Unified Logical Effort - A Method for Delay Evaluation and Minimization in Logic Paths with *RC* Interconnect

Arkadiy Morgenshtein, Eby G. Friedman, Ran Ginosar, and Avinoam Kolodny

Department of Electrical Engineering
Technion – Israel Institute of Technology
Haifa 32000, Israel

May 16, 2007

***Abstract* - A model for delay evaluation and minimization in paths composed of logic gates and *RC* wires is presented. The method, *Unified Logical Effort* (ULE), provides closed-form conditions for timing optimization while overcoming the breakdown of standard logical effort (LE) rules in the presence of interconnect. The ULE delay model unifies the problems of gate sizing and repeater insertion: In cases of negligible interconnect, the ULE method converges to standard LE optimization yielding tapered gate sizes. In the case of long wires, the solution converges towards uniform optimal sizing of the gates as in repeater insertion methodologies. The technique is applied to various logic path examples, in order to investigate the influence of wire length, gate type, and technology. Techniques for combining the ULE method with existing heuristics for buffering and repeater insertion are also proposed.**

I. INTRODUCTION

Timing modeling and optimization are two of the primary issues in high complexity circuit design. The method of logical effort (LE) was first proposed by Sutherland *et al.* [1] for fast evaluation and optimization of delay in logic paths (see Fig. 1a). The technique has since been adopted as a basis for many CAD tools, thanks to the simplicity of LE. The LE method benefits from an uncomplicated and intuitive delay model and closed-form optimization conditions. The optimization rule of logical effort, however, only addresses logic gates and does not consider on-chip wires. As VLSI circuits continue to scale, the contribution of wires to the delay increases and cannot be neglected. This characteristic occurs not only with respect to long wires connecting separate modules but also to the interconnect within logic modules where the delays introduced by the wires connecting closely coupled gates approach and can exceed the gate delays. The useful LE rule that the path delay is minimum when the effort of each stage is equal breaks down, because interconnect has fixed capacitances which do not correlate with the characteristics of the gates (see Fig. 1b). This behavior is described by the authors of the LE method as "one of the most dissatisfying limitations of logical effort" [2].

The objective of this work is to develop a simple method for minimizing delay in logic paths containing both gates and interconnect. Currently, timing optimization is typically treated separately in two cases: (a) logic gates without wires (using the standard LE method), (b) long wires without logic (using repeater

insertion). In this paper, the *Unified Logical Effort* (ULE) method is presented for delay evaluation and optimization in logic paths with general logic gates and *RC* wires. ULE treats a broad scope of design problems with a single analytic model, while combining both logic and interconnect delay optimization.

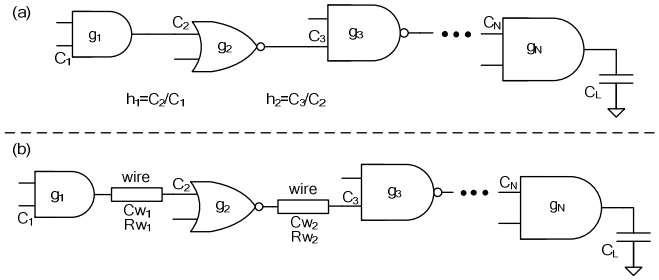


Fig. 1. Cascaded string of logic gates. (a) Logical effort optimization for gates without wires is based on equal stage efforts, $g_1h_1 = g_2h_2$ etc., (b) in the case of gates with wires, the rule of equal effort breaks down because of fixed wire parameters.

The paper is composed of the following sections. Related work is surveyed and discussed in Section III. The Unified Logical Effort model is developed in Section III. Timing optimization based on the ULE model referring to resistive and capacitive wires is presented in Section IV. A simple and fast method for determining the optimal gate size is also demonstrated. Examples of ULE optimization are presented in Section V. Convergence of the model to existing optimization techniques is shown for specific cases. Optimal gate sizing by ULE for long wires is analyzed in Section VI. The application of the analysis to repeater insertion is also presented. Additional optimization techniques are described in Section VII including delay minimization using repeater insertion and sizing by ULE, as well as sizing for power-delay product minimization. Simulation results of benchmark circuits are presented in Section VIII comparing the ULE optimization with the results of an industrial CAD tool optimizer. Finally, a summary of the paper as well as a discussion of future work are provided in Section IX.

