Low-Power Fanout Optimization using Multi Threshold Voltages and Multi Channel Lengths

Behnam Amelifard, Farzan Fallah, *Member*, Massoud Pedram, *Fellow, IEEE*

***Abstract*— This paper addresses the problem of low-power fanout optimization for near-continuous size inverter libraries. It is demonstrated that because of neglecting short-circuit current, previous techniques proposed to optimize the area of a fanout tree may result in excessive power consumption. The paper describes how the problem of low-power fanout optimization can be reduced to inverter chain optimization problem and formulates the minimization of the total power-consumption of an inverter chain as a geometric program. Moreover, it describes an efficient method to minimize the total power consumption of a fanout tree by using multi channel length (multi-L_{Gate}) and multi threshold voltage (multi-V_t) techniques. To do this, the delay and different components of power dissipation (i.e., capacitive, short-circuit, and leakage) of an inverter are accurately modeled as posynomials for multi-L_{Gate} and multi-V_t technologies; therefore, the proposed problem formulation results in a convex mathematical program comprising of a posynomial objective function with posynomial inequality constraints. Experimental results show that the proposed technique can reduce the power consumption of the fanout trees by an average of 11.17% over SIS fanout optimization program.**

***Index Terms*—Low-power design, Logic synthesis, technology mapping, fanout optimization, multiple threshold voltage, multiple channel length[1]**

I. Introduction

VERY often in VLSI circuits, a signal needs to be distributed to several destinations under a required timing constraint at each destination. In practice, there may also be a limitation on the load that can be driven by the source signal. Fanout optimization is the problem of building an inverter tree topology between a source and some sinks and sizing the inverters so that the driving capacitance at the source is less than an upper bound and the timing constraints at sinks are met, while an objective function is minimized [1-3]. Different objective functions have been considered for the fanout optimization problem, such as minimizing area [3-5], minimizing power consumption [4, 6], and minimizing load on the source [7].

Unlike buffer insertion which is a back-end process and is performed after the global routing when the interconnect information is available, fanout optimization is performed during logic synthesis often interleaved with the technology mapping process in order to provide the global placer with accurate information about the number and sizes of the logic gates in the netlist.

The fanout optimization problem to achieve minimum area for libraries with discrete sizes has been proven to be NP-complete [8, 9]. However, it has been shown that using an inverter library with near-continuous sizes greatly simplifies the problem [10]. More precisely, the assumption of near-continuous library allows one to model the problem as a mathematical optimization problem with continuous variables and solve it efficiently. With utilizing a near-continuous library, the mapping of optimized continuous variables to discrete ones in the library results in a near optimal solution.

Several techniques have been proposed to address the fanout optimization problem using simplified delay models. In [11], for example, the delay of a single path has been minimized by assigning equal delay budgets to each buffer on the path. While it is known this approach minimizes the delay from the source to any sink, it does not necessarily result in an optimal solution in terms of other objective functions such as area or power dissipation. Reference [7] introduced two transformations, namely “merging” and “splitting”, used to convert any fanout tree to a set of inverter chains. It was shown that these transformations maintain the area, delay, and input capacitance. Using the transformation introduced in [7], reference [3] proposed a logical effort-based fanout optimizer for area which attempts to minimize the total buffer area under the required time and input capacitance constraints.

Although much research has been done to address fanout optimization problem, there is little work on low-power fanout optimization. More specifically, since both dynamic and leakage power dissipation of a fanout chain are proportional to its area, it has been widely accepted that power minimization of the fanout tree is equivalent to its area optimization [4, 6]. In this paper, however, we show that due to short-circuit power dissipation, minimizing area does not necessarily result in a minimized power dissipation solution. In particular, the solution obtained from an area optimized fanout tree may dissipate excessive

Preliminary version of this manuscript has been published in [1]. The changes made in this paper are as follows: (1) the delay and power modeling has been extended for multi channel length inverters and the inverters have been utilized in the fanout trees, (2) a separate section has been added to the paper to analytically evaluate the power consumption of minimum area fanout chains, (3) the proofs of some lemmas and theorems, which were omitted in [1], are provided in the current paper, (4) a more comprehensive set of experiments has been carried out and the proposed technique has been compared with SIS fanout optimization program by conducting SPICE simulations.

short-circuit power. We formulate the problem of minimizing the power dissipation of a fanout chain and show how to build a fanout tree out of these power-optimized chains. Additionally, to suppress the leakage power dissipation in a fanout tree, we use multi-L_{Gate} [12, 13] and multi-V_t techniques. In the presence of multi-L_{Gate} and multi-V_t options, we accurately model the delay and power dissipation of inverters as posynomials; therefore, our proposed problem formulation results in a convex mathematical program comprising of a posynomial objective function with posynomial inequality constraints. Note that using multi- V_t and multi- L_{Gate} is not just a mathematical exercise. Indeed there are some commercial CAD tools which are based on using multi- V_t and multi-L_{Gate} for reducing leakage power consumption of a VLSI circuit. One example is the Blaze-DFM MO tool [14] which has been used for tape-out of some chips. One report of using this tool for reducing power consumption of a Qualcomm chip may be found at [15]. It is worth mentioning that in multi-L_{Gate} technique, typically a limited and discrete number of L values are chosen for leakage reduction. However, unlike the multi-V_t technique, the number of discrete length values in not limited to two or three. This is due to the fact that to achieve a new V_t, a new mask is needed which adds the total manufacturing cost of the circuit, while different channel lengths can be created by simply changing the geometry of the device and using only one mask. Using discrete values for the channel length, however, is needed for mask production. That is why in our results, after optimally sizing the channel lengths, we round the channel lengths to the nearest 1nm. Such a resolution has also been used in [12, 13].

When there is only one sink, the fanout tree is reduced to a chain of inverters between the source and sink and the fanout optimization problem becomes that of finding the number and sizes of the inverters to satisfy the input capacitance and timing constraints while minimizing some objective function such as area or power dissipation. For multiple sinks, on the other hand, by using the split and merge transformations [7] or by limiting the types of the fanout trees to the so called LT-trees [8], a fanout tree can be constructed from the inverter chains. In this paper we use *fanout chain* to describe the fanout topology with one sink and *fanout tree* to describe it when there are multiple sinks.

The remainder of the paper is organized as follows. Section II describes logical effort technique and its extension for handling multi-V_t and multi-L_{Gate} circuits. It further describes the power model that will be used throughout the paper. Section III investigates the problem of minimizing the area of a fanout chain and shows that a minimized area fanout chain may dissipate excessive short circuit power. Section IV formulates the problem of low-power fanout chain optimization (i.e., when there is only one sink) and shows how to optimize the power consumption of the fanout chain by utilizing multi-V_t and multi-L_{Gate} techniques. Section V shows how a low-power fanout tree can be constructed from the fanout chains. Simulation results and conclusions are given in Sections VI and VII, respectively.

II. Delay and power Models

A. The Delay Model

The delay model we use in this paper is based on logical effort [11]. The logical effort is a technique for modeling and analyzing delay in CMOS circuits and has been widely used to solve a variety of synthesis problems including technology mapping [16, 17], gate sizing [18], and fanout optimization [3, 6, 7]. Additionally, it has also been incorporated in some industry synthesis tools [19, 20]. Although the accuracy of logical effort delay model is reduced for deep-submicron devices, the main advantage of this technique is that it is very simple, quite efficient, and exhibits high fidelity as far as the gate propagations delays are concerned. Therefore, it has found broad applications in the early design stages, when the interconnect information is not available. By using this technique, the initial sizing of logic gates can be performed and the results provided to a global placer. After doing the placement/routing and extracting interconnect information, more accurate models, e.g., non-linear delay models or lookup tables, may be used for delay analysis and resizing of the gates if needed. In this section we first review this model and then describe its extension to handle multi-V_t and multi-L_{Gate} techniques.

Using the notion of logical effort, the delay of a gate with input capacitance C_{in}, which drives the load capacitance C_L, is modeled as,

$$D = \tau_0(p + gh) \tag{1}$$

where τ_0 is a conversion coefficient that characterizes the semiconductor process being used and converts the unit-less part, $p + gh$, to a time unit. For the sake of simplicity, in the remainder of this paper, we set τ_0 to one. Parameter p denotes the parasitic delay of the gate. The major contributor to the parasitic delay is the capacitance of the source/drain regions of the transistors that drive the output. Parameter g denotes the "logical effort" of the gate which depends only on the topology of the gate and its relative ability to produce output current. More precisely, the logical effort of a gate shows how worse it is at producing output current than an inverter if each of its inputs has the same input capacitance as the inverter. Finally, parameter h denotes the "electrical effort" of the gate and is defined as the ratio of the output capacitance of the gate to its input capacitance, i.e., $h = C_L / C_{in}$. The electrical effort describes how the electrical environment of the logic gate affects performance and how the size of the transistors in the gate determines its load-driving capability.

For an inverter, the value of logical effort g equals one and can be shown that p is the ratio of output diffusion

capacitance to input gate capacitance of the template inverter, denoted by $p_0 = C_{diff,T} / C_{in,T}$. Notice that since both input gate and diffusion capacitances of an inverter are scaled linearly by changing the inverter's size, for a scaled inverter, the ratio of diffusion-to-gate capacitance remains constant, i.e.,

$$C_{diff} / C_{in} = p_0 \quad (2)$$

where C_{diff} is the diffusion capacitance at the output and C_{in} is the gate capacitance at the input. In the following, we show how to extend the concept of logical effort to handle multi-V_t and multi-L_{Gate} technologies.

