

Explicit Logical Effort Formulation for Minimum Active Area under Delay Constraints

Caio G. P. Alegretti^{*}, Vinicius Dal Bem[#], Renato P. Ribas[#], André I. Reis[#]

[#]*PGMicro, UFRGS*
Porto Alegre, RS, Brazil

{cgpalegretti, vdbem, rpribas, andreis}@inf.ufrgs.br

^{*}*IFRS*
Canoas, RS, Brazil

Abstract— This paper presents a gate sizing method which formulates minimum active area solutions under delay constraints. It is based on the logical effort delay model. Such minimization of transistor widths has direct impact on the power consumption and circuit area reduction. The explicit formulation of the method takes into account the maximum input capacitance, the output load to be driven, and the imposed timing constraint. Electrical simulations have shown maximum errors of 4.1% in power, 5.62% in delay, and 13.5% in transistor sizes.

Keywords— active area minimization, gate sizing, logical effort, power minimization, design constraints.

I. INTRODUCTION

The problem of sizing a circuit is usually formulated as the task of choosing the sizes of logic gates to respect design delay constraints while minimizing other associated costs. Sizing methods have to rely in some form of delay model to estimate the delays. Logical effort [1] is a simple and practical delay model used by designers to have a first insight of possible delay optimizations that can be applied in a circuit. The popularity of the model comes from the fact that there is a very simple method to compute minimum delay in fan-out free paths by using logical effort. Notice that the name logical effort can be used to refer both to the delay model and to the associated sizing method, which can cause some confusion if not addressed properly.

A number of variations have been proposed both to the logical effort delay model and to the logical effort sizing method. Proposed improvements to the delay model are listed in [2]–[4]. Kabbani [2] modifies the delay model to take into account series-connected MOSFET structures, input transition time, inter-nodal charges, and DSM effects. Lasbouygues [3] proposes an extension to propagation delay representation, which considers I/O coupling capacitance and the input ramp effect. Wang [4] considers slope correction in the delay model and then uses the corrected delay model in a sizing tool. El-Masry [5] also proposed an enhanced model, which is used to study the effect of stacked transistors in complex gates to be used in library free approaches. Finally, the work of Kabbani [6] considers a specific timing design constraint, aiming at sizing a logic path for minimum sum of input capacitances under maximum delay. However, the work of Kabbani [6]

assumes that the path to be sized has an ideal number of stages, and so this minimum sum of capacitances is attained when all the gates bear the same effort [1]. Our method deals with non-ideal number of stages. In this case the premise of equal effort along the logic path is not true.

One open point in logical effort modeling is that the most straightforward delay computation with the method is always associated with minimum delay. However, the definition of the sizing problem is associated with a delay constraint that is normally larger than the minimum achievable delay. In this sense, it would be useful to have a logical effort formulation that respects delay constraints, instead of obtaining minimum achievable delay. In this paper we reformulate the logical effort model to explicitly compute minimum active area under a delay constraint. This way, we obtain a sizing method that, while still following logical effort model, is able to deal with delay constraints. In order to validate the model, we perform experiments to show (a) how well the proposed sizing method respects the delay constraint; and (b) how far from the absolute minimum active area and power consumption our sizing method is.

The rest of this paper is organized as follows. In Section II, the deduction of the new method is explained. In Section III, the proposed method is applied in the sizing of a three-stage subcircuit. Finally, last section is devoted to the conclusions.

II. DERIVING THE METHOD

In this Section, it is initially defined the type of cell sizing this paper refers to. Then, the deriving of the logical effort sizing method is reviewed, in a way to induce the development of the proposed method, which is done next.

A. Cell Sizing

The term cell sizing can be interpreted in two different ways. Cell sizing can be the sizing of a single cell to choose the relative sizes of the transistors in the transistor network used to synthesize the cell [7]. This is an important and necessary step for synthesizing a cell library, such as Nangate Open Cell Library [8]. Cell sizing can also be the choice of the cell sizes that are used for every instance in the final circuit [9]. In this paper, cell sizing has the second meaning. A seed size is obtained from [8] and then this seed size is scaled

by a scale factor affecting all the transistors in the same way.

