Optimal P/N Width Ratio Selection for Standard Cell Libraries

David S. Kung and Ruchir Puri
IBM T. J. Watson Research Center
Yorktown Heights, NY 10598

ABSTRACT

The effectiveness of logic synthesis to satisfy increasingly tight timing constraints in deep-submicron high-performance circuits heavily depends on the range and variety of logic gates available in the standard cell library. Primarily, research in the design of high-performance standard cell libraries has been focused on drive strength selection of various logic gates. Since CMOS logic circuit delays not only depend on the drive strength of each gate but also on its P/N width ratio, it is crucial to provide good P/N width ratios for each cell. The main contribution of this paper is the development of a theoretical framework through which library designers can determine "optimal" P/N width ratio for each logic gate in their high-performance standard cell library. This theoretical framework utilizes new gate delay models that explicitly represent the dependence of delay on P/N width ratio and load. These delay models yield highly accurate delay for CMOS gates in a $0.12\mu m$ L_{eff} deep-submicron technology.

1 INTRODUCTION

The relentless pursuit of high performance has pushed logic and circuit designers to utilize every delay and area optimization technique at their disposal. To emulate the flexibility of custom designs, ASIC and semi-custom designers are providing high-performance standard cell libraries that not only offer a wide range and variety of gate sizes (or drive strengths) but also a wide range of P/N width ratios for each gate size [12][13]. Traditionally in logic synthesis, delay optimization techniques have heavily relied on gate sizing algorithms [2][3][15] which vary drive strengths of gates to optimize circuit delay. Since the delay in CMOS logic circuits not only depends on the drive strength of each stage but also on its P/N width ratio, it is crucial to provide good P/N width ratios for each cell in the ASIC library for satisfying increasingly tight timing constraints of deep-submicron high-performance circuits. Recent research in the area of standard cell library design [1][7][8] has mainly focused on drive strength selection of various logic gates. Unfortunately, there has been no research in the direction of developing a theoretical framework for selecting optimal P/N width ratios.

In general, for selecting P/N width ratios of CMOS logic cells, library designers are often concerned with minimizing the average of the rising and falling path delays[1] because a transition through a chain of CMOS gates incurs alternating rising and falling transitions [5][9][13]. It is known that achieving minimum delay through a chain of inverters requires asymmetric rising and falling transition delays [10]. Asymmetric rise and fall delays through a CMOS gate can be obtained by increasing the size of NMOS devices at the expense of PMOS device sizes or vice-versa. Due to the slower mobility of holes than electrons, the falling gate delay through the NMOS pull-down is more sensitive to device size changes than the rising gate delay through the PMOS pull-up. This is illustrated in Figure 1 which shows the variation of falling and rising transition delays through an inverter with varying NMOS and PMOS device width respectively. It can be seen that increasing the size of NMOS by $W\mu m$ will result in a larger decrease in falling output gate delay as compared to the reduction in rising output gate delay due to same $W\mu m$ increase in PMOS size. For logic gates such as inverter, NAND2, and NORs, the PMOS device contributes more to the input pin capacitance than the NMOS device for equal rise and fall delays. Thus, for these gates it is possible to reduce the average delay by skewing the P/N width ratio in favor of pull-down NMOS devices. However, in the case of gates such as high fanin NANDs, the NMOS device contributes more to the input pin capacitance than the PMOS device for equal rise and fall delays. For these gates the P/N width ratio is skewed in favor of pull-up PMOS devices to minimize average delay. There is an inherent tradeoff in making the P/N width too small or too large. If the P/N width ratio is made too small, the rising transition delay becomes too large; if this ratio is made too large, it will result in a large input pin capacitance which will slow down the driver gate.

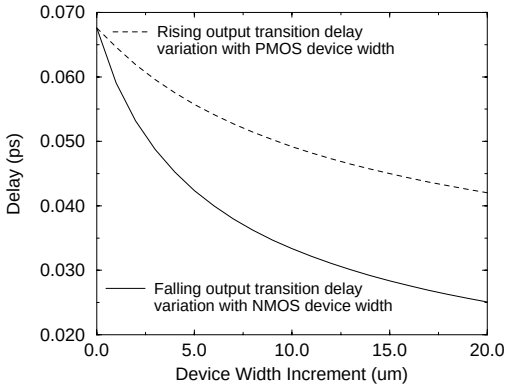

Figure 1: Falling, rising output transition delay variation with NMOS, PMOS device width respectively for an inverter in $0.12\mu m$ L_{eff} CMOS technology.

In this paper, we develop a theoretical framework through which library designers can determine "optimal" P/N width ratio for each logic gate in their high-performance standard cell library. This framework utilizes new gate delay models discussed in detail in the following Section. First, we propose an *analytical delay model* that separates the delay dependence on load from the delay dependence on P/N width ratio. We then generalize this analytical delay model to accurately model device behavior in deep-submicron technologies. The results show that this *generalized delay model* can yield highly accurate delay for gates in a $0.12\mu m$ L_{eff} CMOS technology when compared with device level simulation results. In Section 3, these analytical and generalized delay models are utilized to formulate the P/N width ratio optimization problem. We show that under the analytical delay model, the optimal P/N width ratio of a logic gate for minimum average path delay is independent of its position along the circuit path and the network topology. We then extend this result to the generalized delay model to find optimal P/N width ratios accurately. In this case the optimal P/N width ratio of a logic gate for minimum path delay is dependent on its load and input slew. However, the variation of optimal P/N

[1] In the case of a chain of CMOS stages with even number of gates of same logic type, average delay of rising and falling transition delays is equivalent to the worst case delay. However, for short paths, the average delay may differ slightly from the worst case delay in the case of asymmetric rising and falling gate delays.

0-7803-5832-X /99/$10.00 ©1999 IEEE.

width ratio over the entire design range causes a negligible change in the minimum delay as shown by experimental results given in Section 4. For each pin to pin timing arc of each gate in a standard cell library (in a $0.12\mu m$ L_{eff} CMOS technology), we provide an "optimal" P/N width ratio for minimum path delay.