It is known that when the threshold voltage of a gate is changed, the new delay can be obtained from the alpha-power law [21] by the following equation,

$$d = d_0 \frac{(V_{dd} - V_{t0})^{\alpha}}{(V_{dd} - V_t)^{\alpha}} \quad (3)$$

where α is a technology parameter which is around 2 for long channel devices and 1.3 for short channel devices, V_{dd} is the supply voltage, V_{t0} is the nominal threshold voltage, d_0 is the delay under the nominal threshold voltage, V_t is an arbitrary threshold voltage, and d is the delay under the arbitrary threshold voltage. Using equations (1) and (3) one can verify that in a multi-V_t technology, the values of the logical effort and parasitic delay change as follows,

$$g_v = \frac{(V_{dd} - V_{t0})^{\alpha}}{(V_{dd} - v)^{\alpha}},\ p_v = p_0 \frac{(V_{dd} - V_{t0})^{\alpha}}{(V_{dd} - v)^{\alpha}} \quad (4)$$

where g_v and p_v are the logical effort and parasitic delay for an arbitrary threshold voltage, v.

Equations (1) and (4) are based on the assumption that the channel length of the gate, L, is equal to the nominal channel length of the technology, L_{nom}. In a multi-L_{Gate} technology, however, the delay of a logic gate is an increasing function of the channel length. Our SPICE simulations show when the channel length of an inverter is increased, the new delay can be obtained from the following equation,

$$d_l = d_0 l^{\beta_d} \quad (5)$$

where l is the normalized channel length, i.e., $l = L_{Gate} / L_{nom}$ and β_d is a fitting parameter. Moreover, d_0 is the delay under the nominal channel length, while d_l is the delay of the gate with the normalized channel length l. Fig. 1 demonstrates the validity of this delay model. Using equation (5), one can easily establish that in a multi-L_{Gate} technology, values of the logical effort and parasitic delay change as follows,

$$g_l = l^{\beta_d},\ p_l = p_0 l^{\beta_d} \quad (6)$$

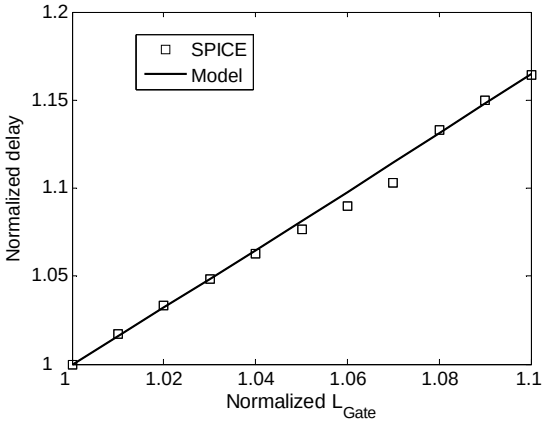

Fig. 1. Delay as a function of channel-length.

B. Power Dissipation Model

The power dissipation of a CMOS gate has three components: capacitive power, short circuit power, and leakage power.

1) Capacitive Power Dissipation

The capacitive power dissipated in inverter capacitances, i.e., input gate capacitance and output diffusion capacitance, is equal to,

$$P_{dyn} = \alpha f V_{dd}^2 C \quad (7)$$

where α is the switching activity of the inverter, f is the frequency, V_{dd} is the supply voltage, and C is the sum of the input gate capacitance and output diffusion capacitance of the inverter, i.e., $C = C_{diff} + C_{in}$. By using (2), equation (7) can be re-written as,

$$P_{dyn} = \alpha f V_{dd}^2 (1 + p_0) C_{in} = k_{dyn} C_{in} \quad (8)$$

In a multi-L_{Gate} technology, the input gate capacitance of the inverter increases as a result of biasing the channel length, while the diffusion capacitance remains unchanged. Therefore, the capacitive power dissipation is obtained from,

$$P_{dyn,l} = k_{dyn} \frac{l + p_0}{1 + p_0} C_{in} \quad (9)$$

where C_{in} denotes the input capacitance of the inverter under nominal gate-length.

2) Short-Circuit Power Dissipation

The second source of power dissipation in digital circuits is short-circuit current. If a circuit is *well-designed*, its short-circuit power dissipation is about 10%-20% of the capacitive power dissipation [22]. If the supply voltage of the inverter is lowered to be below the sum of the absolute values of transistors' threshold voltages ($V_{t,n} + |V_{t,p}|$), the short-circuit current can be eliminated, because both devices cannot conduct simultaneously for any value of the inverter input voltage [23]. However, such a low supply voltage is not compatible with static (fully complementary) CMOS logic design style, which has been the building block of VLSI circuits for decades. Therefore, in this paper, we do not consider operation at power supply voltages

lower than 2 or 3 times $V_{t,n} \approx |V_{t,p}|$ so as to meet the DC noise margins for standard CMOS design.

Several techniques have been proposed to address the problem of short circuit power estimation [22], but due to their complexity, their use tend to be impractical during gate-level optimization. In this paper, by observing the fact that short-circuit power dissipation of an inverter is a linear function of its size and input transition time [22] and also the fact that input transition time itself can be approximated as a linear function of the electrical effort of its fanin gate (see Fig. 2), the short-circuit power dissipation of the i^{th} inverter in a chain is calculated as,

$$P_{sc} = \alpha A_{sc} h_{i-1} f V_{dd} C_{in} = k_{sc} h_{i-1} C_{in} \qquad (10)$$

where A_{sc} is the short-circuit factor which is a technology-dependent parameter, h_{i-1} is the electrical effort of the $(i-1)^{\text{th}}$ inverter and C_{in} is the input capacitance of the i^{th} inverter. From Fig. 2 one can see that this technique, despite its simplicity, is accurate enough to be used in gate-level optimization.

From equations (8) and (10), one can see the ratio of the short-circuit to the dynamic power dissipation of an inverter can be expressed as,

$$\frac{P_{sc}}{P_{dyn}} = \frac{k_{sc}}{k_{dyn}} h_{i-1}. \qquad (11)$$

For various values of h_{i-1} this ratio is plotted in Fig. 2.

It should be noted that in a multi-V_t inverter chain, the short-circuit power dissipation, and consequently, k_{sc} of the i^{th} inverter (henceforth, denoted as $k_{sc,i}$) is a function of the threshold voltages of the i^{th} inverter and its driver (i.e., the $(i-1)^{\text{th}}$ inverter). If there are m threshold voltages in the library, then there will be m^2 distinct values for $k_{sc,i}$'s.

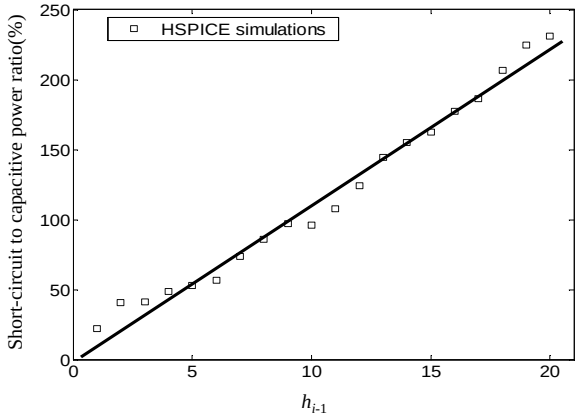

Fig. 2. The percentage ratio of the short-circuit power dissipation of the i^{th} inverter to its dynamic power dissipation, as a function of h_{i-1}.

Utilizing longer channel length for PMOS and NMOS transistors in a CMOS inverter increases the threshold voltage of both transistors; therefore, the time during which both NMOS and PMOS transistors are ON during the output transition is decreased. Thus, the short-circuit power

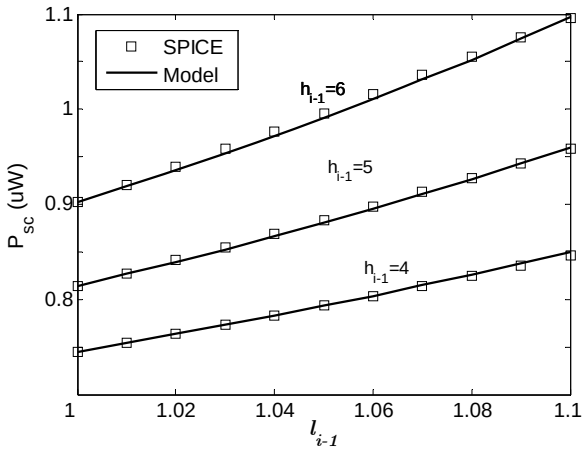

Fig. 3. Short-circuit power dissipation as a function of driver channel length.

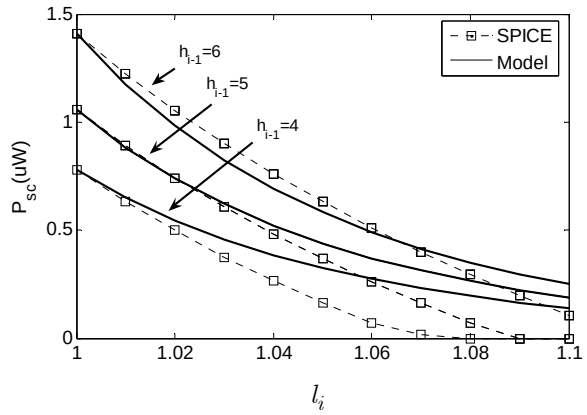

Fig. 4. Short-circuit power dissipation as a function of channel length.

consumption of the inverter is reduced. On the other hand, since the output slew time of an inverter increases when using a longer channel length, the short circuit power of the fanout gate increases. Therefore, in an inverter chain, the short-circuit power dissipation of the i^{th} inverter is inversely proportional to the channel length of the inverter, i.e., l_i, and directly proportional to the channel length of its driver, i.e., l_{i-1}. Based on these observations, we model the short-circuit power dissipation of the i^{th} inverter in a chain as,

$$P_{sc} = k_{sc} h_{i-1} l_i^{-\beta_{sc1}} l_{i-1}^{\beta_{sc2}} C_{in} \qquad (12)$$

where β_{sc1} and β_{sc2} are technology constants found by fitting (12) to data extracted from SPICE level simulations. Fig. 3 and Fig. 4 compare (12) with the actual SPICE data for various values of l_{i-1} and l_i. It should be mentioned that although the accuracy of the model is reduced for large l_i's, since for these values of l_i the short-circuit power dissipation becomes quite small compared to the capacitive power, the error in the total power consumption model remains small (the maximum error is less than 11%).

