CMOS Gate Sizing under Delay Constraint

Alexandre Verle, Xavier Michel, Philippe Maurine, Nadine Azemard, Daniel Auvergne

CMOS Gate Sizing under Delay Constraint

A. Verle, X. Michel, P. Maurine, N. Azémard, and D. Auvergne

LIRMM, UMR CNRS/Université de Montpellier II, (C5506),
161 rue Ada, 34392 Montpellier, France

Abstract. In this paper we address the problem of delay constraint distribution on a CMOS combinatorial path. We first define a way to determine on any path the reasonable bounds of delay characterizing the structure. Then we define two constraint distribution methods that we compare to the equal delay distribution and to an industrial tool based on the Newton-Raphson like algorithm. Validation is obtained on a 0.25µm process by comparing the different constraint distribution techniques on various benchmarks.

1 Introduction

The goal of gate sizing is to determine optimum sizes for the gates in order that the path delay respect the constraints with the minimum area/power cost. Another parameter to consider is the feasibility of the constraint imposed on a given path. For that, ways must be found to get indicators for exploring the design space, allowing to select one among the available optimization alternatives such as sizing, buffering or technology remapping. The target of this paper is twofold: defining the delay bounds of a given path and determining a way for distributing a delay constraint on this path with the minimum area/power cost.

The problem of transistor sizing has been widely investigated using non linear programming techniques [1] or heuristics based on simple delay models [2]. Recently, in a pedagogical application [3] of the τ model, Sutherland [4], describing the gate delay as the product of electrical and logical efforts, proposed to minimize the delay on a path by imposing an equal effort that is a constant delay on all the elements of the path.

This way to select cell sizes can be proven mathematically exact [3] for a fanout-free path constituted of ideal gates (no parasitic capacitance nor divergence branches). However this evenly budget distribution is far to be the optimal one with respect to delay and area for a real path, on which divergence branches and routing capacitance are not negligible. Starting from the definition of the design space in terms of minimum and maximum delay permissible on a given path, we propose in this paper a design space exploration method allowing an area/power efficient distribution of constraint on a combinatorial path.

The delay bound determination and the constraint distribution method are based on a realistic delay model [5] that is input slope dependent and able to distinguish between falling and rising signals. This model is shortly presented in part 2. In part 3

J.J. Chico and E. Macii (Eds.): PATMOS 2003, LNCS 2799, pp. 60–69, 2003.

we give a method for defining delay bounds on a path. Different approaches for distributing a delay constraint are considered in part 4 and compared in part 5 on different benchmarks of increasing complexity. We finally conclude in part 6.

2 Gate Delay Modeling

As previously mentioned sizing at the physical level imposes the use a realistic delay computation that must consider a finite value of the gate input transition time. As developed in [5] we introduce the input slope effect and the related input-to-output coupling in the model as

$$\begin{aligned} t_{HL}(i) &= \frac{v_{TN}}{2}\tau_{INLH}(i-1) + (1+\frac{2C_M}{C_M+C_L})t_{HLstep}(i) \\ t_{LH}(i) &= \frac{v_{TP}}{2}\tau_{INHL}(i-1) + (1+\frac{2C_M}{C_M+C_L})t_{LHstep}(i) \end{aligned} \quad (1)$$

where $v_{TN,P}$ represents the reduced value (V_T/V_{DD}) of the threshold voltage of the N,P transistors, $\tau_{INHL,LH}$ is the duration time of the input signal, taken to be twice the value of the step response of the controlling gate. C_M is the coupling capacitance between the input and output nodes. Indexes (i), (i-1) specify the switching and the controlling gates, respectively.
Following [2] the step response of each edge is defined by the time interval necessary to load (unload) the gate output capacitance under the maximum current available in the structure:

$$\begin{aligned} t_{HLstep} &= \frac{C_L \cdot \Delta V}{I_N} \\ t_{LHstep} &= \frac{C_L \cdot \Delta V}{I_P} \end{aligned} \quad (2)$$

Following the elegant model of [3] the evaluation of this step response on logic gates supplies a general expression given by:

$$\begin{aligned} t_{HLstep} &= \tau \cdot S_{HL} \cdot \frac{C_L}{C_{IN}} \\ t_{LHstep} &= \tau \cdot S_{LH} \cdot \frac{C_L}{C_{IN}} \end{aligned} \quad . \quad (3)$$

where τ is a time unit characterizing the process. C_{IN}, the gate input capacitance, is defined in terms of configuration ratio ($k=W_P/W_N$) between the P and N transistors. For simplicity, the S factors (logical effort of [3]) include all the current capability difference between the pull up and pull down equivalent transistors, they are configuration ratio dependent and characterize for each edge, the ratio of current available in an inverter and a gate of identical size.