II. RELATED WORK

Research has previously been carried out to increase the accuracy of the logical effort model by considering I/O coupling and ramp input effects [7], as well as internodal charge and deep submicrometer effects [8]. While increasing the accuracy of the LE method for logic gate delays, the research described in these papers do not address the issue of interconnects. Some publications propose an extension of logical effort to address the delay of the wires, using the term 'interconnect effort'. A logical effort optimization methodology is proposed in [9] for logic blocks driving interconnect with uniform and non-uniform repeaters.

Traditional timing optimization procedures have been developed assuming capacitive interconnect [10], [11], [12], focusing on optimally tapering those buffers driving the interconnect. In [13], [14], the wire capacitance between the gates is assumed to be correlated to the gate size, resulting in a fixed tapering factor similar to the logical effort model. In [12], local interconnect capacitances are considered to be independent of the gate size and the optimization is based on constant capacitance-to-current ratio tapering. In order to accurately consider resistive interconnect, post-routing design steps have been added, involving wire segmentation and repeater insertion [3],[4],[5],[6],[9]. These optimization techniques include equal

sizing and spacing of the repeaters [3], as well as tapering the repeater size and wire segments [9]. Most of these techniques for timing optimization in interconnect have been developed independent of the logical effort model, while focusing on inverters as buffers and repeaters driving long interconnects, rather than on general logic paths with wire segments.

The LE delay expression is applied to optimal wire segmentation by placing general logic gates rather than repeaters in [15] and [16]. The model presented in [15] uses a concept of resistive effort which is included within the logical effort expression. The model provides a specific optimization technique for logic gates spaced along the interconnect. An extension of logical effort is also used in [16] for developing an optimization methodology utilizing logic gates as repeaters. The delay expressions in these papers are only used as a basis for partitioning a long wire by logic gates, and have not been analyzed for the general case of a logic path with interconnects.

III. DELAY MODEL OF LOGIC GATES WITH WIRES

The logical effort model is modified here to include the interconnect delay. This change is achieved by extending the gate logical effort delay by the wire delay, establishing a *Unified Logical Effort* (ULE) model. In this section, the delay components of the ULE model are developed for use in the following section in order to determine the conditions for delay minimum.

A circuit composed of logic gates with wires is shown in Fig. 2. The interconnect is represented by a π-model. Following [17], the Elmore delay model [18] is used to describe the wire segment delay. Note that the analytic approach described in this paper can also be successfully applied to other interconnect models, both lumped and distributed. The total combined delay expression is

$$D_i = R_i \cdot \left(C_{p_i} + C_{w_i} + C_{i+1} \right) + R_{w_i} \cdot \left(0.5 \cdot C_{w_i} + C_{i+1} \right), \tag{1}$$

where R_i is the effective output resistance of the gate i, C_{p_i} is the parasitic output capacitance of gate i, C_{w_i} and R_{w_i} are, respectively, the wire capacitance and resistance of segment i, and C_{i+1} is the input capacitance of gate $i+1$.

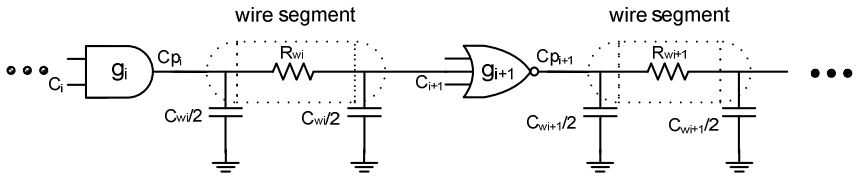


Fig. 2. Logic gates with interconnect load.