2 DELAY MODEL

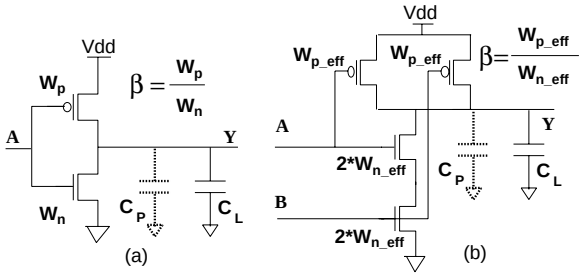

Figure 2: Schematic of (a) Inverter (b) NAND2.

The delay of a transition through a CMOS logic gate is a function of its load and input pin capacitance, its P/N width ratio, and its input transition time (input slew). In our analysis, we first use a delay model that is derived from a step transition response (zero input slew) of an inverter shown in Figure 2(a). Subsequently, this model is generalized to include explicitly P/N width ratio (denoted as β in this paper) for modeling asymmetric rise and fall delays. Thus, we model rise-fall and fall-rise gate delays as an explicit function of β. Since input slew has significant effect on gate delays, we model slew dependence of delay by using different coefficients for various slew values in our delay model. We will show that this generalized delay model is very accurate despite its simplicity.

Assuming a non-linear I-V MOSFET model:

$$I_{ds} = k\left((V_{gs} - V_t)V_{ds} - \frac{{V_{ds}}^2}{2}\right),$$

the delay of a rising or a falling input step transition through an inverter with output load C_l is given by:

$$t_d = \frac{C_l + C_p}{k}\left(\frac{2|V_T|}{(V_{DD} - |V_T|)^2} + \frac{1}{V_{DD} - |V_T|} \cdot ln\frac{3V_{DD} - 2|V_T|}{V_{DD}}\right)$$

where C_p is the internal parasitic capacitance, V_{DD} is the supply voltage, V_T is the threshold voltage and k is the device transconductance. The factor $\frac{1}{k}\left(\frac{2|V_T|}{(V_{DD}-|V_T|)^2} + \frac{1}{V_{DD}-|V_T|} \cdot ln\frac{3V_{DD}-2|V_T|}{V_{DD}}\right)$ can be interpreted as the device resistance R, so the above equation can be simplified as $t_d = R(C_l + C_p)$. This simple R-C delay equation is also known as Elmore delay model [4][11]. For a MOSFET, transconductance parameter k is dependent on device size $\frac{W}{L}$, carrier mobility μ, and gate capacitance per unit area C_{ox} and is given by $\mu C_{ox}\frac{W}{L}$ (i.e., $k = \mu C_{ox}\frac{W}{L}$). Since parameters V_{DD}, V_T, and C_{ox} are fixed for any given technology, we represent $\frac{L}{C_{ox}}\left(\frac{2V_T}{(V_{DD}-V_T)^2} + \frac{1}{V_{DD}-V_T} \cdot ln\frac{3V_{DD}-2V_T}{V_{DD}}\right)$ as a constant k', assuming that all MOSFETs within the technology are of constant length L. This further simplifies delay t_d to yield:

$$t_d = k'\left(\frac{C_l + C_p}{\mu W}\right)$$

It is known that both gate capacitance and the parasitic capacitance (i.e., diffusion capacitance) of a MOSFET scale with its size. Thus, it is reasonable to assume that the ratio of parasitic capacitance (C_p) of a logic gate to its input pin capacitance (C_{in}) remains constant, i.e., $\frac{C_p}{C_{in}} = constant\,(K)$. Figure 3 shows the variation in the ratio of parasitic capacitance to its input pin capacitance (i.e., $\frac{C_p}{C_{in}}$) for an inverter with the increase in inverter size. It can be seen that "$\frac{C_p}{C_{in}}$" is almost invariant with any change in inverter size. We utilize this invariance of $\frac{C_p}{C_{in}}$ to obtain the analytical delay model given in equations 2 and 3. However, as discussed later, due to non-linear nature of deep-submicron MOSFET, miller effect etc., $\frac{C_p}{C_{in}}$ may not remain invariant for all cases in a deep-submicron technology. Thus, we do not utilize this invariance in obtaining a more practical gate delay model given in equations 4 and 5.

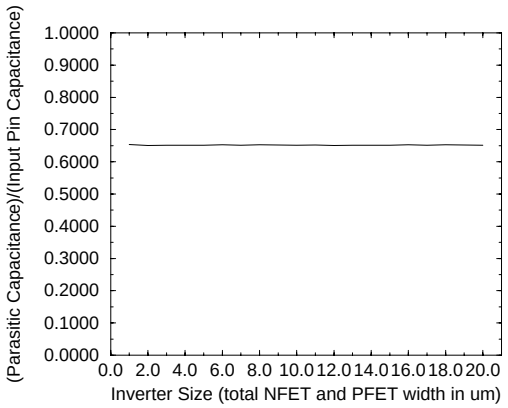

Figure 3: Variation in ratio of parasitic capacitance and input pin capacitance ($\frac{C_p}{C_{in}}$) with increasing inverter size.

In general, designers reason about delays in terms of "gain" rather than load and input pin capacitance [12][13]. In this paper, we assume that the technology library in question is designed using the semi-custom methodology [12]. Therefore we parameterize logic gates using gain instead of size, so that gates with different sizes of the same type can be modeled by the same delay equation [1]. The gain from an input pin to the output pin of a CMOS gate is defined as the ratio of gate load capacitance (C_l) to the input pin capacitance (C_{in}), i.e., gain $g = \frac{C_l}{C_{in}}$. Thus, delay t_d can be rewritten by substituting $C_l = g \cdot C_{in}$ and $C_p = K \cdot C_{in}$ in the delay equation above, i.e.:

$$t_d = k'\left(C_{in}\frac{g + K}{\mu W}\right)$$

In a CMOS gate, a rising input transition causes the gate output to be discharged to GND through the NMOS pull-down tree. Similarly, a falling input transition in a CMOS gate causes the gate output to be charged to V_{DD} through the PMOS pull-up tree. Let W_n and W_p be the width of the NMOS and PMOS devices in an inverter and let μ_n and μ_p be the mobility of electrons and holes respectively. Then, falling and rising step input delays through the inverter are given by:

$$t_{rf} = k'\left(C_{in}\frac{g + K}{\mu_n W_n}\right), \quad t_{fr} = k'\left(C_{in}\frac{g + K}{\mu_p W_p}\right)$$