3) Leakage Power Dissipation

The third source of the power dissipation is the leakage current. In the present CMOS technologies, the major components of the leakage current are sub-threshold and

gate-tunneling currents [24]. The sub-threshold leakage is the drain-source current of a transistor operating in the weak inversion region which can be expressed as [24],

$$I_{sub} = A_{sub}\mu_0 C_{ox}\left(\frac{w}{L_{eff}}\right)\exp\left(\frac{q}{n'kT}(V_{gs} - V_{t0} - \gamma' V_{sb} + \eta V_{ds})\right) \times (1 - \exp(-qV_{ds}/kT)) \quad (13)$$

where $A_{sub} = (kT/q)^2 \exp(1.8)$, μ_0 is the zero bias mobility, C_{ox} is the gate oxide capacitance per unit area, w and L_{eff} denote the width and effective length of the transistor, k is the Boltzmann constant, T is the absolute temperature, and q is the electrical charge of an electron. In addition, V_{t0} is the zero biased threshold voltage, γ' is the linearized body-effect coefficient, η denotes the Drain-Induced Barrier Lowering (DIBL) coefficient, and n' is the sub-threshold swing coefficient of the transistor.

Let C_N denote the input capacitance of an NMOS transistor. Since V_{ds} of the OFF transistor is V_{dd} which is more than a few $kT/q \approx 26mV$ and noting that in an NMOS transistor $w_N = C_N/(L_{eff}C_{ox})$, the sub-threshold leakage power of an NMOS transistor can be written as,

$$P_{sub,N} = A'_{sub}\, C_N \mu_N e^{-\lambda V_{t0,n}} \quad (14)$$

where $\lambda = q/n'kT$ and $A'_{sub} = A_{sub}V_{dd}/L_{eff}^2 \exp(\lambda\eta V_{dd})$ are technology constants. A similar formula can be derived for the sub-threshold leakage power of a PMOS transistor. From the sub-threshold leakage power expressions for the NMOS and PMOS transistors, the sub-threshold leakage power dissipation of an inverter, P_{sub}, can be written as,

$$P_{sub} = \rho P_{sub,P} + (1-\rho)P_{sub,N} \quad (15)$$

where ρ is the probability that the input of the inverter is at logic 1. If the ratio of the width of the PMOS transistor to that of the NMOS transistor is γ, i.e., $w_P/w_N = \gamma$, by considering the fact that for an inverter $C_{in} = C_N + C_P$, (15) can be re-written as,

$$P_{sub} = \frac{A'_{sub}}{1+\gamma}\left(\rho\gamma\mu_P e^{-\lambda V_{t0,p}} + (1-\rho)\mu_N e^{-\lambda V_{t0,n}}\right)C_{in} = k_{sub}C_{in} \quad (16)$$

From (16) one can see increasing the threshold voltage results in an exponential decrease in sub-threshold leakage current. Based on this observation, multi-V_t and gate-length biasing techniques have been proposed to reduce the leakage power dissipation. Without losing generality, we assume the threshold voltage of the NMOS and PMOS transistors are equal. In this case, when the threshold voltage of an inverter is changed to v, the new sub-threshold leakage power consumption is obtained as,

$$P_{sub,h} = k_{sub}\exp(-\lambda(v - V_{t0}))C_{in} = k_{sub,h}C_{in} \quad (17)$$

Utilizing a longer channel length for an inverter increases the threshold voltage of both PMOS and NMOS transistors, which in turn reduces the sub-threshold leakage. Based on these observations, we model the sub-threshold power dissipation of the i^{th} inverter in an inverter chain as,

$$P_{sub,l} = k_{sub}l^{-\beta_{sub}}C_{in} \quad (18)$$

where β_{sub} is a technology constant. As one can see from Fig. 5, despite its simplicity, this model is quite accurate.

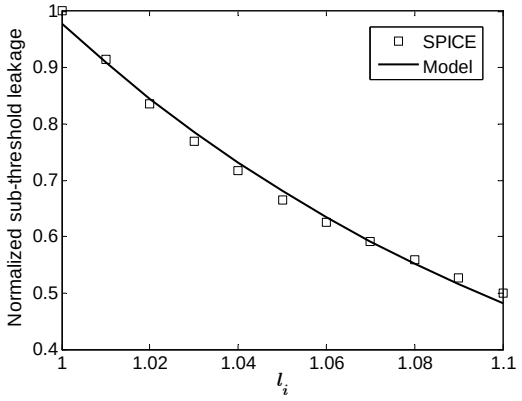

Fig. 5. Sub-threshold power dissipation as a function of channel length.

The other major source of the leakage power dissipation is the gate-oxide tunneling current. If SiO_2 is used for the gate oxide, the main source of gate-oxide tunneling leakage in CMOS circuits is the gate-to-channel tunneling current of the ON NMOS transistors, which can be modeled as [24, 25],

$$I_{ox} = A_{ox}w_N L_{eff}\left(\frac{V_{ox}}{t_{ox}}\right)^2 e^{-B_{ox}\frac{t_{ox}}{V_{ox}}} \quad (19)$$

where A_{ox} and B_{ox} are technology constants, t_{ox} is the oxide thickness, and V_{ox} is the potential drop across the oxide. When the transistor is ON, $V_{ox} = V_{gs} - \psi_s$, where ψ_s is the surface potential of the transistor. Ignoring the gate-tunneling leakage of the PMOS transistor, the gate tunneling leakage power dissipation of an inverter, P_{ox}, can be calculated by,

$$P_{ox} = \frac{A'_{ox}}{1+\gamma}\rho C_{in} = k_{ox}C_{in} \quad (20)$$

where $A'_{ox} = A_{ox}V_{dd}(V_{dd} - \psi_s)^2 \exp(-B_{ox}t_{ox}/(V_{dd}\psi_s))/(t_{ox}\varepsilon_0\varepsilon_{ox})$ is independent of the size and the threshold voltage of the inverter. From (19) one can see that the gate-oxide tunneling leakage is proportional to the area of the gate; therefore, in a multi-L_{Gate} technology, (20) should be modified as,

$$P_{ox,l} = k_{ox}lC_{in} \quad (21)$$

III. Minimum Area Fanout Chain

In minimizing the area of a fanout chain, shown in Fig. 6, the goal is to find the number of inverters in the chain and their corresponding sizes so that the delay constraint for the sink and the load capacitance constraint for the source are satisfied, while the total area of the chain is minimized:

$$\begin{cases} Min & Area \\ s.t. & (i)\ Delay \le T \\ & (ii)\ C_1 \le C_{in,\max} \end{cases} \quad (22)$$

where T is the required time at the sink, C_1 is the input capacitance of the first inverter and $C_{in,\max}$ is the maximum tolerable load at the source.

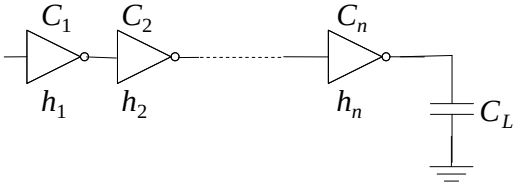

Fig. 6. A fanout chain driving a lumped capacitance.

In [3], based on the fact that the area of an inverter chain is proportional to the sum of input capacitance of the inverters in the chain and noticing that in an inverter chain with *n* inverters, the input capacitance of the i^{th} inverter can be expressed as $C_i = C_L / \prod_{j=i}^{n} h_j$, it is shown that the problem of minimizing the area of the chain with n inverters can be formulated in the logical effort notion as,

$$\begin{cases} Min & Area(\vec{h}) = \sum_{i=1}^{n} \dfrac{C_L}{\prod_{j=i}^{n} h_j} \\ s.t. & (i) \quad \sum_{i=1}^{n} p_0 + h_i \le T \\ & (ii) \quad H = \prod_{i=1}^{n} h_i \ge \dfrac{C_L}{C_{in,\max}} \end{cases} \tag{23}$$

where C_L is the load capacitance and $\vec{h} = (h_1,...,h_n)$.

Problem stated in (23) is called the Fanout Chain Optimization for Area with n inverters, $FCOA(n)$. The minimized area fanout chain can be found by solving $FCOA(n)$ for different values of n. However, depending on the polarity of the sink, only even or odd values for n should be considered. On the other hand, it can be shown that [3] for a fixed number of inverters in the chain (i.e., a fixed n), (23) will have a solution when $n\left(C_L / C_{in,\max}\right)^{1/n} + np_0 \le T$. This inequality defines a lower bound and an upper bound for the values of n satisfying the constraints of (23) and limits the number of $FCOA(n)$ instances needed to be solved to find the minimum area fanout chain [3].