Then considering an array of gates, the delay path can easily be obtained from (1) and (3) as a technology independent posynomial representation:

$$\frac{t_{HL,LH}}{\tau} = \theta = S'_1 \cdot \frac{C_2 + C_{P1}}{C_1} + \cdots + S'_{i-1} \cdot \frac{C_i + C_{Pi-1}}{C_{i-1}} + S'_i \cdot \frac{C_{i+1} + C_{Pi}}{C_i} + \cdots + S'_n \cdot \frac{C_L}{C_n} \qquad (4)$$

where the S_i' include the logical effort and the input ramp effect, C_i represents the input capacitance of the gate and C_{pi} the output node total parasitic capacitance, including the interconnect and branching load.

3 Delay Bound Definition

We consider realistic combinatorial paths on which two parameters are known and imposed:

- the output load capacitance of the last gate, that is determined by the input capacitance of the next register,
- the input capacitance of the first gate imposed by the loading conditions of the input register.

In that conditions the path delay is bounded. These bounds can be determined, considering that the delay of a path (4) is a convex function of the gate input capacitance. This is illustrated in Fig.1, where we represent the path delay sensitivity to the transistor sizing of a combinatorial path constituted of 13 gates.

As shown the delay value decreases from a maximum value, obtained when all the transistors have the minimum size, to a minimum value that will be determined below. Note that the maximum delay value is a „reasonable" one, it is very easy to get a much greater value by loading minimum gates with high driving capacity one. This curve illustrates what we define by exploring the design space:

- near the maximum value Θ_{Max} of the delay the path sensitivity to the gate sizing is very important, a small variation of the gate input capacitance results in a large change in delay,
- at the contrary near the minimum Θ_{Min} the sensitivity is becoming very low and in that range any delay improvement is highly area/power expensive.

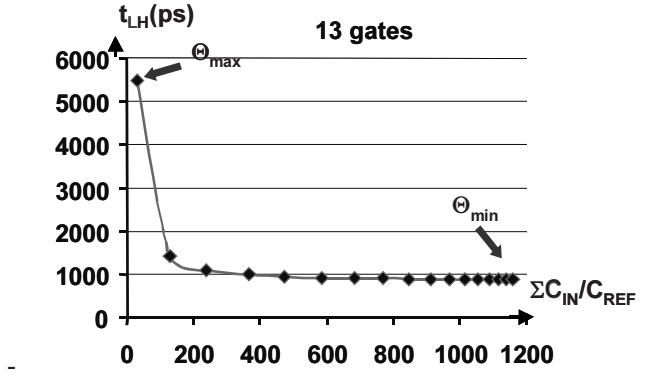

Fig. 1. Illustration of the sensitivity of the path delay to the gate sizing.

Evaluating the feasibility of a delay constraint Θ_c imposes to compare its value to the preceding bounds. If the Θ_c value is closed to the maximum Θ_{Max} the constraint satisfaction will be obtained at reasonable cost by transistor sizing otherwise it would be more profitable to reconfigure the logic or to insert buffers [6]. Let us define these bounds. As previously mentioned we consider for Θ_{Max} the „reasonable“ value obtained when all the gates are implemented with transistors of minimum size. For the minimum bounds we just use the posynomial property [7] of (6). Canceling the derivatives of (4) with respect to the gate input capacitances C_i we obtain a set of link equations such as:

$$\begin{aligned} S'_{i-1} \cdot \frac{C_i}{C_{i-1}} - S'_i \cdot \frac{C_{i+1} + C_{Pi}}{C_i} = 0 \\ S'_i \cdot \frac{C_{i+1}}{C_i} - S'_{i+1} \cdot \frac{C_{i+2} + C_{Pi+1}}{C_{i+1}} = 0..... \end{aligned} \tag{5}$$

Cell sizes can then be selected to match the minimum delay, by visiting all the gates in a topological order, starting from the output, such as:

$$C_i^2 = \frac{S'_i}{S'_{i-1}} \cdot C_{i-1} \cdot (C_{i+1} + C_{Pi}) \tag{6}$$

This results in a set of n linked equations that can be easily solved by iterations from an initial solution that considers C_{i-1} known and equal to a reference value C_{REF} that can be set to the minimum value available in the library or any other one.