The above delay equation can be applied to complex CMOS gates as well by replacing W_n by the effective N width of pull-down tree $W_{n_{eff}}$ and replacing W_p by the effective P width of pull-up tree $W_{p_{eff}}$. Thus, for a complex CMOS gate:

$$t_{rf} = k'\left(C_{in}\frac{g + K}{\mu_n W_{n_{eff}}}\right), t_{fr} = k'\left(C_{in}\frac{g + K}{\mu_p W_{p_{eff}}}\right). \quad (1)$$

Consider a two-input NAND gate shown in Figure 2(b) with effective N width $W_{n_{eff}}$ and effective P width $W_{p_{eff}}$ and P/N width

ratio $\beta = \frac{W_{p_{eff}}}{W_{n_{eff}}}$. Since the resistance through a transistor is inversely proportional to its width, each NMOS device in the pull-down tree has a width of $2 \cdot W_{n_{eff}}$ and each PMOS device in the pull-up tree has a width of $W_{p_{eff}}$. Thus, the capacitance of a NAND2 input pin is given by $C_{in} = (W_{p_{eff}} + 2 \cdot W_{n_{eff}}) \cdot L \cdot C_{ox}$. Substituting $W_{p_{eff}} = \beta \cdot W_{n_{eff}}$, $C_{in} = (W_{p_{eff}} + 2 \cdot W_{n_{eff}}) \cdot L \cdot C_{ox}$, and $k'' = k' \cdot L \cdot C_{ox}$ in rise and fall delay equations 1, we get the following rise and fall delays for a NAND2:

$$t_{rf} = k'' \frac{(\beta + 2)(g + K)}{\mu_n}, \quad t_{fr} = k'' \frac{(\beta + 2)(g + K)}{\mu_p \beta}$$

In general, a NMOS device in a complex gate is assigned a width of $M_n W_{n_{eff}}$ and a PMOS device is assigned a width of $M_p W_{p_{eff}}$, where M_n and M_p denote the NMOS and PMOS multiplication factors for a given input pin of the complex gate [12]. For example, in the case of a inverter, $M_n = 1$ and $M_p = 1$. Similarly, in the case of a NAND2, $M_n = 2$ and $M_p = 1$. Due to the non-linear nature of MOSFET resistances, the effective width of two series NMOS devices, with a width of $2 \cdot W$ each, is actually more than W. Thus, in practice, these multiplication factors are obtained from AS/X simulations[2] of gates in a given technology. Using these N and P multiplication factors, the input pin capacitance C_{in} is expressed as $(M_p \cdot W_{p_{eff}} + M_n \cdot W_{n_{eff}}) \cdot L \cdot C_{ox}$. Substituting $W_{p_{eff}} = \beta \cdot W_{n_{eff}}$, $C_{in} = (M_p \cdot W_{p_{eff}} + M_n \cdot W_{n_{eff}}) \cdot L \cdot C_{ox}$, and $k'' = k' \cdot L \cdot C_{ox}$ in rise and fall delay equations 1, we get the following rise and fall delays for a general CMOS complex gate:

$$t_{rf} = k'' \frac{(M_p \beta + M_n)(g + K)}{\mu_n}, t_{fr} = k'' \frac{(M_p \beta + M_n)(g + K)}{\mu_p \beta}$$

The delay equations above can be rewritten as:

$$\boxed{t_{rf} = k'' \left(\frac{M_n K}{\mu_n} + \frac{M_p K}{\mu_n} \cdot \beta + \frac{M_n}{\mu_n} \cdot g + \frac{M_p}{\mu_n} \cdot g\beta \right),} \quad (2)$$

$$\boxed{t_{fr} = k'' \left(\frac{M_p K}{\mu_p} + \frac{M_n K}{\mu_p} \cdot \frac{1}{\beta} + \frac{M_p}{\mu_p} \cdot g + \frac{M_n}{\mu_p} \cdot \frac{g}{\beta} \right).} \quad (3)$$

In the remainder of this paper, delay equations 2 and 3 are referred as *analytical delay model*. It is interesting to note that for a fixed P/N width ratio β, our analytical delay model above actually reduces to the linear gain-delay model, $delay = p + l \cdot gain$, employed in [14] and [6]. The analytical delay model equations 2 and 3 provide a useful understanding of delay dependence on gain, P/N width ratio, carrier mobility, and topology of CMOS gates. However, they give an over-simplified view of the CMOS gate behavior in deep-submicron technologies in addition to the fact that zero input slew has been assumed. In general, a MOSFET does not behave as a linear resistor even if driven by a step input. In addition, capacitances in MOSFETs are time and voltage dependent, i.e., they are dynamic in nature and do not have fixed values. The carrier mobility in deep-submicron technologies is modulated by high electric field effects. Also, Miller effect can cause significant deviation from the simplified view of real delays. As a result, the coefficients of gain and β variables in delay equations 2 and 3 deviate from their simplified values. In spite of these effects, we postulate that the delay dependence on gain and β retains the form of equations 2 and 3. That means for each timing arc (input pin to output pin connection) of a logic gate, the rise-fall and fall-rise delay equations can be written as:

$$\boxed{t_{rf} = a_0^{rf} + a_1^{rf} \cdot \beta + a_2^{rf} \cdot g + a_3^{rf} \cdot g\beta,} \quad (4)$$

[2] AS/X is IBM's electrical-level simulator similar to SPICE.

$$\boxed{t_{fr} = a_0^{fr} + a_1^{fr} \cdot \frac{1}{\beta} + a_2^{fr} \cdot g + a_3^{fr} \cdot \frac{g}{\beta}} \quad (5)$$

Since input slew has significant effect on gate delays, we model slew dependence of delay by using a different set of coefficients for each input slew value.

In the remainder of this paper, delay equations 4 and 5 are referred as *generalized delay model*.

Table 1: Average % error and worst case error (ps) for falling and rising transition delays from generalized delay model in comparison to AS/X simulated delays.