Lemma 1: In the optimum solution of $FCOA(n)$, the delay of the fanout chain is exactly equal to the required time T, i.e., [3]

$$\sum_{i=1}^{n} p_0 + h_i = T \,. \tag{24}$$

A. Convex Representation

In the following, we show one important property of $FCOA(n)$ which guarantees the problem of minimizing area of a fanout chain has an optimal polynomial-time solution. More precisely, we show with a slight modification, the problem shown in (23) is converted to a convex program. A convex optimization problem is one of the form [26],

$$\begin{cases} Min & f_0(\vec{x}) \\ s.t. & f_i(\vec{x}) \le b_i, \qquad i = 1,...,m \end{cases} \tag{25}$$

where the functions $f_0,...,f_m : \Re^n \to \Re$ are convex, $b_1,...,b_m$ are some positive real numbers, and $\vec{x} = (x_1,...,x_n)$ is a vector. One important property of convex optimization problem is that a local optimal solution is also the global optimum solution.

Lemma 2: Function f defined as $f(\vec{x}) = 1/\prod_{i=1}^{m} x_i$ is convex on $dom(f) = \Re_{++}$.

Proof: We use the fact that f is convex if and only if its domain is convex and its Hessian is positive semi-definite [26], i.e., for all x belonging to $dom(f)$, $\nabla^2 f \ge 0$. One can see that,

$$\nabla^2 f(\vec{x}) = \frac{1}{\prod_{i=1}^{m} x_i}\left(\operatorname{diag}\left(1/x_1^2,...,1/x_m^2\right) + zz^T\right) \tag{26}$$

where z is a vector such that $z_i = 1/x_i$ and $\operatorname{diag}(.)$ is a diagonal matrix. To verify $\nabla^2 f \ge 0$ we should show that for any vector u,

$$u^T \nabla^2 f(\vec{x}) u \ge 0 \tag{27}$$

However, it can be verified that,

$$u^T \nabla^2 f(\vec{x}) u = \frac{1}{\prod_{i=1}^{m} x_i}\left(\sum_{i=1}^{m} (u_i/x_i)^2 + \left(\sum_{i=1}^{m} u_i/x_i\right)^2\right) \ge 0. \tag{28}$$

Therefore, f is convex. ■

Theorem 1: By changing the second constraint of $FCOA(n)$ as

$$\frac{1}{\prod_{i=1}^{n} h_i} \le \frac{C_{in,\max}}{C_L} \tag{29}$$

$FCOA(n)$ becomes a convex optimization problem for all values of n.

Proof: According to Lemma 2 the objective function of $FCOA(n)$ is a summation of convex functions and because the summation operation preserves the convexity property [26], the objective function of the problem given by (23) is convex. On the other hand, the first constraint of (23) is a linear function of h_i's; hence, it is convex. The function $f(\vec{x}) = \prod_{i=1}^{n} x_i$ is neither convex nor concave [26]. However, according to Lemma 2, by re-writing it as (29) it becomes convex. Since the objective function and constraints of (23) are convex on $\Re_{++}$, the mathematical problem stated in (23) is convex. ■

Since $FCOA(n)$ is a convex program, it can be efficiently solved by using standard mathematical program solvers.

B. Minimum Area versus Minimum Power Fanout Chain

Since both dynamic and leakage power dissipation of a fanout chain are proportional to its area, it has been widely

accepted that power minimization of a fanout chain is equivalent to its area optimization [4, 6]. In the following, however, we show that due to short-circuit power dissipation, minimizing area does not necessarily result in a minimized power dissipation solution and the solution obtained from an area optimization technique may dissipate excessive short-circuit power.

First, note if the constraints of (23) do not intersect at any point, i.e., $n\left(C_L/C_{in,\max}\right)^{1/n}+np_0>T$ there is no solution for the problem. On the other hand, if the intersection of the constraints of (23) results in exactly one point, i.e., when $n\left(C_L/C_{in,\max}\right)^{1/n}+np_0=T$, the only solution to $FCOA(n)$ is when all h_i's are equal to $T/n-p_0$. In other cases the optimization problem (23) can be solved by using the Lagrangian relaxation technique. In this technique, the constraints are relaxed and summed up in the objective function after multiplying them by non-negative coefficients, called the Lagrange multipliers. The new objective function is called the Lagrangian. In $FCOA(n)$, the Lagrangian is written as,

$$\begin{aligned} &L\left(\vec{h},\lambda_1,\lambda_2\right)=Area\left(\vec{h}\right)+\\ &+\lambda_1\left(\sum_{i=1}^{n}h_i-T_0+np_0\right)+\lambda_2\left(H_0-\prod_{i=1}^{n}h_i\right)\end{aligned} \tag{30}$$

where λ_1 and λ_2 are non-negative Lagrange multipliers, $\vec{h}=(h_1,...,h_n)$, and $H_0=C_L/C_{in,\min}$.

The set of Kuhn-Tucker conditions implies that at the optimal solution of $FCOA(n)$,

$$\frac{\partial L}{\partial h_i}=0 \qquad i=1,...,n \tag{31}$$

and

$$\lambda_1\left(\sum_{i=1}^{n}h_i-T_0+np_0\right)=0 \tag{32}$$

$$\lambda_2\left(H_0-\prod_{i=1}^{n}h_i\right)=0\,. \tag{33}$$

Now, considering the first set of conditions shown in (31), from $\partial L/\partial h_1=0$, it is concluded that,

$$-\frac{1}{h_1\pi_1}+\lambda_1-\frac{\pi_1}{h_1}\lambda_2=0 \tag{34}$$

where π_i is defined as,

$$\pi_i=\prod_{i}^{n}h_i\,. \tag{35}$$

Similarly, because $\partial L/\partial h_i=\partial L/\partial h_{i+1}=0$, we have $h_i\partial L/\partial h_i=h_{i+1}\partial L/\partial h_{i+1}$, which results in,

$$\lambda_1 h_i=\lambda_1 h_{i+1}-\frac{1}{\pi_{i+1}}\,. \tag{36}$$

One immediate result of (36) is that in the optimal solution of $FCOA(n)$, the values of h_i's are increasing, i.e.,

$$h_1\le h_2\le ...\le h_n\,. \tag{37}$$

The equality happens if and only if the required time and input capacitance constraints intersect at exactly one point.

Going back to the remaining Kuhn-Tucker conditions, from Lemma 1, one can see (32) is already satisfied. The remaining condition, as given in (33), implies that one of its terms is zero. If the input capacitance constraint of the optimization problem is "loose", i.e., in the optimal solution $H_0<\prod_{i=1}^{n}h_i$, it is necessary that $\lambda_2=0$. In this case, (33) implies that $\lambda_1=1/(h_1\pi_1)$ and (34) may be re-written as,

$$\frac{1}{h_1\pi_1}h_i=\frac{1}{h_1\pi_1}h_{i+1}-\frac{1}{\pi_{i+1}}\,. \tag{38}$$

Similarly,

$$\frac{1}{h_1\pi_1}h_{i-1}=\frac{1}{h_1\pi_1}h_i-\frac{1}{\pi_i} \tag{39}$$

and since $\pi_i=h_i\pi_{i+1}$, from (38) and (39), it is concluded that,

$$h_{i+1}=h_i(h_i-h_{i-1}+1) \tag{40}$$

where $h_0=0$.

Equation (40) is a recursive equation from which the values of all h_i's may be found as functions of h_1. Some of these values are shown in Table I. Plugging the values of h_i's as functions of h_1 into (24) and solving the polynomial equation, the value of h_1 which minimizes the objective function is found. To the best of our knowledge, there is no closed form solution to (40); however, one important property of this recurrence equation may be expressed by the following Lemma.

Lemma 3: In recurrence equation (40),

$$h_i>h_1^{2^{i-1}}\,. \tag{41}$$

Proof: We first show that all coefficients in polynomial $\Delta_i(h_1)=h_i-h_{i-1}$ are positive. We do this by using mathematical induction. First we note that $\Delta_1(h_1)=h_1$ is a positive-coefficient polynomial. Next, assuming $\Delta_k(h_1)$ is a positive coefficient for $k\ge 1$ (induction hypothesis). $\Delta_{k+1}(h_1)$ can be written as,

$$\begin{aligned}&\Delta_{k+1}(h_1)=h_{k+1}-h_k=h_k(h_k-h_{k-1}+1)-h_k\\&=h_k\Delta_k(h_1)\end{aligned} \tag{42}$$

hence, it is a positive-coefficient polynomial. Now, since for every i, $\Delta_i(h_1)$ is a positive-coefficient polynomial and $h_i=h_{i-1}(\Delta_{i-1}(h_1)+1)$, it follows that h_i is also a positive-coefficient polynomial with variable h_1; i.e.,

$$h_i=\sum_{j=1}^{ub}a_j h_1^j \tag{43}$$

where $a_j\ge 0$. It is easily verified that in equation (43), $ub=2^{i-1}$ and $a_{ub}=1$; hence, (41) holds. ■

From Lemma 3, one can see when the input capacitance constraint of $FCOA(n)$ is loose, in the optimal solution of (23) the values of h_i's grow exponentially and based on (11) and Fig. 2, the ratio of short circuit to dynamic power dissipation of the inverters grows accordingly. For example, if $T=23$, $C_L/C_{in,\max}=90$, $p_0=1$, and the polarity of the sink is positive, it can be verified that the optimum values for h_i's in $FCOA(2)$ are 6 and 15, and in $FCOA(4)$ the optimum values are 1, 2, 4, and 12, respectively. From Fig. 2 one can see that both these

scenarios result in excessive short-circuit power dissipation in the last stage of the chain.

