introduction to Multivariate analysis*, Chapman and Hall, 1980.

[20] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge, 2004.

[21] Kim *et al.,* "A Heuristic for Optimizing Stochastic Activity Networks with Applications to Statistical Circuit Sizing", *preprint.*

[22] A. Prekopa, *Stochastic Programming*, Kluwer Academic, 1995

[23] http://www.mosek.com/documentation.html#manuals

[24] Y. Cao *et al.*, "New paradigm of predictive MOSFET and interconnect modeling for early circuit design," *Proc. of IEEE CICC*, 2000, pp. 201-204.

[25] M. Pelgrom *et al.,* Matching Properties of MOS Transistors*, IEEE Journal of Solid-State Circuits,* vol. 24, 1989, pp. 1433-1440.