B. The Logical Effort Sizing Method

The groundwork for the logical effort sizing method [1] is the homonymous gain-based delay model, which states that the absolute delay (d_{abs}) of a logic gate is given by:

$$d_{abs} = \tau(gh + p) \quad (1)$$

where τ is the delay of an inverter driving an identical inverter with no parasitics, g is the logical effort of the gate, h is the electrical effort, and p is the parasitic delay [1]. The relative delay d is given by the ratio between d_{abs} and τ .

Based on this model, the logical effort sizing method may be derived as follows. It is assumed that the total delay of a logic path is given by the sum of the delays of every logic gate in such path. Taking the derivative of this total delay with respect to the electrical effort h , one can see that minimum delay is obtained when the product gh is the same for each logic gate in the path.

C. The Proposed Method

The proposed method is also based on the logical effort delay model. However, it aims at achieving minimum active area for a specified delay D , rather than minimum delay. Therefore, the expression for the total active area of the subcircuit is derived with respect to the size of each logic gate. This size can be represented either as the input capacitance of the logic gate or as the scale factor [10] of such gate, since the scale factor is the ratio between the input capacitance of the logic gate and the input capacitance of the corresponding seed size in [8]. In this paper, the proposed method is deducted for a 3-stage fanout free subcircuit, with fixed topology, fixed extra parasitic capacitances, and fixed subcircuit input capacitance, as depicted in Fig. 1. Due to the lack of space, it is not explicitly shown that the method can also handle subcircuits with variable input capacitance and branching.

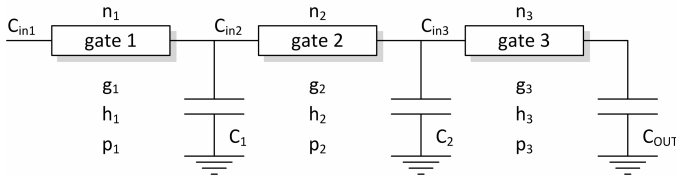


Fig. 1. Model of a 3-stage subcircuit

In the subcircuit shown, parameters g_i , h_i , and p_i ($i=1, 2, 3$) come from the logical effort delay model. n_i is the ratio between the total input capacitance of gate i and the capacitance of the input pin of gate i that belongs to the logic path under analysis. C_{in1} is the capacitance of the input pin of gate 1 that belongs to the logic path. C_1 and C_2 are fixed extra parasitic capacitances. C_{out} is the output capacitance, which might encompass another fixed extra parasitic capacitance in the output of gate 3.

The total relative delay (D) for the subcircuit depicted in Fig. 1 is specified by a delay constraint and is given by:

$$D = (g_1 h_1 + g_2 h_2 + g_3 h_3 + p_1 + p_2 + p_3) \quad (2)$$

Since $h_1 = (C_{in2} + C_1)/C_{in1}$, $h_2 = (C_{in3} + C_2)/C_{in2}$, $h_3 = C_{out}/C_{in3}$, $p_1 + p_2 + p_3 = P$, and $C_{in1} = C_{in1}^{fixed}$, equation (2) may be rewritten as

$$D = g_1 \frac{(C_{in2} + C_1)}{C_{in1}^{fixed}} + g_2 \frac{(C_{in3} + C_2)}{C_{in2}} + g_3 \frac{C_{out}}{C_{in3}} + P \quad (3)$$

Equation (3) shows the relationship between the variables C_{in2} and C_{in3} so that all design constraints for the subcircuit are fulfilled. Moreover, (3) may be rearranged as a univariate polynomial equation of the second degree on C_{in2} . Therefore, since all the other terms are constant parameters, C_{in2} may be expressed as a function of C_{in3} .