TABLE I
SOME TERMS OF RECURSIVE EQUATION (40)

i	h_i
1	h_1
2	$h_1^2 + h_1$
3	$h_1^4 + h_1^3 + h_1^2 + h_1$
4	$h_1^8 + 2h_1^7 + 2h_1^6 + 2h_1^5 + 2h_1^4 + h_1^3 + h_1^2 + h_1$

IV. LOW-POWER FANOUT CHAINS

The discussion in Section III establishes that minimizing the area of a fanout chain will not minimize its power consumption. In this section, we generalize the problem and propose a mathematic program for low-power fanout chain design in multi-V_t and multi-L_{Gate} technologies. More precisely, we assume m discrete threshold voltages are available to be used in the inverters of the chain. In addition, we assume the channel length of inverters can be increased up to $L_{\max}$. The objective is to find the optimal number of inverters and their corresponding threshold voltages, channel lengths, and sizes to achieve the minimum power consumption in the active mode. When $m = 1$ and $L_{\max} = L_{nom}$, this problem simply becomes that of finding the optimal number of inverters and their corresponding sizes.

A. Problem Formulation

A multi-V_t and multi-L_{Gate} fanout chain is shown in Fig. 7. In this figure, h_i's denote the electrical efforts of the inverters, C_i's are the input capacitances, l_i's denote the In this figure, h_i's denote the electrical efforts of the channel lengths, and v_i's are the threshold voltages of the inverters. The goal is to find the number of inverters, n, h_i's, l_i's, and v_i's to minimize the total power dissipation while meeting both a timing constraint and an input capacitance upper bound constraint. Moreover, there is an upper bound on the length of the channel and the threshold voltage of each inverter should be selected from a given set of available threshold voltages.

Since increasing the channel length increases the threshold voltage of a transistor as well, we do not consider increasing both the channel length and threshold voltage of an inverter because the delay penalty tends to be too high. Moreover, we assume a multi-V_t design is achieved by ion implantation in the channel of the gate. Since changing the channel doping has negligible effect on the diffusion and gate capacitance, this assumption implies the dynamic and gate-tunneling leakage power consumptions are not affected by changing threshold voltages. However, changing the threshold voltage of an inverter alters its delay and sub- threshold leakage according to equations (4) and (16). On the other hand, as discussed in Section 0, this change also has an effect on the short-circuit power consumption of the fanout chain. Changing the channel length, on the other hand, alters delay and all components of power dissipation, as described in Section 0.

To simplify the equations, without loss of generality, we assume the driver and load of the chain are fixed-sized inverters. The driver is called the 0^{th} inverter, while the load is called the $(n+1)^{\text{th}}$ inverter.

Using the formulation derived in Section II, the power dissipation of the i^{th} inverter in the chain with the normalized channel length l_i can be expressed as,

$$P_i = \frac{C_L\left(\gamma_i k_{dyn} + k_{sub,i} l^{-\beta_{sub}} + k_{ox} l_i + k_{sc,i} h_{i-1} l_i^{-\beta_{sc1}} l_{i-1}^{\beta_{sc2}}\right)}{\prod_{j=i}^{n} h_j} \tag{44}$$

where $\gamma_i = (l_i + p_0)/(1 + p_0)$. Moreover, $k_{sub,i}$ is obtained from equation (17) and $k_{sc,i}$ is the short-circuit factor for the i^{th} inverter.

Therefore, the problem of optimizing the fanout chain for power dissipation becomes,

$$\begin{cases} Min & P(\vec{h}) = \sum_{i=1}^{n} P_i + k_{sc,n+1} h_n C_L \\ s.t. & (i) \quad \sum_{i=1}^{n} (p_i + g_i h_i) l_i^{\beta_d} \le T \\ & (ii) \quad H = \prod_{i=1}^{n} h_i \ge \frac{1}{l_1} \frac{C_L}{C_{in,\max}} \\ & (iii) \quad 1 \le l_i \le \frac{L_{\max}}{L_{nom}} \\ & (iv) \quad v_i \in \{V_1, ..., V_m\} \end{cases} \tag{45}$$

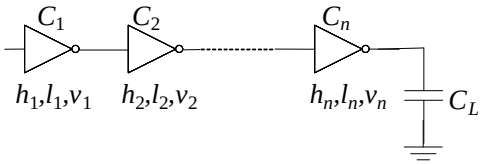

Fig. 7. A multi-V_t fanout chain.

where p_i and g_i are the parasitic delay and logical effort of the i^{th} inverter which operates with the threshold voltage of v_i. The first two constraints in (45) are the delay and input capacitance constraints while the third constraint of (45) imposes that there is an upper bound on the length of the channels. Finally, the forth constraint of (45) enforces the threshold voltages of the transistors of the inverters to be from the set of available threshold voltages $\{V_1, ..., V_m\}$, where V_1 is the nominal threshold voltage and $V_1 \le ... \le V_m$. The size and threshold voltage of the load are fixed; therefore, the dynamic and leakage power dissipations of the load inverter are constant. However, the short-circuit power dissipation of the load inverter is a function of the electrical effort of the last stage in the chain, i.e., h_n; thus, we include the short-circuit power dissipation of the load into the objective function.

Problem stated in (45) which is the Fanout Chain Optimization for minimum Power with n inverters, m threshold voltages, and an upper bound $L_{\max}$ for the channel length will be called $FCOP(n, m, L_{\max})$ in the rest

of this paper. To find the minimum-power fanout chain, $FCOP(n,m,L_{\max})$ should be solved for different values of n. Based on the polarity of the sink, only even or odd numbers should be considered for n.

Lemma 4: In the $FCOP(n,m,L_{\max})$ problem, the total electrical effort, H, is maximized when all v_i's are equal to V_1 and all l_i's are 1, and all h_i's are equal.

Proof: The geometric mean of a number of positive numbers is less than or equal to their arithmetic mean. The equality holds if and only if all values are equal. From the first constraint it can be seen that,

$$\begin{aligned} T &\geq \sum_{i=1}^{n} p_i l_i^{\beta_d} + \sum_{i=1}^{n} g_i h_i l_i^{\beta_d} \\ &\geq \sum_{i=1}^{n} p_i l_i^{\beta_d} + n\prod_{i=1}^{n}\left(g_i h_i l_i^{\beta_d}\right)^{1/n} \end{aligned} \quad (46)$$

From (46) it is concluded that in order to have a solution to $FCOP(n,m,L_{\max})$, the following relation must hold,

$$\frac{T - \sum_{i=1}^{n} p_i l_i^{\beta_d}}{n\prod_{i=1}^{n}\left(g_i l_i^{\beta_d}\right)^{1/n}} \geq \prod_{i=1}^{n}(h_i)^{1/n} = H^{1/n}. \quad (47)$$

Since $p_i \geq p_0$, $l_i \geq 1$ and $g_i \geq 1$, the maximum of H happens when all h_i's are equal, all l_i's are equal to 1, and all p_i's and g_i's assume their minimum values at p_0 and 1, respectively. The latter condition implies that all v_i's are equal. In this case, the maximum value of $H = \prod_{i=1}^{n} h_i$ is $H_{\max} = (T/n - p_0)^n$. ■

According to Lemma 4, there is a maximum value for H, $H_{\max}$, for any given buffer count; on the other hand, since $l_1 \leq L_{\max}/L_{nom}$, the second constraint of $FCOP(n,m,L_{\max})$ implies that H must be greater than $C_L/C_{in,\min} \times L_{nom}/L_{\max}$. Therefore, the only feasible buffer counts are those for which $H_{\max}$ is not less than the ratio $C_L/C_{in,\min} \times L_{nom}/L_{\max}$.

One important property of $FCOP(n,m,L_{\max})$ is that in its optimal solution, the delay of the fanout chain may not be equal to the specified required time T. To see why this is true, notice the objective function of $FCOP(n,m,L_{\max})$ is not a decreasing function of h_i's or l_i's; therefore, increasing h_i's or l_i's up to the point that $\sum_{i=1}^{n}(p_i + g_i h_i) l_i^{\beta_d} = T$ may not result in the minimum objective function.

If the design is not multi-L_{Gate}, i.e., $L_{\max} = L_{nom}$, then the third constraint in (45) will be eliminated from the problem and values of all l_i's become 1. Similarly, if the design is not multi-V_t, i.e., $m = 1$, the fourth constraint in (45) is eliminated and the values of all p_i's and g_i's become p_0 and 1, respectively. One can verify that constraints of $FCOP(n,1,L_{\max})$ are the same as $FCOA(n)$.

If the design is multi-V_t, i.e., $m \geq 2$, due to discrete values of v_i's in $FCOP(n,m,L_{\max})$, a posynomial problem solver needs to enumerate all possible assignments of the threshold voltages, i.e., m^n assignments, and solve the resulting mathematical program to find the minimum-power fanout chain by optimally selecting h_i's and l_i's. Due to its exponential runtime, such an enumeration is not possible. Hence, we use the same approach as in [6] to assign the threshold voltages. In this approach, the assignment of the threshold voltages is done as follows: starting from the source and going to sink, the values of the threshold voltages are increased. This heuristic called *monotone assignment* of the threshold voltages, greatly simplifies the problem and reduces the number of possible candidates to nm.

It is known that each additional threshold voltage needs one more mask layer in the fabrication process which results in increasing the fabrication cost. As a result, in many cases, only two threshold voltages are utilized in the circuit. At the same time, there are studies that show the benefit of having more than two threshold voltages is small [27]. So, in the sequel we concentrate on the problem of 2-V_t low-power fanout optimization, i.e., $FCOP(n,2,L_{\max})$. The results can be extended to handle more threshold voltages.