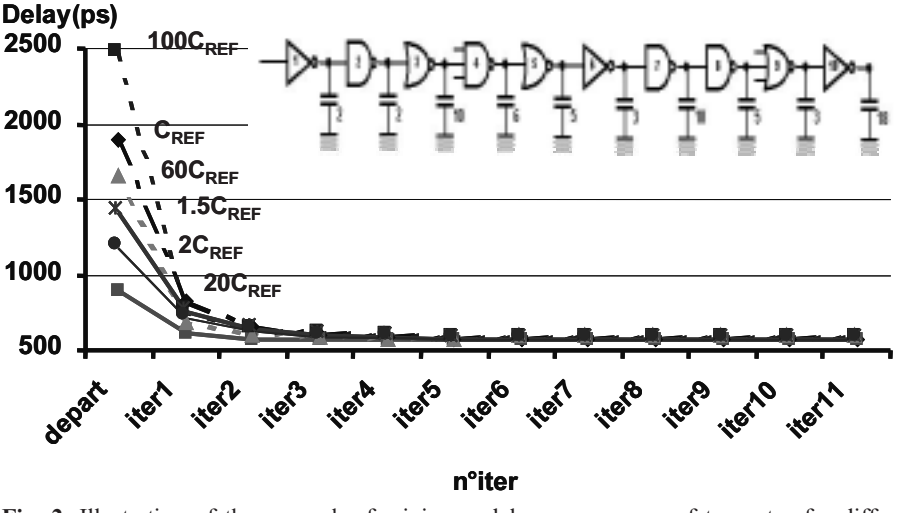

Fig. 2. Illustration of the research of minimum delay on an array of ten gates for different values of the initial reference capacitance; the output load of each gate on the array is given in unit of C_{REF}.

As shown in Fig.2, whatever is the value of the initial C_{i-1} controlling capacitance, only few iterations are required to obtain a fast convergence to the minimum delay value achievable on the array.

4 Constraint Distribution

Determining the possible bounds of delay for a given path topology, the next step is to evaluate the feasibility of a constraint to be imposed on a path. The theory of constant effort or constant delay [4,8] provides an easy way to select the cell size for each stage but for real configuration it is far from the optimum and often results in oversized structures. For that we propose two techniques for the gate size selection in order to satisfy a constraint that we will compare in the next part to the constant delay method.

To define the first method we consider that imposing equal delay to the gates with an important value of the logical effort (S'_i), will result in an important over-sizing of these complex gates. The determination of the lowest delay bound directly provides the corresponding delay distribution on the path, that appears to be the fastest one. So we can use this distribution to define for each gate a weight or gain Θ_i relative to this distribution $\Theta_{Min}=\Sigma\Theta_{Mini}$. In that case we propose to distribute the delay constraint Θc on a path using a weight defined on the minimum delay distribution as:

$$\theta_i = \frac{\theta_{Mini} \cdot \theta_c}{\sum_i \theta_{Mini}} \tag{7}$$

Then processing backward from the output of the path, this directly gives, for each gate, the value of Θ_i that determines from (1,3) the size of the corresponding gate.

The second method of equal sensitivity is directly deduced from (5). Instead to search for the minimum we impose the same path delay sensitivity to the sizing, by solving:

$$\begin{aligned} S_{i-1} \cdot \frac{1}{C_{i-1}} - S_i \cdot \frac{C_{i+1} + C_{Pi}}{C_i^2} &= a \\ S_i \cdot \frac{1}{C_i} - S_{i+1} \cdot \frac{C_{i+2} + C_{Pi+1}}{C_{i+1}^2} &= a..... \end{aligned} \tag{8}$$

where "a" is a constant with negative value, representing the slope to the curve of Fig.3, that represents the variation of the delay between the bounds previously defined. Following the procedure used for the first method, the size of the gates is obtained from the iterated solution of (8) using as initial solution the sizing for the maximum delay value (all gates sized at C_{REF}). The different points on the curve of Fig.3 have been obtained from (8), varying the value of „a“, until „a“ = 0 to get the minimum.