$$C_{in2} = \frac{-\beta \pm \sqrt{\beta^2 - 4g_1 C_{in3} \gamma}}{2g_1 C_{in3}} \quad (4)$$

where:

$$\beta = g_1 C_1 C_{in3} + C_{in1}^{fixed} g_3 C_{out} - C_{in1}^{fixed} (D - P) C_{in3} \quad (5)$$

$$\gamma = C_{in1}^{fixed} g_2 C_{in3} (C_{in3} + C_2) \quad (6)$$

According to [10], the active area of the subcircuit in Fig. 1 may be considered as

$$A(C_{in2}, C_{in3}) = n_1 C_{in1}^{fixed} + n_2 C_{in2} + n_3 C_{in3}. \quad (7)$$

I.e., the active area of a logic gate is monotonically related to its total input capacitance. Replacing the expression obtained in (4) for C_{in2} into (7), $A(C_{in2}, C_{in3})$ becomes a univariate equation on C_{in3} :

$$A(C_{in3}) = n_1 C_{in1}^{fixed} + n_2 \frac{(-\beta \pm \sqrt{\beta^2 - 4g_1 C_{in3} \gamma})}{2g_1 C_{in3}} + n_3 C_{in3} \quad (8)$$

Let:

$$\beta' = \frac{d\beta(C_{in3})}{dC_{in3}} = g_1 C_1 - C_{in1}^{fixed} (D - P) \quad (9)$$

$$\gamma' = \frac{d\gamma(C_{in3})}{dC_{in3}} = 2C_{in1}^{fixed} g_2 C_{in3} + C_{in1}^{fixed} g_2 C_2 \quad (10)$$

$$\nu = \beta^2 - 4g_1 C_{in3} \gamma \quad (11)$$

$$\nu = (g_1 C_1 C_{in3} + C_{in1}^{fixed} g_3 C_{out} - C_{in1}^{fixed} (D - P) C_{in3})^2 - 4g_1 g_2 C_{in1}^{fixed} C_{in3}^2 (C_{in3} + C_2) \quad (12)$$

$$\nu' = \frac{d\nu(C_{in3})}{dC_{in3}} = 2\beta\beta' - 4(g_1 \gamma + g_1 C_{in3} \gamma') \quad (13)$$

Taking the derivative of $A(C_{in3})$ in (8) and introducing the expressions in (9)-(13), we have:

$$\frac{dA(C_{in3})}{dC_{in3}} = \frac{n_2}{2g_1C_{in3}} \left(-\beta' \pm \frac{v'}{2v^{1/2}} \right) + \frac{n_2}{2g_1C_{in3}^2} \left(\beta \mp v^{1/2} \right) + n_3 \quad (14)$$

At this point, it is straightforward to obtain the minimum active area; it suffices to find the zeroes of (14):

$$n_2 \left[\left(-\beta' \pm \frac{v'}{2v^{1/2}} \right) g_1 C_{in3} + (\beta \mp v^{1/2}) g_1 \right] + 2n_3 (g_1 C_{in3})^2 = 0 \quad (15)$$

Equation (15) can be efficiently solved numerically, providing C_{in3} for minimum active area. Solving (4) with C_{in3} just obtained, we have the corresponding value of C_{in2} . Eventually, more than one pair of values is attained, but only one of them corresponds to minimum active area.

III. RESULTS

In this section we investigate the validity of the method compared with results obtained with electrical simulations. The circuit used for validation is shown in Fig. 2. The comparison was based on the NOCL Library [8], and the specific values of logical effort parameters g and p for the logic gates were obtained by simulation according to [1]. The formulation proposed herein was developed so that the input capacitance of the subcircuit under design may have either a fixed or a maximum value. Working with a variable – albeit limited – input capacitance gives an additional degree of freedom to the problem, making it easier to come to a global optimum. However, in this paper, for the sake of comparison with the logical effort sizing method, the input capacitance of the subcircuit was made equal to the minimum input capacitance of the NAND gate in [8].