The pseudo-code for the $BestChain$ algorithm is provided in Fig. 8. First, by using the result of Lemma 4, for a given $C_{in,\max}$, C_L, and T, the $BestChain$ algorithm finds the lower and upper bounds of n. Based on the polarity of the sink node, only even or odd numbers of inverters between these bounds are considered when searching for the optimum solution. For a given n, the $BestChain$ algorithm attempts to solve the $FCOP(n,2,L_{\max})$ problem with all threshold voltages set to V_1, i.e., the nominal threshold voltage. If there is no feasible solution, then the timing and/or input capacitance constraints are too tight. The algorithm goes through a number of iterations where in each iteration, the threshold voltages of the last m inverters in the chain are set to V_2. This process is repeated until we find $\widetilde{m}$ such that there exists a feasible solution to the $FCOP(n,2,L_{\max})$ with $\widetilde{m}$ inverters, but not with $\widetilde{m}+2$ inverters[1]. In the pseudo-code, function $FCOP-FV$ finds the optimum solution to the $FCOP(n,2,L_{\max})$ problem with known threshold voltage values as captured by the assignment vector, $\vec{v}$. More precisely, $FCOP-FV$ algorithm finds l_i's of the first $n-m$ inverters, which have the nominal threshold voltage, and also h_i's of all inverters. Note since the $FCOP-FV$ function is called for fixed $\vec{v}$'s; this optimization problem is the minimization of a posynomial function with posynomial inequality constraints. This posynomial formulation is translated into a

[1] Alternatively, one can set all threshold voltages to V_2 and start reducing the threshold voltages to make the solution feasible. The result of this approach, however, will be the same as the one which starts with all threshold voltages set to V_1.

convex one by a change of variables $h_i = \exp(x_i)$ and $l_i = \exp(y_i)$ and is solved in polynomial time [26].

```
BestChain(C_in,max, C_L, T, pol){
 (ñ1, ñ2) = solution (C_L/C_in,max · L_nom/L_max) = (T/n − p0)^n;
 n1 = ⌊ñ1⌋ or ⌊ñ1⌋ + 1 (depending on pol)
 n2 = ⌊ñ2⌋
 (pwr*, h*, l*, v*) = (+∞, ∅, ∅, ∅)
 For n = n1 to n2 step 2{
  For i = 1 to n
   v(i) = V2
  (h, l, pwr) = FCOP − FV(n, T, C_in,max, C_L, v)
  If h = ∅
   continue
  If pwr < pwr*
   (pwr*, h*, l*, v*) = (pwr, h, l, v)
  For m = n to 1 step -1{
   v(m) = V2
   (h, l, pwr) = FCOP − FV(n, T, C_in,max, C_L, v)
   If pwr > pwr*
    (pwr*, h*, l*, v*) = (pwr, h, l*, v)
  }
 }
 Return (pwr*, h*, l*, v*)
}
```

Fig. 8. $BestChain$ algorithm

V. Building a Fanout Tree

In this section we show how to build a fanout tree with more than one sink. Reference [7] introduced two transformations that could be performed on a fanout tree, namely merging and splitting, and showed these transformations preserve area, delay, and input capacitance of the fanout tree. We have extended the merging and splitting transformations to handle multi-V_t and multi-L_{Gate} fanout trees, as depicted in Fig. 9.

Theorem 2: The extended split/merge transformations applied to a multi-V_t and multi-L_{Gate} fanout tree as depicted in Fig. 9 preserve the delay, input capacitance, and power dissipation values of the tree.

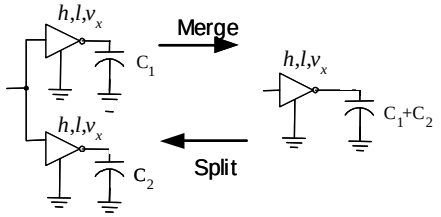

Fig. 9. Extended split/merge transformations for Multi-V_t and multi-L_{Gate} inverters

Proof: We provide the proof for the split transformation. Before splitting, the delay of the inverter is $(p_x + g_x h)l^{\beta_d}$ while the input capacitance is $(C_1 + C_2)/h$. After splitting the original inverter into two inverters with equal electrical efforts of h and equal channel length l and threshold voltages of v_x, the delay through the inverter in either branch will be $(p_x + g_x h)l^{\beta_d}$ while the input capacitances will be C_1/h and C_2/h which sum up to $(C_1 + C_2)/h$. Therefore, this transformation preserves the delay and input capacitance values. Since this transformation does not change the input capacitance, the electrical effort of the previous stage, which characterizes the short-circuit power dissipation of two inverters before the merge transformation, does not change; it is easy to see the capacitive and leakage power consumption of the tree remains the same after the transformation. Moreover, since this transformation does not change the channel length of the inverter transistors, the short circuit power dissipations of C_1 and C_2 remain the same. Hence, the total power dissipation of the fanout tree before and after the split transformation remains the same. ■

Since extended split/merge transformations preserve the delay, input capacitance, and power dissipation values, by using these transformations, any fanout optimization problem with m sink nodes, can be converted to m fanout chain optimization problems, whose respective power dissipations will be the same.

To apply these transformations, two issues should be addressed. The first issue is the input capacitance allocation to different chains in a decomposed fanout tree and the second issue is the validity of a continuous-size inverter library. In the following we address these questions.

A. Input Capacitance Allocation

The Input Capacitance Allocation to achieve minimum Power (ICAP) problem is defined as follows: Given a number of sinks, each with a required time, polarity, and capacitive load, and a total budget on input capacitance $C_{in,tot}$, allocate portions of $C_{in,tot}$ to each fanout chains such that the total power is minimized while the given constraints for all sinks are satisfied. In this section we show the ICAP problem is NP-complete and we use a heuristic to allocate the input capacitance to different chains in a decomposed fanout tree.

Lemma 5: For a fixed number of inverters in a multi-V_t and multi-L_{Gate} fanout chain, the power cost is a decreasing function of the input capacitance bound, $C_{in,\max}$.

Proof: From the second constraint in (45), it is seen that increasing the input capacitance constraint of a fanout chain expands the feasible space of the optimization problem. Therefore, there exists either a better solution with lower power consumption or one with the same power consumption; that is, the power cost in a fanout chain is a decreasing function of the input capacitance bound. ■

Theorem 3: The ICAP problem is NP-Complete.

Proof: To prove that ICAP is NP-Complete, we show the 0-1 Knapsack problem may be reduced to the ICAP

problem. In the 0-1 Knapsack problem, there are some items, each with its own value and weight; the objective is to select some items such that the total value of the selected items is maximized while their total weight is not more than a given budget. In the ICAP problem, however, the objective is to minimize power. To make ICAP a maximization problem, we consider the negative of power as the objective function. According to Lemma 5, the power cost is a decreasing function of the input capacitance constraint; therefore, the graph of the maximum of negative power over all inverter counts looks like Fig. 10. Notice this graph exhibits a piecewise behavior because power is represented by different functions for different inverter counts. The piecewise nature of power versus input capacitance helps us to reduce the 0-1 Knapsack problem to the ICAP problem.

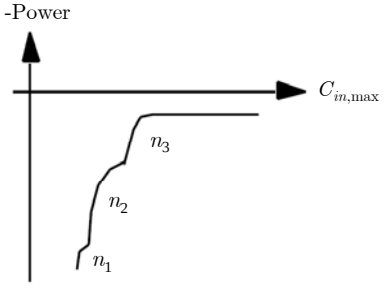

Fig. 10. Negative of power dissipation versus the input capacitance curve.

This reduction is similar to the reduction of the Knapsack problem to the problem of input capacitance allocation for minimum area, hence, it is omitted here. Interested readers may refer to [3] for details. After proving the ICAP is NP-Hard, we show the decision version of the ICAP can be tested in polynomial time. This is clearly true because one can add up the input capacitances of each branch and compare it with the input capacitance budget in linear time. Therefore, the ICAP is in NP; since it was shown that the ICAP is NP-Hard, therefore, the ICAP problem is NP-Complete. ■

The heuristic we use for solving the ICAP problem is similar to that of [3] and starts by allocating the minimum input capacitance required for each branch to have a feasible fanout chain solution. Next, the remaining total input capacitance is divided between the chains in proportion to the positive slopes of $H_{\max,i}$ versus n_i for each branch i.

B. Discrete-Size Inverter Library

The second issue to address is the assumption of the availability of a continuous-size inverter library. In reality, in the ASIC libraries, although many different inverter sizes are available, these sizes are discrete (there are typically 8-16 different inverter sizes in an industrial state-of-the-art ASIC library.) So the solution needs to be mapped onto one of the available inverters in the library. The main problem when rounding the inverter sizes is that it may result in significant errors. To address this problem, reference [3] defined a constant ε_h and merged two inverters on different chains if the difference between their electrical efforts was less than or equal to ε_h. Notice, in general, two inverters are merged if the rounding error after merging is smaller than the sum of the rounding errors of inverters before the merge operation. We adopt the same heuristic with the additional requirement that the two candidate inverters should also have the same threshold voltage and the difference between l_1 and l_2 should be smaller than a constant ε_l. Merging is performed starting at the source of the signal and proceeds toward sinks.

VI. Simulation Results

The proposed technique in Section IV, which we call *LPFO*, has been developed in the SIS framework [28]. The MOSEK convex optimization tool [29] has been used to solve the mathematical problems. To extract the parameters used in the optimization problems, we performed transistor level simulation of devices in HSPICE [30] on a 65nm technology node [31]. The simulations have been done at the frequency of 1GHz, supply voltage of 1.1V, and die temperature of 100°C. Moreover, we assumed the switching activity of the source node is 5% and the probability of this node being at logic one is 0.5 in all circuits.

To extract the short-circuit power consumption, the method introduced in [32] was utilized. Furthermore, the Matlab optimization toolbox was used to find values of fitting parameters for the delay and power dissipation models.