As expected, for a given value of the sensitivity „a“ to the sizing, this curve represents the locus of the minimum delay solutions. Thus, varying the „a“ value gives the possibility to explore the design space and to determine the minimum area sizing condition satisfying the delay constraint. As shown in the figure the results are compared with the solutions obtained from an industrial optimization tool (Amps from Synopsys) that gives nearly equivalent exploration.

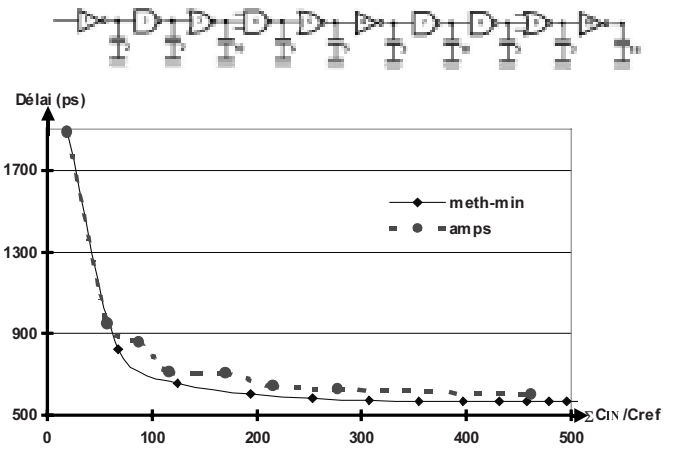

Fig. 3. Exploration of the design space with constant sensitivity constraint.

5 Validation

In order to validate these sizing and constraint distribution techniques we compare on different benchmarks the minimum delay value and the area obtained using the three investigated methods:

- equal distribution of delays (C), ($\Theta_i = \Theta_c/n$), where n is the number of stages,
- weighted distribution (7), (B),
- equal gate sensitivity (8), (A),

and using an industrial tool based on a Newton-Raphson based algorithm (D), [9] (Amps from Synopsys).

These benchmarks are constituted of array of gates with different loading conditions, their structure is shown in Table 1, where the output load of each gate C_P is given in unit of C_{REF} (C_{REF} =3.5fF).

The comparison of the minimum delay values obtained with each technique is given in Table 2 for different paths. The targeted process is the STm 0.25µm with τ = 7.05ps. As shown the lowest minimum value of delay is obtained with both the weighted and the equal sensitivity techniques.

This ascertains the method used to determine the lowest bound of delay on a logical path. Note that around the minimum value of delay the area penalty is very large, in that case this value must be considered more as an indicator for the feasibility of the constraint than as a target to be reached.

The next step is to compare for a given delay constraint the area of implementation obtained with the different distribution techniques. For that we impose on the different benchmarks a delay constraint defined between the bounds previously defined. Then we compare in Table 3 the area corresponding to the gate sizing allowing, with the different techniques, to match the constraint. We can observe that if for a weak constraint value the different techniques appear quite equivalent, for a

tighter constraint the equal sensitivity distribution technique (A) allows a match with a much smaller area than the others. Note that all the values given in Table 2 and 3 are obtained from Spice simulations (MM9 model) of the different benchmarks.

Table 1. Composition of the benchmarks used to validate the delay distribution methods.