Cell Type	*Arc*	Fall-rise delay Error		Rise-fall delay Error	
		Average % Error	*Absolute Worst (ps)*	*Average % Error*	*Absolute Worst (ps)*
inv	A	1.5	3.7	1.6	5.2
nand2	A	0.2	1.3	0.7	3.3
nand2	B	0.1	0.7	0.3	1.6
nand3	A	0.1	0.7	0.2	2.1
nand3	B	0.1	0.9	0.2	1.3
nand3	C	0.2	1.2	0.1	0.8
nand4	A	0.1	1.2	0.1	0.5
nand4	B	0.1	1.3	0.1	0.9
nand4	C	0.1	1.3	0.1	0.8
nand4	D	0.2	1.9	0.1	0.6
nor2	A	0.3	1.4	0.4	2.6
nor2	B	0.1	0.8	0.4	1.9
nor3	A	0.8	2.5	0.2	1.9
nor3	B	0.2	1.4	0.4	2.6
nor3	C	0.2	1.4	0.6	4.7
aoi12	A1	0.1	0.7	0.1	1.6
aoi12	A2	0.1	0.5	0.1	0.9
aoi12	B	0.1	0.6	0.2	1.4
aoi21	A1	0.3	1.7	0.8	3.9
aoi21	A2	0.2	1.3	0.8	3.9
aoi21	B	0.3	0.9	0.2	1.9
aoi22	A1	0.3	1.4	0.1	1.3
aoi22	A2	0.1	1.2	0.1	0.7
aoi22	B1	0.1	1.2	0.3	1.8
aoi22	B2	0.1	0.8	0.3	1.9
oai12	A1	0.1	0.5	0.2	1.5
oai12	A2	0.1	0.7	0.3	1.3
oai12	B	0.1	0.5	0.1	0.6
oai21	A1	0.2	1.0	0.2	1.4
oai21	A2	0.2	1.2	0.3	1.6
oai21	B	0.1	0.5	0.3	2.5
oai22	A1	0.1	0.4	0.1	0.9
oai22	A2	0.1	0.6	0.2	1.1
oai22	B1	0.1	0.5	0.1	0.9
oai22	B2	0.1	0.7	0.2	1.2

For every logic gate in a high-performance standard cell library[3], the delay vs. gain and β data of each timing arc are obtained using AS/X simulation in a $0.12\mu m$ L_{eff} deep-submicron CMOS technology for the input slew values 50, 100, 150, 200, 250, 300, 350 picoseconds. The simulations were performed for nominal technology parameters, i.e., a supply voltage of 1.8V and a temperature of 75°C. Typically, designers limit maximum gain allowed for any give CMOS cell to 10 in order to avoid slew limit violations; and a gain range of 1 to 10 is considered to be representative of loading conditions in high-performance circuits [1][12]. In addition, designers limit the P/N width ratio (i.e., β) of logic gates between 1 and 4 in order to avoid noise margin violations [12][13]. Thus, while performing simulations, the gain was varied from 1 to 10 in increments of 1 and the β was varied from 1 to 4 in increments of 0.3. A least square fit is used to extract the coefficients of the delay equations (equations 4 and 5) for each set of data. For each slew value, the generalized delay equations 4 and 5 model the delay behavior with very high degree of accuracy, for all the CMOS gates in the library over the entire range of gain, β, and slew. Table 1 shows the average % error and worst case delay error in picoseconds derived by comparing delay values

[3] The logic gates considered are: inverter, nand2, nand3, nand4, nor2, nor3, aoi21, aoi12, aoi22, oai21, oai12, and oai22.

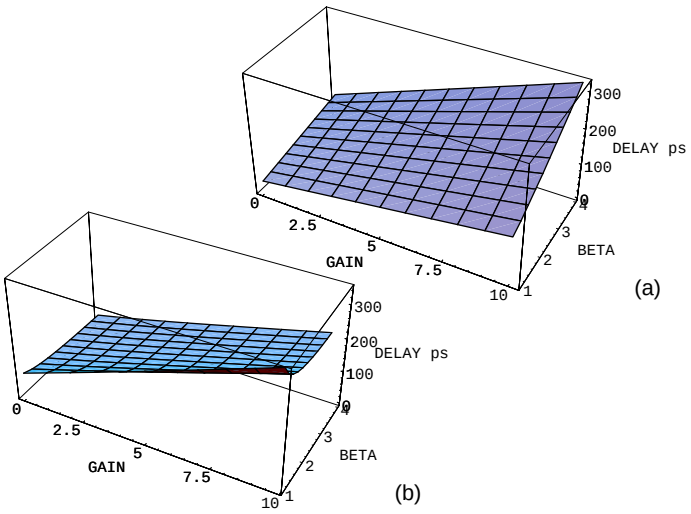

Figure 4: Dependence of delay (simulated) on gain and β for an AOI22, pin A1 to output Y (a) Rising input transition (b) Falling input transition.

from generalized delay model with respect to AS/X simulated delays (for a representative slew of 150 picoseconds). It can be seen from Table 1 that the average error in all cases is less than 1.6%. Surprisingly, as shown in Table 1 our delay equations yield even higher degree of accuracy for complex gates such as AOIs, OAIs, and high fanin NANDs and NORs. Figure 4(a) shows the plot of simulated rising input A1 delay of an AOI22 as a function of gain and P/N width ratio β for a fixed slew of 150 picoseconds. As discussed above, this delay can be fitted using the delay equation $t_{rf} = 0.0299 + 0.0176 \cdot \beta + 0.0047 \cdot g + 0.00472 \cdot g\beta$, resulting in a maximum error of 1.3 picoseconds and an average error of 0.1% in comparison to the simulated delay results shown in Figure 4(a). Similarly, Figure 4(b) shows the plot of simulated falling input A1 delay of an AOI22 as a function of gain and β for a fixed slew of 150 picoseconds. This delay can be fitted using the delay equation $t_{fr} = 0.0480 + 0.0427 \cdot \frac{1}{\beta} + 0.0133 \cdot g + 0.0138 \cdot \frac{g}{\beta}$, resulting in a maximum error of 1.4 picoseconds and an average error of 0.3% in comparison to the simulated delay results shown in Figure 4(b).