The parameters of this technology node are shown in Table II. In this table, $k_{sc,LH}$ is the short-circuit factor of an inverter whose threshold voltage is high while the threshold voltage of its driver is low. $k_{sc,LL}$, $k_{sc,HL}$, and $k_{sc,HH}$ are defined similarly. The values of short circuit factors as well as $k_{sub,low}$, $k_{sub,high}$, and k_{ox} are normalized with respect to k_{dyn}. In this set of experiments, a standard cell library consisting of sixteen different inverters was used to map the fanout trees.

TABLE II
TECHNOLOGY PARAMETERS USED IN SIMULATIONS

Parameter	Value	Parameter	Value
$V_{t,low}$	0.2V	$k_{sc,LL}$	0.069
$V_{t,high}$	0.3V	$k_{sc,LH}$	0.006
γ	3.5	$k_{sc,HL}$	0.099
τ_0	8.6e-12	$k_{sc,HH}$	0.014
p_0	1.33	β_{sub}	7.4
k_{dyn}	1.000	β_{sc1}	22.5
$k_{sub,low}$	0.343	β_{sc2}	4.4
$k_{sub,high}$	0.078	β_d	1.6
k_{ox}	0.096	$L_{\max} / L_{nom}$	1.1

To study the efficiency of our technique in reducing the power consumption of the fanout trees, we conducted two

sets of experiments. In the first set of experiments, whose results are shown in Table III, we assumed the options of multi-V_t and multi-L_{Gate} are not available in the library and compared the results of *LPFO* with the results of low-area fanout optimization (*LEOPARD*) [3] for a few random problems in the form of fanout chains. In this table $C_{in.\max}$ denotes the maximum allowed capacitance at the input of the fanout chain, C_{out} is the load capacitance, and pol is the polarity of the sink. In each fanout chain, first the path delay was minimized using the technique proposed in [11]. Next, each chain was given some additional slack and either *LPFO* or *LEOPARD* algorithm was invoked to minimize the power dissipation or the area of the fanout chain. Each optimized chain was mapped to the library of inverters, and detailed SPICE simulation was carried out on the circuit to measure the power consumption. From Table III one can see minimizing the area of the fanout chains in many cases increases the total power consumption. On the other hand, when the fanout chains are optimized for power, by increasing the available slack in the chain, the power reduction saturates at some point. From the table, the power consumption of the minimum power fanout chains is not always a decreasing function of available slack. This is due to round-off error in mapping the continuous-size inverters to discrete-size inverters in the library.

The second set of experimental results compares *LPFO* with *LEOPARD* and the SIS fanout optimization program for a set of problems in the form of fanout trees. SIS runs different fanout optimization algorithms, namely *Two-Level*, *Bottom-Up*, *Balanced*, *LT-Tree*, and reports the best one [8]. In this set of experiments, the same standard cell library used for *LPFO* and *LEOPARD* has been utilized as the SIS library. For each inverter $\tau_{intrinsic}$ and R_{out} were specified for the SIS library delay model and p_0 and τ_0 were specified for the logical effort delay model. A very close match between the SIS delay and logical effort delay model values was enforced.

The fanout optimization programs of SIS were first used to perform fanout optimization for a set of problems. Next the delay and input capacitance resulting from SIS were used as constraints for *LPFO* and *LEOPARD*. After performing the fanout optimization, the SPICE netlist for each circuit was generated and detailed HSPICE simulation was performed to measure the delay and the power consumption of the circuit. The results of these experiments are reported in Table IV. The first column is the name of the problem instance, the second column denotes the number of sinks in the fanout problem, columns 3 and 4 respectively show the area and power consumption of each fanout problem achieved by running the SIS fanout optimization and the remaining columns show the area and power reduction of *LEOPARD* and *LPFO* algorithms over corresponding values of SIS program. From Table IV one can see fanout trees resulting from *LEOPARD*, on average, consume 11.79% more power than those achieved by SIS. Utilizing *LPFO*, on the other hand, reduces not only the power consumption of fanout trees by an average of 11.17% but also their area by an average of 29.64%.

The runtime of our algorithm for the largest problem with 30 sinks is about 5 seconds when the options of multi-V_t and multi-L_{Gate} are not available, 7 seconds when only the multi-L_{Gate} option is available, 21 seconds when only the multi-V_t option is available, and 24 seconds when both multi-V_t and multi-L_{Gate} options are available.

To show how precisely the fanout chains are optimized, Table V reports design parameters for one of the fanout chain problems, i.e., FC10, for four different cases corresponding to three cases where only sizes, only threshold voltages, or only channel lengths of the inverters are optimized and a fourth case where all of these parameters are used in the optimization process. The design parameters of each inverters are shown as a triplet (x,y,z), where the entries of the triplet respectively correspond to the size, threshold voltage, and normalized channel length of the inverter. The corresponding leakage and total power dissipation reductions over the minimum delay fanout chain are also reported in this table. To optimize the power consumption of the chain when only the option of multi-V_t (multi-L_{Gate}) is available, first the chain is optimized for delay by assigning equal electrical efforts to different stages; next the extra slack in the chain is utilized to minimize the total power consumption by adjusting the threshold voltages (channel lengths) of the inverters.

From this table one can see that when only the option of sizing is available, the reduction in total power is comparable to the case that all options are used in the optimization process. This is due to the fact that in the technology we used for simulations, the subthreshold leakage power dissipation of a low-V_t inverter is a rather small portion of its total power consumption (about 20%). Therefore, the effectiveness of multi-V_t and multi-L_{Gate} techniques is reduced. However, as the sixth column of Table V shows, there is a significant difference in the leakage power consumption of these two cases. Therefore, in other technologies or at higher die temperatures where subthreshold leakage becomes larger, the effectiveness of the fourth technique will be more pronounced.

TABLE III
THE COMPARISON OF THE TOTAL POWER CONSUMPTION IN MINIMUM DELAY FANOUT CHAINS, LEOPARD, AND LPFO

Circuit	Circuit Specification			Min Delay Circuit		Power Reduction (%)							
						LEOPARD				LPFO			
	$C_{in,\max}$	C_{out}	pol	Org Power (μW)	Org Delay (ps)	Slack 10%	Slack 20%	Slack 30%	Slack 40%	Slack 10%	Slack 20%	Slack 30%	Slack 40%
FC1	1	64	+	20.9	140	5.94	-31.51	-55.9	-55.9	10.3	10.17	7.10	7.10
FC2	1	100	+	14.3	129.8	-2.54	-12.85	-41.79	-72.3	3.81	4.52	2.57	2.59
FC3	20	100	+	23.9	61.2	13.13	16.44	16.68	15.95	13.25	17.2	18.04	18.62
FC4	30	80	+	7.5	36.9	21.11	28.5	33.49	35.14	21.61	28.77	33.58	36.14
FC5	50	200	+	7.6	52.3	17.16	24.2	28.37	29.84	18.52	25.65	29.75	31.33
FC6	20	50	-	9.4	69	5.02	7.32	7.98	7.70	5.02	7.32	7.98	7.70
FC7	15	200	-	22.5	65.2	15.04	14.72	12.69	-27.62	15.92	17.84	18.32	18.05
FC8	2	100	-	48.4	94.6	-7.23	-20.06	-35.59	-47.64	0	0	0	0
FC9	8	50	-	7.5	115.2	-7.06	-17.61	-33.83	-33.83	0	0	0	0
FC10	10	150	-	19.1	42.2	13.48	12.17	9.46	5.00	13.87	15.85	17.27	18.25
Average						**7.40**	**2.13**	**-5.84**	**-14.37**	**10.23**	**12.73**	**13.46**	**13.98**

TABLE IV
COMPARISON OF SIS, LEOPARD, AND LFPO FANOUT OPTIMIZATION ALGORITHMS

Circuit	Sink	SIS		LEOPARD		LPFO	
		Area	Power (μW)	Area Reduction over SIS	Power Reduction over SIS	Area Reduction over SIS	Power Reduction over SIS
FT1	5	304	14.4	47.70	11.81	43.09	16.67
FT2	7	1082	119.0	62.38	-16.81	9.89	6.72
FT3	8	1026	63.3	48.34	-18.17	42.01	12.48
FT4	10	1139	68.3	79.54	-16.40	53.99	13.47
FT5	20	1347	105.0	54.94	-28.57	18.63	2.76
FT6	12	928	64.4	45.37	-8.07	26.51	12.73
FT7	14	1490	109.1	67.92	-22.82	45.97	17.60
FT8	14	838	86.3	34.01	-9.04	-7.28	9.15
FT9	25	2853	150.0	78.48	-18.00	56.78	15.33
FT10	30	2496	160.0	60.10	-15.63	27.92	6.88
FT11	10	715	46.7	52.73	-0.86	30.91	13.49
FT12	12	1465	73.4	59.73	3.00	50.17	13.62
FT13	15	1218	92.8	38.83	-11.31	16.67	13.15
FT14	16	1099	94.1	38.31	-7.76	8.64	8.29
FT15	22	1334	115.0	48.20	-18.26	20.69	5.22
Average				**54.44**	**-11.79**	**29.64**	**11.17**