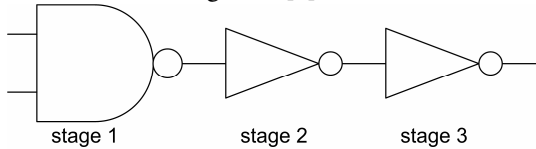


Fig. 2. Subcircuit to be sized.

The method was used to size this circuit for 20 different configurations of output load and delay constraints, which are shown in Table I. The column labeled $C\#$ presents the configuration identifier label, ranging from C1 to C20. The column labeled *Load* represents the output load used in the configuration, expressing how many times the load is larger than the minimum (X1) inverter in [8]. The third column (*Const.*) represents the *delay constraint* of the experiment configuration, given in picoseconds (ps). The column entitled *LE Ratio* explains how the delay constraint was obtained from the minimum achievable delay. First, we calculated the minimum possible delay for every load X_i , namely, LE_i , given by the logical effort sizing method [1]. Then, the minimum

achievable delay is augmented by a factor within the range 1/0.9 to 1/0.5, thus introducing a slack in the delay constraint. This slack is increased as long as no gate is sized to a scale factor smaller than 1, which would make any comparison meaningless.

For each of the case studies (C01 to C20), the circuit was sized with the proposed method, and the corresponding results are shown in the three columns in Table I under the *Proposed Method* title. The column labeled ΣW shows the sum of the transistor widths in the circuit obtained with the method. The column entitled *Delay (Pow)* shows the corresponding delay in picoseconds (power) delivered by HSPICE simulations for the circuit obtained with the method.

Table I presents two columns used as reference, listed under the title *Reference*. In order to generate the reference data, an exhaustive set of HSPICE simulations (level 6, using PTM 45 nm technology [11]) was performed for each output load (ranging from X4 to X100). The goal is to compare the results given by the sizing method with the minimum area obtained by exhaustive electrical simulations. The column labeled ΣW (*Pow*) shows the minimum possible ΣW (power) respecting the design constraints.

The comparison of the results from the method against HSPICE references is presented in three columns of Table I, under the title *Proposed Method (%)*. The column entitled $\Sigma W(\%)$ gives the percentage difference between the sum of widths obtained by the proposed method and the minimum reference obtained from HSPICE simulation datasets. The circuit can be oversized by 6.6% in the worst case. The column entitled $D(\%)$ gives the percentage difference between the delay for the circuit obtained by the proposed method and the delay constraint. Sometimes the delay is slightly larger than the delay constraint (by 3.5%), which is acceptable for a first fast computation. The column labeled $Pw(\%)$ gives the percentage difference between the power obtained by the proposed method and the minimum reference obtained from HSPICE simulation datasets. Notice that the delay difference generally has opposite signs with respect to both power and sum of widths differences, as expected.

The proposed method presents improvements over previous approaches. According to [6], the efforts [1] of the logic gates in the subcircuit should be the same for attaining minimum area. Nevertheless, HSPICE simulations show that, for the load X16 and delay constraint $LE_{16}/0.9$, the efforts for the 1st, 2nd and 3rd stages are given by 1.60, 2.54 and 4.85. The proposed method finds 1.39, 2.56 and 5.55, respectively. This result given by our method is much closer to the optimal result obtained by HSPICE simulations because, unlike [6], the proposed method takes into account the following facts: (a) the input capacitance of the subcircuit may have either fixed or variable – although limited – value; (b) the number of stages in the subcircuit may differ from the ideal number predicted by the logical effort sizing method [1]; (c) the cost function for the subcircuit area in (7) encompasses each logic

gate in its entirety, not just the capacitance of the input pin that belongs to the logic path under design.