In the following section, we utilize the delay models derived above to formulate the P/N width optimization problem along a general path in CMOS logic circuits; and develop a theoretical framework through which library designers can determine "optimal" P/N width ratio for each logic gate in their high-performance standard cell library.

3 OPTIMIZING P/N WIDTH RATIO β

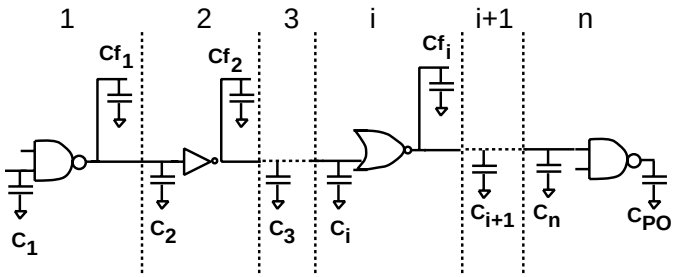

Figure 5: A general circuit path with fanouts.

In this section, we focus on the path delay optimization problem. Consider a general circuit path shown in Figure 5 where:

- C_i denotes the pin capacitance of the on-path input pin of the i^{th} stage.
- Cf_i denotes the off-path fanout capacitive load driven by the i^{th} stage.
- β_i denotes the P/N width ratio of the i^{th} stage.
- C_{PO} denotes the capacitive load of the last (n^{th}) stage in the path.

For the path shown in Figure 5, the gain of each CMOS stage can be written as:

$$g_1 = \frac{Cf_1 + C_2}{C_1}, g_2 = \frac{Cf_2 + C_3}{C_2}, \ldots g_i = \frac{Cf_i + C_{i+1}}{C_i}, \ldots$$
$$\ldots g_{n-1} = \frac{Cf_{n-1} + C_n}{C_{n-1}}, g_n = \frac{C_{PO}}{C_n}$$

The rising input and the falling input delay along the path are given by:

$$T_r = t_{rf_1} + t_{fr_2} + t_{rf_3} + \ldots, \quad T_f = t_{fr_1} + t_{rf_2} + t_{fr_3} + \ldots$$

where t_{rf_i} and t_{fr_i} are the rise-fall and fall-rise delay equations for the i^{th} gate given by equations 4 and 5 respectively. The average of rising and falling input delays along the path is

$$T_{av} = \frac{T_r + T_f}{2} = \frac{1}{2} \sum_{i=1}^{N} (t_{rf_i} + t_{fr_i}).$$

The path delay optimization problem is stated as follows:

Given a path of CMOS logic gates in a general logic network, find an assignment of P/N width ratio and gain value to each logic gate such that the average of rising and falling input delays along the path is minimized under the constraint that C_1 is less than or equal to the primary input capacitance limit C_{PI}.

We assert that the minimum delay solution must saturate the primary input capacitance limit, i.e., $C_1 = C_{PI}$ at the minimum. If the minimum occurred at $C_1 < C_{PI}$, we could increase C_1 to C_{PI} and reduce g_1. Thus the delay of the first stage could be reduced and we would arrive at a solution with path delay less than the minimum, which is a contradiction. In terms of the gain variables, the primary input capacitance constraint translates to

$$C_{PO} - C_{PI} \prod_{i=1}^{n} g_i + Cf_1 \prod_{i=2}^{n} g_i + \ldots Cf_j \prod_{i=j}^{n} g_i + \ldots Cf_{n-1} \cdot g_n + Cf_n = 0 \quad (6)$$

Therefore the cost function of the path delay optimization problem can be formulated as

$$T = T_{av} - \lambda \cdot f(g_1, \ldots g_n, Cf_1, \ldots Cf_n, C_{PO}, C_{PI}),$$

where λ represents the Lagrange multiplier and the function f is the left hand side of equation 6. Although the gain and the P/N width ratio variables are not independent of each other, we show in Appendix A that

$$\frac{\partial T}{\partial \beta_i} = 0, \quad \frac{\partial T}{\partial g_i} = 0. \quad (7)$$

still holds at the minimum. Since $f(g_1, \ldots g_n, Cf_1, \ldots Cf_n, C_{PO}, C_{PI})$ is not explicitly a function of β

$$\frac{\partial T}{\partial \beta_i} = \frac{\partial T_{av}}{\partial \beta_i}. \quad (8)$$

In the following subsections, we apply equations 7 and 8 to to obtain the optimal P/N width ratio of each gate in the path delay optimization problem.

3.1 Minimizing delay under analytical delay model

The analytical delay equations for rising and falling input gate delay is given by equations 2 and 3. In this subsection, we focus on the i^{th} stage of the path where the pin multipliers M_p and M_n refer to those of the i^{th} stage. From equations 7 and 8 we know that at the point of minimum average delay $\frac{\partial T_{av}}{\partial \beta_i} = 0$. Thus:

$$\frac{k''(g_i + K_i)}{2}\left(M_p \cdot \left(\frac{1}{\mu_n} + \frac{1}{\beta_i \cdot \mu_p}\right) - (M_p\beta_i + M_n) \cdot \frac{1}{{\beta_i}^2 \cdot \mu_p}\right) = 0$$

Solving for β_i in the above equation, we obtain a surprisingly simple result for P/N width ratio of a gate at minimum delay, i.e.:

$$\boldsymbol{\beta_i = \sqrt{\frac{M_n \mu_n}{M_p \mu_p}}}. \tag{9}$$

The significance of this result is that the optimal P/N width ratio of any CMOS gate depends only on the gate type and the corresponding timing arc but is entirely independent of the structure of the circuit path. For example, $M_n = 1$ and $M_p = 1$ for an inverter, $M_n = N$ and $M_p = 1$ for an N-input NAND and $M_n = 1$ and $M_p = N$ for an N-input NOR. It follows that the optimal P/N width ratios for an inverter, an N-input NAND and an N-input NOR are $\sqrt{\frac{\mu_n}{\mu_p}}$, $\sqrt{\frac{N\mu_n}{\mu_p}}$ and $\sqrt{\frac{\mu_n}{N\mu_p}}$ respectively. Thus, the optimal P/N width ratio of a NOR gate is always less than that of an inverter while the optimal P/N width ratio of a NAND gate is always larger than that of an inverter. As discussed in section 2, the analytical delay model over-simplifies the realities of device behavior in deep-submicron technologies. In the next subsection we use the generalized delay equations (equations 4 and 5) to obtain more realistic optimal P/N width ratio for minimum average delays.