Ver9									
Type	nd2	nd2	inv	nd2	nr2	nd3	nr2	nd2	nd3
Cp /Cref	4	6	8	6	6	4	6	4	5
Ver91									
Type	nd2	nd3	inv	nd2	nr3	nd3	nr2	nd2	nr3
Cp /Cref	4	6	8	6	6	4	6	4	9
Ver11											
Type	nr2	nd2	inv	nd2	nr3	inv	nr2	nd2	nd3	nd2	nr3
Cp /Cref	3	4	3	3	8	4	6	2	8	5	12
Ver15															
Type	nr3	nd2	nd3	inv	nr3	nd3	nr2	nd2	inv	nr2	inv	nd3	nr3	nd2	nr2
Cp /Cref	3	5	7	5	8	6	9	2	11	2	14	7	9	5	18
Ver151															
Type	nr2	nd2	inv	nd3	inv	nd2	nr3	nd2	nr3	inv	nr2	nd3	nr3	inv	nr3
Cp /Cref	3	7	2	5	8	2	3	7	11	2	8	7	7	3	18
Ver21															
Type	nr3	nd2	inv	nd2	nr2	nd3	nr2	nd2	nd3	nd2	nr3	nd2			
Cp /Cref	2	6	9	6	8	4	9	2	10	4	9	4			
Type	nr3	nd3	nr2	nd2	nr3	inv	nd2	nd3	inv	nd3	nr2				
Cp /Cref	9	2	9	8	7	7	4	3	11	7	15				
Ver31																
Type	nr2	nd3	nd2	inv	nr2	nd3	nr2	nd2	nd3	nd2	nr3	nd3	nr2	nd2	nr3	inv
Cp /Cref	2	6	9	6	8	4	9	2	10	4	9	2	9	8	7	7
Type	nd2	nd3	inv	nd3	nr2	inv	nr3	inv	nr2	nd2	nd3	nd3	nr3	nd2	nr2	
Cp /Cref	4	3	11	7	8	2	4	10	8	5	9	7	6	4	15	

The weighted distribution (B) still results in a quite equivalent area but the equal delay distribution (C) and Amps (D) may result for quite complex paths in an important increase of area. For some constraint values they may fail to find a solution (xxx).

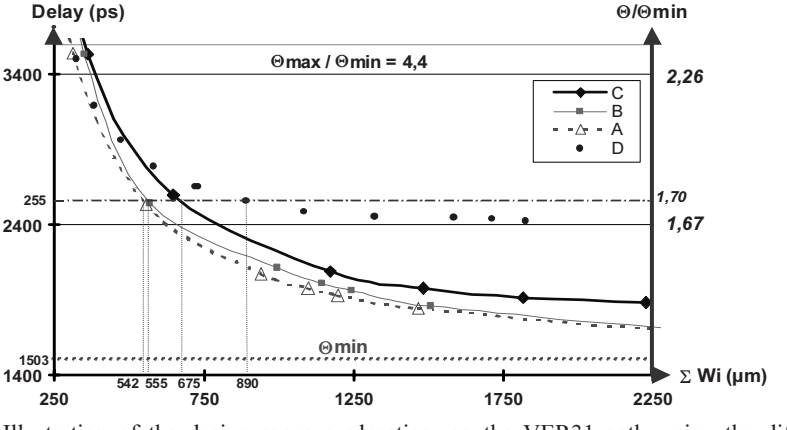

Fig. 4. Illustration of the design space exploration, on the VER31 path, using the different constraint distribution methods.

An illustration of these results is given in Fig.4 where we show for the path VER31 the complete exploration of the design space using the preceding constraint distribution methods. As shown for a delay constraint value smaller than $\Theta_{max}/2$ the gain in area (power) using the equal sensitivity or the weighted distribution method is quite significant.

Table 2. Comparison of the minimum delay values reached with the different sizing techniques: (A) is the equal sensitivity technique (8), (B) the weighted distribution, (C) the equal delay [4], (D) from Amps optimization tool.

Bench	**Gate nb**	**Max. delay value (ps)**	ΣW_i (μm)	Sizing Techn.	Min. delay value (ps)	ΣW_i (μm)
				A	544	668
Ver9	9	1399	38	B	544	668
				C	606	240
				D	602	403
				A	620	987
Ver91	9	1874	42	B	620	987
				C	676	391
				C	633	632
				A	698	1448
Ver11	11	2085	46	B	698	1448
				C	777	440
				D	937	348
				A	974	4077
Ver15	15	3588	64	B	974	4077
				C	1154	800
				D	1224	1047
				A	923	4337
Ver151	15	3479	3479	B	923	4337
				C	1023	1083
				D	960	3067
				A	1192	8419
Ver21	21	4583	94	B	1192	8419
				C	1484	1039
				D	1693	1152
				A	1503	21578
Ver31	31	6560	138	B	1503	21578
				C	1881	2226
				D	2426	1826