IV. CONCLUSIONS

This paper presented a new method for sizing circuits based on an explicit logical effort formulation. The method is able to find the minimum active area of a subcircuit analytically, thus dismissing the use of iterative methods such as mathematical programming or algorithmic approaches. The minimum active area is achieved by solving a one-variable equation, which tends to be faster than iterative methods. Since power consumption is closely related to active area, this method is also capable of minimizing power.

The model accuracy has been validated with respect to HSPICE simulations, showing a 3.5% maximum delay error. This inaccuracy is inherent to the logical effort delay model. The usage of more accurate versions [2]-[5] of the logical effort delay model is under study. Such model version should consider the impact of the input signal slope on the delay of a logic gate. This way, the new sizing method would be able to cope with non-posynomial delay models. Such models cannot be solved by convex programming [12], and their solution by non-convex programming is not granted.

Another future work is related to the generalization of the proposed sizing method for subcircuits with an arbitrary number of stages. Currently, the method is derived for a finite set of logic path lengths. In addition, the method may be generalized in order to optimize power delay product. To the best knowledge of the authors this is the first approach for analytical sizing under delay constraints based on a logical effort formulation.

ACKNOWLEDGMENT

This work was partially supported by Brazilian funding agencies CAPES, CNPq, and FAPERGS, under grant 11/2053-9 (Pronem).

REFERENCES

- [1] I. Sutherland, B. Sproull, and D. Harris, "Logical Effort: Designing Fast CMOS Circuits." San Francisco: Morgan Kaufmann, 1999.
- [2] A. Kabbani, D. Al-Khalili, A. J. Al-Khalili, "Delay Analysis of CMOS Gates Using Modified Logical Effort Model," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, v. 24, n. 6, pp. 937-947, Jun. 2005.
- [3] B. Lasbouygues et al., "Logical Effort Model Extension to Propagation Delay Representation," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, v. 25, n. 9, pp. 1677-1684, Sept. 2006.
- [4] C. C. Wang, and D. Markovic, "Delay Estimation and Sizing of CMOS Logic Using Logical Effort with Slope Correction," *IEEE Trans. on Circuits and Systems*, vol. 56, n. 8, pp. 634-638, Aug. 2009.
- [5] H. El-Masry, and D. Al-Khalili, "Cell stack length using an enhanced logical effort model for a library-free paradigm," in *Proc. 18th IEEE Int. Conf. on Electronics, Circuits and Systems*, 2011, pp.703-706.
- [6] A. Kabbani, "Logical effort based dynamic power estimation and optimization of static CMOS circuits," *Integration, the VLSI journal*, vol. 43, pp. 279-288, 2010.
- [7] S. Roy, W. Chen, C. C. P. Chen, and Y. H. Hu, "Numerically Convex Forms and Their Application in Gate Sizing," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, n. 9, pp. 1637-1647, Sept. 2007.
- [8] Nangate 45nm Open Cell Library v1_3_v2010_12. Available: <http://www.nangate.com>
- [9] S. Hu, M. Ketkar, and J. Hu, "Gate Sizing for Cell-Library-Based Designs," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, n. 6, pp. 818-825, Jun. 2009.
- [10] S. P. Boyd, S. J. Kim, D. D. Patil, and M. A. Horowitz, "Digital Circuit Optimization via Geometric Programming," *Operations Research*, v.53, n.6, pp. 899-932, Nov.-Dec. 2005.
- [11] PTM 45 nm model. Available: <http://www.eas.asu.edu/~ptm>
- [12] H. Tennakoon, and C. Sechen, "Nonconvex Gate Delay Modeling and Delay Optimization," *IEEE Trans. On Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, n. 9, pp. 1583-1594, Sept. 2008.