3.2 Minimizing delay under generalized delay model

The generalized delay equations for rising and falling input gate delay is given by equations 4 and 5. Again we will focus on the i^{th} stage of the path. So for the rest of the subsection, the coefficients (a's) refer to those of the i^{th} stage. As in the previous subsection, the necessary condition at the minimum is given by $\frac{\partial T_{av}}{\partial \beta_i} = 0$. Thus, at the minimum

$$a_1^{rf} + a_3^{rf} \cdot g_i - a_1^{fr} \cdot \frac{1}{\beta_i^2} - a_3^{fr} \cdot \frac{g_i}{\beta_i^2} = 0. \tag{10}$$

Comparing analytical delay model (equations 2 and 3) and the generalized delay model (equations 4 and 5), we infer that if the generalized model followed the analytical delay model, we will have

$$\frac{a_1^{fr}}{a_1^{rf}} = \frac{a_3^{fr}}{a_3^{rf}}$$

and the optimal P/N width ratio would be $\sqrt{a_1^{fr}/a_1^{rf}}$. Let us represent the ratio a_1^{fr}/a_1^{rf} by a constant γ, i.e.:

$$\gamma = \frac{a_1^{fr}}{a_1^{rf}}, \text{ and let us define } \overline{a}_3^{rf} \text{ by } \frac{a_3^{fr}}{\overline{a}_3^{rf}} = \gamma.$$

We define another constant Δ that measures the deviation of the generalized delay model from its ideal analytical behavior

$$a_3^{rf} = \Delta + \overline{a}_3^{rf}$$

Thus Δ reduces to 0, if the generalized delay model follows the analytical delay model (i.e., $a_1^{fr}/a_1^{rf} = a_3^{fr}/a_3^{rf}$).

Table 2: Variation of Optimal P/N Width Ratio over entire gain (1-10) and slew (50ps-350ps) range.

Cell Type	*Arc*	Optimal P/N Ratio β			*Max % gate delay variation for using β_r w.r.t using β_l or β_u*
		Lower bound β_l	*Upper bound β_u*	*Recommended β_r*	
inv	A	1.30	1.48	1.41	0.1
nand2	A	1.77	2.26	2.03	0.3
nand2	B	2.41	2.54	2.47	0.0
nand3	A	1.93	2.74	2.36	0.7
nand3	B	2.65	2.96	2.87	0.1
nand3	C	3.18	3.38	3.29	0.0
nand4	A	2.07	3.07	2.59	0.9
nand4	B	2.85	3.35	3.13	0.1
nand4	C	3.58	3.69	3.66	0.0
nand4	D	3.87	4.16	4.05	0.0
nor2	A	1.10	1.32	1.16	0.2
nor2	B	0.91	1.03	0.96	0.0
nor3	A	0.94	1.21	1.03	0.4
nor3	B	0.82	0.86	0.83	0.0
nor3	C	0.70	0.80	0.74	0.1
aoi12	A1	1.45	1.62	1.53	0.0
aoi12	A2	1.72	1.89	1.80	0.0
aoi12	B	0.78	0.94	0.83	0.3
aoi21	A1	1.12	1.49	1.28	0.5
aoi21	A2	1.46	1.59	1.51	0.0
aoi21	B	1.16	1.48	1.28	0.3
aoi22	A1	1.57	1.67	1.62	0.0
aoi22	A2	1.77	1.99	1.89	0.0
aoi22	B1	1.00	1.38	1.14	0.8
aoi22	B2	1.26	1.47	1.33	0.2
oai12	A1	1.70	1.75	1.72	0.0
oai12	A2	1.35	1.60	1.46	0.1
oai12	B	2.61	2.93	2.81	0.1
oai21	A1	1.86	2.28	2.08	0.3
oai21	A2	1.74	1.80	1.77	0.0
oai21	B	1.56	2.13	1.83	0.5
oai22	A1	1.57	1.66	1.61	0.0
oai22	A2	1.27	1.56	1.39	0.3
oai22	B1	1.98	2.52	2.29	0.4
oai22	B2	1.86	2.06	1.99	0.1

Substituting γ and Δ in equation 10, we get:

$$a_1^{rf}\left(1 - \frac{\gamma}{\beta_i^2}\right) + \Delta \cdot g_i + \overline{a}_3^{rf} \cdot g_i \cdot \left(1 - \frac{\gamma}{\beta_i^2}\right) = 0$$

Solving for β_i in the above equation yields:

$$\boldsymbol{\beta_i = \sqrt{\gamma}\left(1 + \frac{\Delta \cdot g_i}{a_1^{rf} + \overline{a}_3^{rf} \cdot g_i}\right)^{-\frac{1}{2}}} \tag{11}$$

This result shows that in practice, the optimal P/N width ratio at which T_{av} is minimum depends on the gain distribution along the path. The amount of dependence is a function of Δ, a measure of the deviation of the generalized delay model from the analytic delay model. However, the experimental results in the next section reveal that the variation of "optimal β" (equation 11) over the entire gain and slew design range has a negligible impact in the minimum average delay.

4 EXPERIMENTAL RESULTS

In this section, we discuss the effect of gain and slew variation on optimal P/N width ratios and average delays of various logic gates in a $0.12\mu m$ L_{eff} deep-submicron CMOS technology. We show that using optimal P/N width ratios for various gates in the standard cell library can significantly improve the timing performance of high-performance CMOS circuits. Based on the delay coefficients extracted in section 2, we use equation 11 to compute the optimal P/N width ratio as a function of gain for each input slew value.