TABLE V
DESIGN PARAMETERS OF INVERTERS IN FANOUT CHAIN FC10 WHEN DIFFERENT POWER OPTIMIZATION TECHNIQUES ARE USED

	Power Reduction Technique	Design Parameters of Inverters			Leakage Reduction (%)	Total Power Reduction (%)
		Inverter1	Inverter 2	Inverter 3		
Slack=10%	Sizing only	(8, 0.2, 1.00)	(12, 0.2, 1.00)	(32, 0.2, 1.00)	45.83	13.87
	Multi-V_t only	(8, 0.2, 1.00)	(24, 0.2, 1.00)	(64, 0.2, 1.00)	0.00	0.00
	Multi-L_{Gate} only	(8, 0.2, 1.03)	(24, 0.2, 1.09)	(64, 0.2, 1.05)	36.67	4.85
	All together	(8, 0.2, 1.00)	(12, 0.2, 1.00)	(32, 0.2, 1.00)	45.83	13.87
Slack=20%	Sizing only	(6, 0.2, 1.00)	(8, 0.2, 1.00)	(32, 0.2, 1.00)	52.08	15.41
	Multi-V_t only	(8, 0.2, 1.00)	(24, 0.2, 1.00)	(64, 0.3, 1.00)	51.51	8.38
	Multi-L_{Gate} only	(8, 0.2, 1.03)	(24, 0.2, 1.10)	(64, 0.2, 1.06)	41.25	5.04
	All together	(8, 0.2, 1.00)	(12, 0.2, 1.00)	(40, 0.3, 1.00)	69.69	15.85
Slack=30%	Sizing only	(4, 0.2, 1.00)	(8, 0.2, 1.00)	(32, 0.2, 1.00)	54.17	15.71
	Multi-V_t only	(8, 0.2, 1.00)	(24, 0.3, 1.00)	(64, 0.3, 1.00)	77.26	12.08
	Multi-L_{Gate} only	(8, 0.2, 1.03)	(24, 0.2, 1.10)	(64, 0.2, 1.06)	41.25	5.04
	All together	(6, 0.2, 1.00)	(8, 0.2, 1.00)	(40, 0.3, 1.00)	75.94	17.27
Slack=40%	Sizing only	(3, 0.2, 1.00)	(8, 0.2, 1.00)	(32, 0.2, 1.00)	55.21	16.08
	Multi-V_t only	(8, 0.3, 1.00)	(24, 0.3, 1.00)	(64, 0.3, 1.00)	77.26	12.90
	Multi-L_{Gate} only	(8, 0.2, 1.03)	(24, 0.2, 1.10)	(64, 0.2, 1.06)	41.25	5.04
	All together	(4, 0.2, 1.00)	(8, 0.2, 1.00)	(40, 0.3, 1.00)	78.02	18.25

Our last set of experimental results demonstrates how the size of inverter library affects the quality of results in the proposed technique (the size of a library is defined as the number of gates in it). Table VI shows the average and maximum error in power consumption of fanout chains (shown in Table III) as a result of mapping continuous inverter sizes to discrete values in inverter libraries with different sizes. From this table one can see that with an inverter library size of 10 or more, the mapping error becomes quite negligible.

TABLE VI
MAPPING ERROR AS A FUNCTION OF INVERTER LIBRARY SIZE

Inverter Library Size	Maximum Error (%)	Average Error (%)
4	15.5	57.3
6	4.1	8.7
8	3.4	7.3
10	1.8	7.3
12	0.8	2.1
14	0.9	2.1

Note in our problem setup and in the simulation results, we ignored the interconnect power dissipation and delay costs. The reason is that we do the fanout optimization during logic synthesis and prior to generating layout. Therefore, locations of the source and the sinks are not known. As a result the interconnect delay information cannot be accurately modeled. It is thus reasonable to assume the expected values of delay and power dissipation per wire in the inverter chain or the fanout tree are nearly the same. This constant contribution can, thus, be taken out of the problem formulation by properly adjusting the required time constraints on the sinks and adding a constant term to the total power equation.

VII. CONCLUSION

In this paper we showed the fanout optimization with area and power objective functions are not the same and a fanout tree optimized for area may dissipate excessive short-circuit power. By modeling all components of power dissipation, i.e., dynamic, short-circuit, sub-threshold leakage and gate tunneling leakage, we formulated the fanout optimization problem as a geometric program for a circuit with one sink. To reduce the leakage power consumption, we proposed using multi-V_t and multi-L_{Gate} inverters in the fanout trees. Experimental results show the proposed technique is effective in reducing the total power consumption of fanout trees.

REFERENCES

[1] B. Amelifard, F. Fallah, and M. Pedram, "Low-power fanout optimization using MTCMOS and multi-Vt techniques," in *Proc. of International Symposium on Low Power Electronics and Design*, 2006, pp. 334 -337.

[2] A. Salek, J. Lou, and M. Pedram, "Hierarchical buffered routing tree generation," *IEEE Trans. on Computer Aided Design*, vol. 21, no. 5, May 2002, pp. 554-567.

[3] P. Rezvani and M. Pedram, "A fanout optimization algorithm based on the effort delay model," *IEEE Trans. on Computer Aided Design*, vol. 22, no. 12, Dec. 2003, pp. 1671-1678.

[4] D. Zhou and X. Liu, "Minimization of chip size and power consumption of high-speed VLSI buffers," in *Proc. of International Symposium on Physical Design*, 1997, pp. 186-191.

[5] K. J. Singh and A. Sangiovanni-Vincentelli, "A heuristic algorithm for the fanout problem," in *Proc. of Design Automation Conference*, 1990, pp. 357-360.

[6] B. Amelifard, F. Fallah, and M. Pedram, "Low-power fanout optimization using multiple threshold voltage inverters," in *Proc. of International Symposium on Low Power Electronics and Design*, 2005, pp. 95-98.

[7] D. S. Kung, "A fast fanout optimization algorithm for near-continuous buffer libraries," in *Proc. of Design Automation Conference*, 1998, pp. 352-355.

[8] H. Touati, "Performance-oriented technology mapping," Ph.D. dissertation, University of California, Berkeley, 1990.

[9] C. L. Berman, J. L. Carter, and K. F. Day, "The fanout problem: from theory to practice," in *Proc. of Decennial Caltech Conference Advanced Research in VLSI*, 1989, pp. 69-99.

[10] K. Kodandapani, J. Grodstein, A. Domic, and H. Touati, "A simple algorithm for fanout optimization using high-performance buffer libraries," in *Proc. of International Conference on Computer-Aided Design*, 1993, pp. 466-471.

[11] I. Sutherland, B. Sproull, and D. Harris, *Logical Effort: Designing Fast CMOS Circuits*. San Francisco, CA: Morgan Kaufmann, 1999.

[12] N. Sirisantana, L. Wei, and K. Roy, "High performance low power CMOS circuits using multiple channel length and multiple oxide thickness," in *Proc. of International Conference on Computer Design*, 2000, pp. 227-232.

[13] P. Gupta, A. B. Kahng, P. Sharma, and D. Sylvester, "Selective gate-length biasing for cost-effective runtime leakage control," in *Proc. of Design Automation Conference*, 2004, pp. 327-330

[14] Blaze-DFM MO Tool [online] http://www.blaze-dfm.com/products/products1.html

[15] Chip Design Magazine [online] http://www.chipdesignmag.com/display.php?articleId=475

[16] B. Hu, Y. Watanabe, A. Kondratyev, and M. Marek-Sadowska, "Gain-based technology mapping for discrete-size cell libraries," in *Proc. of Design Automation Conference*, 2003, pp. 574-579.

[17] S. Karandikar and S. Sapatnekar, "Logical effort based technology mapping," in *Proc. of International Conference on Computer-Aided Design*, 2004, pp. 419-422.

[18] W. Chen, C. Hsieh, and M. Pedram, "Simultaneous gate sizing and fanout optimization," in *Proc. of International Conference on Computer-Aided Design*, 2000, pp. 374-378.

[19] Magma Design Automation. Gain Based Synthesis: Speeding RTL to Silicon, 2002.

[20] L. Stok, D. S. Kung, D. Brand, and A. D. Drumm, "BooleDozer: logic synthesis for ASICs," *IBM Journal of Research and Development*, vol. 40, no. 4, Jul. 1996, pp. 407-430.

[21] T. Sakurai and A. R. Newton, "Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 2, Apr. 1990, pp. 584-594.

[22] M. Pedram, "Power minimization in IC design: principles and applications," *ACM Trans. on Design Automation of Electronic Systems*, vol. 1, no. 1, Jan. 1996, pp. 3-56.

[23] A. P. Chandrakasan and R. W. Brodersen, *Low Power Digital CMOS Design*. Norwell, MA: Kluwer, 1995.

[24] V. De, A. Keshavarzi, S. Narendra, and J. Kao, "Techniques for leakage power reduction," in *Design of High-Performance Microprocessor Circuits*, A. Chandrakasan, W. J. Bowhill, and F. Fox, Eds. Piscataway, NJ: IEEE, 2001.

[25] D. Lee, D. Blaauw, and D. Sylvester, "Gate oxide leakage current analysis and reduction for VLSI circuits," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 2, Feb. 2004, pp. 155-166.

[26] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2003.

[27] A. Sirvastava, "Simultaneous Vt selection and assignment for leakage optimization," in *Proc. of International Symposium on Low Power Electronics and Design*, 2003, pp. 146-151.

[28] E. M. Sentovich, K. J. Singh, L. Lavagno, C. Moon, R. Murgai, A. Saldanha, and H. Savoj, "SIS: A System for Sequential Circuit Synthesis," University of California, Berkeley, Report M92/41, May 1992.

[29] MOSEK Optimization Software, [online] http://www.mosek.com

[30] HSPICE: The gold standard for accurate circuit simulation, [online] http://www.synopsys.com/products/mixedsignal/hspice/hspice.html

[31] Predictive Technology Model, [online] http://www.eas.asu.edu/~ptm/

[32] A. Alvandpour, P. Larsson-Edefors, and C. Svensson, "Seperation and extraction of short-circuit power consumption in digital CMOS circuits," in *Proc. of International Symposium on Low Power Electronics and Design*, 1998, pp. 245-249.