TABLE I
RESULTS OF THE SIZING METHOD COMPARED WITH HSPICE REFERENCE FOR SEVERAL EXPERIMENT CONFIGURATIONS

Experiment Configuration				Proposed method			Reference		Proposed method (%)			Kabbani [7] (%)		
C#	Load	Const.	LE Ratio	ΣW	Delay	Pow.	ΣW	Pow.	$\Sigma W(\%)$	D(%)	Pw(%)	$\Sigma W(\%)$	D(%)	Pw(%)
C01	X4	42.7	(LE ₄ /0.9)	2.02	43.603	1.21	2.2	1.26	-8.0	+2.00	-4.0	+70,90	-8,38	+26,90
C02	X16	56.2	(LE ₁₆ /0.9)	4.01	58.210	3.24	4.6	3.38	-12.8	+3.45	-4.1	+77,59	-7,21	+23,40
C03	X16	63.2	(LE ₁₆ /0.8)	3.00	64.703	3.03	3.2	3.07	-6.3	+2.29	-1.3	+155,29	-17,48	+35,86
C04	X16	72.3	(LE ₁₆ /0.7)	2.49	72.769	2.92	2.6	2.94	-4.3	+0.70	-0.68	+214,20	-27,87	+41,87
C05	X32	65.7	(LE ₃₂ /0.9)	6.14	68.041	5.83	7.1	6.05	-13.5	+3.50	-3.6	+72,40	-6,76	+19,01
C06	X32	73.9	(LE ₃₂ /0.8)	4.49	75.669	5.47	4.8	5.54	-6.4	+2.38	-1.3	+155,01	-17,10	+29,96
C07	X32	84.4	(LE ₃₂ /0.7)	3.51	84.816	5.26	3.6	5.28	-2.4	+0.47	-0.38	+240,01	-27,42	+36,36
C08	X32	98.5	(LE ₃₂ /0.6)	2.90	96.865	5.14	2.9	5.14	+0.1	-1.67	0.0	+322,08	-37,81	+40,08
C09	X40	69.2	(LE ₄₀ /0.9)	7.05	71.711	7.08	8.1	7.33	-12.9	+3.50	-3.4	+72,48	-6,59	+17,98
C10	X40	77.8	(LE ₄₀ /0.8)	5.18	79.677	6.69	5.5	6.75	-5.9	+2.29	-0.89	+154,02	-16,92	+28,12
C11	X40	89.0	(LE ₄₀ /0.7)	3.99	89.349	6.43	4.1	6.45	-2.7	+0.42	-0.31	+240,75	-27,37	+34,08
C12	X64	77.6	(LE ₆₄ /0.9)	9.44	80.381	10.8	10.9	11.1	-13.4	+3.49	-2.7	+69,85	-6,26	+15,77
C13	X64	87.3	(LE ₆₄ /0.8)	6.81	90.203	10.2	7.4	10.3	-8.0	+3.25	-0.97	+150,19	-16,68	+24,76
C14	X64	99.7	(LE ₆₄ /0.7)	5.35	99.945	9.90	5.4	9.90	-0.9	+0.20	0.0	+242,85	-27,04	+29,80
C15	X64	116	(LE ₆₄ /0.6)	4.21	113.78	9.65	4.1	9.63	+2.7	-2.27	+0.21	+351,56	-37,29	+33,44
C16	X100	86.8	(LE ₁₀₀ /0.9)	12.5	90.002	16.2	14.4	16.7	-13.3	+3.51	-3.0	+68,56	-5,78	+13,29
C17	X100	97.7	(LE ₁₀₀ /0.8)	9.26	99.562	15.5	9.8	15.6	-5.5	+1.87	-0.64	+147,68	-16,29	+21,28
C18	X100	112	(LE ₁₀₀ /0.7)	7.14	111.47	15.0	7.1	15.1	+0.6	-0.16	-0.67	+241,87	-26,98	+25,30
C19	X100	130	(LE ₁₀₀ /0.6)	5.54	126.98	14.7	5.3	14.6	+4.4	-2.58	+0.68	+357,98	-37,09	+29,59
C20	X100	156	(LE ₁₀₀ /0.5)	4.37	148.00	14.5	4.1	14.4	+6.6	-5.62	+0.69	+492,03	-47,58	+31,39