Table 2 shows the upper and lower bounds of the optimal P/N width ratio (β_u and β_l respectively) over the entire gain-slew range

(i.e., gain values between 1 and 10 and slew values between 50 and 350 picoseconds). Typically, in critical region of optimized CMOS circuits, the most frequent gain occurs in the gain interval of 2 to 3 [1][12]. In addition, in optimized CMOS circuits, typical slew values range from 100 to 200ps. Thus, we select a gain of 3 and input slew of 150ps to be the most frequently occurring gain, slew data values. The fifth column in Table 2 gives the recommended P/N width ratio which is the optimal P/N width ratio at gain 3 and input slew 150 picoseconds. The last column in Table 2 gives the worst case percentage delay error incurred over the entire gain-slew range as a result of using the recommended P/N width ratio instead of the optimal P/N width ratio at that gain and slew value. Although the recommended value of optimal P/N width ratio (β_r) is based on fixed gain and slew values, the % error in delay due to selecting β_r instead of selecting the β for that specific gain and slew value is negligible. We illustrate this point further in Figure 6(a) which shows the delay variation over the lower and upper bounds of optimal P/N width ratio $[\beta_l, \beta_u]$ for a specific timing arc (input pin A, output pin Y) for the NANDs and NORs. The delay curves are all nearly flat, reinforcing the data in Table 2. However, timing arc delays are not totally insensitive to β variations. In Figure 6(b) we extend the plot of Figure 6(a) to a much larger range of β. It is clear that if a P/N width ratio is chosen in the steep part of the curves, the gate delay can be substantially larger than the optimal one. Therefore it is particularly important to select a P/N width ratio close to the value recommended (β_r) in Table 2.

Table 3: Impact of optimal P/N ratio on real designs

Design	Standard cell library with:			
	P/N width ratio for balanced rise/fall delay		P/N width ratio equal to optimal value (i.e., β_r)	
	Worst Slack(ps)	Area	Worst Slack(ps)	Area
d1	-640.36	20844	-562.15	21061
d2	-646.04	11509	-498.16	11678
d3	-347.87	10420	-326.45	10532
d4	-466.02	9519	-381.89	9580
d5	-421.59	6588	-401.69	6623
d6	-453.34	5255	-415.38	5425
d7	-438.78	3667	-402.51	3927
d8	-124.19	3028	-110.11	3170
Total	-3538.19	70830	-3098.34	71996

We now investigate the impact our theoretical results have on real designs. In general, for every logic gate, a library cell with β value that yields balanced rise and fall transition delays is always present in the standard cell library. Although a β value yielding equal rise and fall transition delays is optimum for noise, most often, it is not optimum for speed[4]. We experimented with several design partitions from control logic of a 800MHz microprocessor design in $0.12\mu m$ L_{eff} deep-submicron technology. Each design partition was synthesized twice. First, we synthesized the design with a standard cell library that contains logic cells with P/N width ratios yielding balanced rise/fall transition gate delays. The P/N width ratios corresponding to balanced rise/fall delays for various gates in this library were: inverter (2.7), nand2 (3.6), nand3 (4.2), nand4(5.0), nor2 (1.95), nor3 (1.5), aoi21 (2.7), aoi12 (2.7), aoi22 (2.7), oai21 (2.7), oai12 (2.78), and oai22 (2.58). Subsequently, we synthesized the designs using the library with optimal P/N width ratios shown in Table 2. For almost all design partitions, the library with optimal P/N width ratio yielded significantly better worst case delay (slack) with almost no penalty in area. Results on some of the design partitions are shown in Table 3. As shown, on an average, the worst slack improved by 12.5% with only a 1.6% increase in area by using optimal P/N width ratio cells as compared to balanced rise/fall delay cells. Thus, it is crucial for library designers to include library cells with optimal P/N width ratios in their high-performance design library.

[4] Some critical circuit paths may yield better worst case delay with β values different from the optimal ones due to different rising and falling transition arrival times assertions on primary inputs.

5 CONCLUSION

In this paper, we developed a theoretical framework through which library designers can determine "optimal" P/N width ratio for each logic gate in their high-performance standard cell library. This theoretical framework utilizes new gate delay models that explicitly represent the dependence of delay on P/N width ratio and load. These delay models yield highly accurate delay for CMOS gates in a $0.12\mu m$ L_{eff} deep-submicron technology. For each timing arc of a set of commonly used cells in a high-performance standard cell library (in $0.12\mu m$ L_{eff} CMOS technology), we derived a P/N width ratio that gives practically optimal delay within a normal range of input slew and output load. Experimental results with real designs demonstrated that selection of good P/N width ratios in standard cell library is crucial for achieving higher-performance. It is well known that delay trades off with noise margin through varying the P/N width ratio. Using our theoretical framework to study noise issues is a natural extension of this work.

Appendix A

The gain of the i^{th} stage depends on the P/N width ratio of the i^{th} and $i+1^{th}$ stage through the pin capacitances. However, we can formulate the path delay minimization problem in terms of independent variables such as the effective N-fet widths (W_i's) and the P/N width ratios (β_i's). Then g_i is a function of W_i, β_i, W_{i+1}, β_{i+1}. For example, under the analytic delay model

$$g_i = \frac{Cf_i + M_{n_{i+1}} W_{i+1} + M_{p_{i+1}} \beta_{i+1} W_{i+1}}{M_{n_i} W_i + M_{p_i} \beta_i W_i},$$

Let the path delay cost function T be formulated as a function of g's and β's, $F(\beta_1, \ldots \beta_n, g_1, \ldots g_n)$, and equivalently as a function of W's and β's, $F'(\beta_1, \ldots \beta_n, W_1, \ldots W_n)$. That means F and F' can be transformed into one another by a change of variable. The partial derivatives of the two equivalent functions are related by the following pair of equations

$$\frac{\partial F'}{\partial \beta_i} = \frac{\partial F}{\partial \beta_i} + \frac{\partial F}{\partial g_i} \cdot \frac{\partial g_i}{\partial \beta_i} + \frac{\partial F}{\partial g_{i-1}} \cdot \frac{\partial g_{i-1}}{\partial \beta_i} \tag{12}$$

$$\frac{\partial F'}{\partial W_i} = \frac{\partial F}{\partial g_i} \cdot \frac{\partial g_i}{\partial W_i} + \frac{\partial F}{\partial g_{i-1}} \cdot \frac{\partial g_{i-1}}{\partial W_i} \tag{13}$$

where i runs from 1 to n. Since β and W are independent variables the minimum of F' occurs at

$$\frac{\partial F'}{\partial \beta_i} = 0, \quad \frac{\partial F'}{\partial W_i} = 0.$$

for each i according to the Kuhn-Tucker condition. The Jacobian of the change of variable is non-zero, therefore at the minimum of F

$$\frac{\partial F}{\partial \beta_i} = 0, \quad \frac{\partial F}{\partial g_i} = 0 \text{ also hold for each } i.$$

References

[1] F. Beeftink, P. Kudva, D. Kung, and L. Stok. Gate-Size Selection for Standard Cell Libraries. In *Proc. of the International Conference on Computer-Aided Design*, pages 545–550, 1998.