6 Conclusion

Based on a simple but realistic delay model for combinatorial gates, we have first determined an easy way to characterize the feasibility of a delay constraint to be imposed on a combinatorial path. We have defined reasonable maximum and real minimum delay bounds. Then we proposed two techniques to match a delay constraint on a path: the equal sensitivity and the weighted method that is a budgeting method. We have applied these methods on different benchmarks with various constraint conditions and compared the area of the resulting implementations with that obtained from an equal delay distribution and with an industrial tool. If for weak constraints the different methods are quite equivalent, for values near the minimum, the proposed methods always find a solution and result in an important area/power saving. Another point to be clarified further is to define at which distance of the minimum delay value it is reasonable to impose a constraint.

Table 3. Comparison of the area (given as the sum of the widths of the transistors) necessary to match the delay constraint in the different distribution techniques.

Bench	Sizing Techn.	$\Theta_{Max}/\Theta_{Min}$	Θ_C/Θ_{Min}	Area (µm)	Θ_C/Θ_{Min}	Area (µm)	Θ_C/Θ_{Min}	Area (µm)	Θ_C/Θ_{Min}	Area (µm)
	A			96		130		159		191
Ver9	B	2.6	1.4	100	1.23	132	1.15	159	1.11	193
	C			103		140		180		240
	D			104		155		245		338
	A			137		197		284		535
Ver91	B	3	1.4	147	1.2	202	1.1	292	1.02	560
	C			161		224		337		xxx
	D			144		202		289		632
	A			66		108		166		310
Ver11	B	3	2.15	80	1.6	134	1.34	195	1.1	330
	C			94		152		228		440
	D			70		127		348		xxx
	A			178		250		367		484
Ver15	B	3.7	1.7	202	1.47	270	1.27	384	1.18	538
	C			229		330		485		800
	D			195		321		789		xxx
	A			302		426		648		1333
Ver151	B	3.8	1.4	310	1.25	432	1.14	636	1.04	1407
	C			410		580		846		xxx
	D			324		486		786		3067
	A			196		406		522		553
Ver21	B	3.8	2.1	214	1.5	421	1.34	529	1.31	558
	C			230		479		651		715
	D			198		441		738		1152
	A			364		490		611		1275
Ver31	B	4.4	2.1	400	1.8	511	1.6	641	1.26	1361
	C			427		591		766		1970
	D			377		670		1826		xxx

References

[1] J. M. Shyu, A. Sangiovanni-Vincentelli, J. Fishburn, A. Dunlop, "Optimization-based transistor sizing" IEEE J. Solid State Circuits, vol.23, n°2, pp.400–409, 1988.

[2] J. Fishburn, A. Dunlop, "TILOS: a posynomial programming approach to transistor sizing" in Proc. Design Automation Conf. 1985,pp.326–328.
[3] C. Mead, M. Rem, "Minimum propagation delays in VLSI", ", IEEE J. Solid State Circuits, vol.SC17, n°4, pp.773–775, 1982.
[4] I. Sutherland, B. Sproull, D. Harris, "Logical Effort: Designing Fast CMOS Circuits", Morgan Kaufmann Publishers, INC., San Francisco, California, 1999.
[5] K. O. Jeppson, "Modeling the influence of the transistor gain ratio and the input-to-output coupling capacitance on the CMOS inverter delay", IEEE J. Solid State Circuits, vol.29, pp.646–654, 1994.
[6] S. Chakraborty, R. Murgai „Lay-out driven timing optimization by generalized DeMorgan transform“ IWLS 2001, pp.53–59.
[7] M. Ketkar, K. Kasamsetty, S. Sapatnekar "Convex delay models for transistor sizing" Proc. of the 2000 Design Automation Conf. pp.655–660.
[8] J. Grodstein, E. Lehman, H. Harhness, B. Grundmann, Y. Wanatabe „A delay model for logic synthesis of continuously-sized networks“, ICCAD 95, Nov 95.
[9] R. K. Brayton, R. Spence „Sensitivity and Optimization“ Elsevier 1980