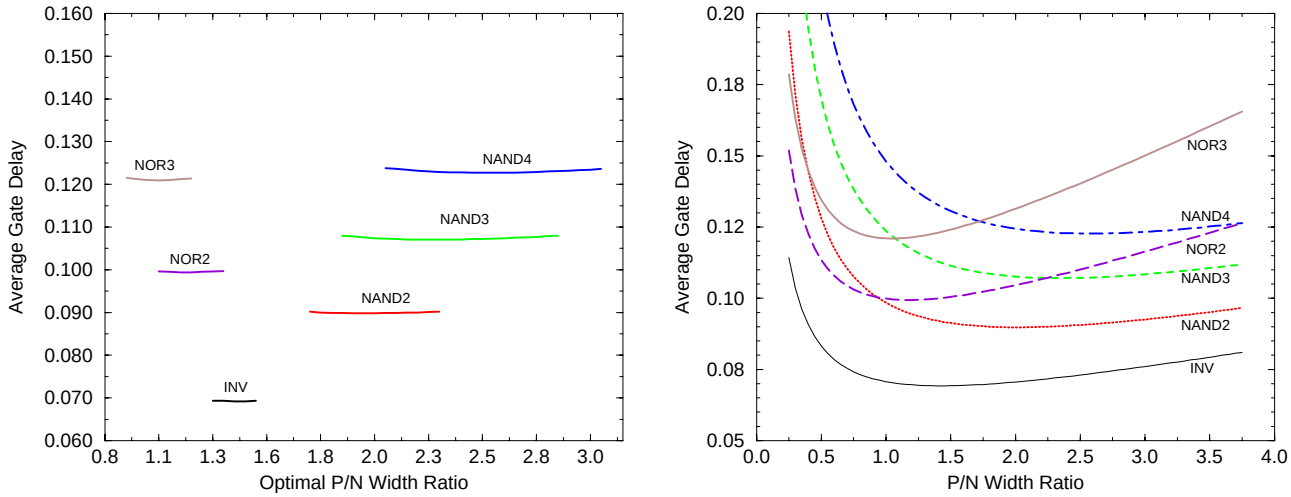

Figure 6: (a) Variation of Delay with optimal P/N width ratio between lower and upper bounds (i.e., β_l and β_u) (b) Variation of Delay with a wider range of P/N width ratios at gain = 3.

[2] M. Berkelaar and J. Jess. Gate Sizing in MOS Digital Circuits with Linear Programming. In *Proc. of the European Design Automation Conference (EDAC)*, pages 217–221, 1990.

[3] O. Coudert, R. Haddad, and S. Manne. New Algorithms for Gate Sizing: A Comparitive Study. In *Proc. of the Design Automation Conference (DAC)*, pages 734–739, 1996.

[4] W. C. Elmore. The Transient Response of Damped Linear Networks with Particular Regard to Wideband Amplifiers. *Journal of Applied Physics*, 19(1):55–63, 1948.

[5] C. Fisher, R. Blankenship, J. Jensen, T. Rossman, and K. Svilich. Optimization of Standard Cell Libraries for Low Power, High Speedm or Minimal Area Designs. In *Proc. of the Custom Integrated Circuits Conference*, pages 493–496, 1996.

[6] J. Grodstein, H. Harkness, B. Grundmann, and Y. Watanabe. A Delay Model for Logic Synthesis of Continuously-Sized Networks. In *Proc. of the International Conference on Computer-Aided Design*, pages 458–462, 1995.

[7] R. Haddad, L. Van Ginneken, and N. Shenoy. Drive Selection for Library Design. In *Proc. of the International Workshop on Logic Synthesis*, 1997.

[8] K. Keutzer and K. Scott. Improving Cell Libraries for Synthesis. In *Proc. of the International Workshop on Logic Synthesis*, 1993.

[9] T. Mozdzen. Design Methodology for A 1.0μm Cell-Based Library Efficiently Optimized for Speed and Area. In *Proc. of the IEEE International ASIC Conference and Exhibit*, pages P12(3.1)–P12(3.5), 1990.

[10] D. A. Pucknell and K. Eshraghian. *Basic VLSI Design : Systems and Circuits*. Prentice Hall, Sydney, Australia, 1988. Practical Realities and Ground Rules: Optimization of NMOS and CMOS Inverters.

[11] J. Rubinstein, P. Penfield, and M. A. Horowitz. Signal Delay in RC Tree Networks. *IEEE Trans. on Computer-Aided Design*, 2(3):202–210, 1983.

[12] K. Shepard and et al. Design Methodology for the S/390 Parallel Enterprise Server G4 Microprocessor. *IBM Journal of Research and Development*, 41:515–547, 1997.

[13] L. Sigal and et al. Circuit Design Techniques for the High-performance CMOS IBM S/390 Parallel Enterprise Server G4 Microprocessor. *IBM Journal of Research and Development*, 41:489–501, 1997.

[14] I. Sutherland and R. Sproull. The Theory of Logical Effort: Designing for Speed on the Back of an Envelope. In *Advanced Research in VLSI*, University of California at Santa Cruz, 1991.

[15] C. Weitong, S.S. Sapatnaker, and I.N. Hajj. Delay and Area Optimization for Discrete Gate Sizes under Double-Sided Timing Constraints. In *Proc. of the Custom Integrated Circuits Conference*, pages 9.4.1–9.4.4, 1993.