This expression is rewritten by introducing the delay of a minimum size inverter as a technology constant $\tau = R_0 \cdot C_0$, where R_0 and C_0 are the output resistance and input capacitance of a minimum sized inverter, respectively,

$$D_i = \tau \cdot \left[\frac{R_i}{R_0} \cdot \frac{\left(C_{w_i} + C_{i+1} + C_{p_i} \right)}{C_0} + \frac{R_{w_i}}{R_0 \cdot C_0} \cdot \left(0.5 \cdot C_{w_i} + C_{i+1} \right) \right]. \tag{2}$$

The components in the delay expression are transformed into the form used in the logical effort (LE) model [1]. The delay of a logic gate is expressed in LE as a value d_i, independent of the particular technology and normalized with respect to a minimum inverter delay τ,

$$D_i = \tau \cdot d_i = \tau \cdot \left(g_i \cdot h_i + p_i \right), \tag{3}$$

where $g_i = \left(R_i \cdot C_i \right) / \left(R_0 \cdot C_0 \right)$ is the logical effort related to the gate topology, $h_i = C_{i+1} / C_i$ is the electrical effort describing the driving capability, and $p_i = \left(R_i \cdot C_{p_i} \right) / \left(R_0 \cdot C_0 \right)$ is the delay factor of the parasitic impedance.

The expression of the total stage delay, using LE terms, is

$$d_i = g_i \cdot \left(h_i + \frac{C_{w_i}}{C_i} \right) + \frac{R_{w_i} \cdot \left(0.5 \cdot C_{w_i} + C_{i+1} \right)}{\tau} + p_i \,. \tag{4}$$

The *capacitive interconnect effort* h_w and the *resistive interconnect effort* p_w are

$$h_{w_i} = \frac{C_{w_i}}{C_i} \quad . \tag{5}$$

$$p_{w_i} = \frac{R_{w_i} \cdot \left(0.5 \cdot C_{w_i} + C_{i+1} \right)}{\tau} \,. \tag{6}$$

As shown in (5), h_w expresses the influence of the interconnect capacitance on the electrical effort of the gate. The component p_w in (6) is the delay of the loaded wire in terms of the gate delay (τ). The component $\left(R_w \cdot 0.5 \cdot C_w \right) / \tau$ is technology specific.

The final expression of the ULE delay for a single stage is

$$d = g \cdot \left(h + h_w \right) + \left(p + p_w \right). \tag{7}$$

The ULE delay expression for an *N* stage logic path with *RC* wires is

$$d = \sum_{i=1}^{N} g_i \cdot \left(h_i + h_{w_i} \right) + \left(p_i + p_{w_i} \right). \tag{8}$$

Note that in the case of short wires, the resistance R_w of the wire may be neglected, eliminating p_w and only leaving the capacitive interconnect effort h_w in the expression. The extended delay expression of (4) reduces to the LE delay equation when no wires exist along the logic path.

IV. DELAY MINIMIZATION USING UNIFIED LOGICAL EFFORT

In this section, the ULE method is developed for minimizing the delay of a general logic path with interconnect. The optimum conditions in terms of the electrical effort are presented for resistive and capacitive wires. Close-form expressions are determined for optimal gate sizing followed by a method for fast and simple calculation of optimal gate sizing along a logic path with wires.

As the first step in optimizing the ULE expression, a two-stage portion of a logic path with wires (as shown in Fig. 2) is considered. In this case, the ULE expression of the total delay is

$$d = g_i \cdot \left(h_i + h_{w_i}\right) + \left(p_i + p_{w_i}\right) + g_{i+1} \cdot \left(h_{i+1} + h_{w_{i+1}}\right) + \left(p_{i+1} + p_{w_{i+1}}\right), \tag{9}$$

where the electrical effort of each stage is $h_i = C_{i+1}/C_i$ and $h_{i+1} = C_{i+2}/C_{i+1}$. A plot of the delay as a function of the gate capacitance is illustrated in Fig. 3. The input capacitance is related to the driving ability of the gate. The plot illustrates an extreme point, where the optimal gate capacitance results in the minimum delay.

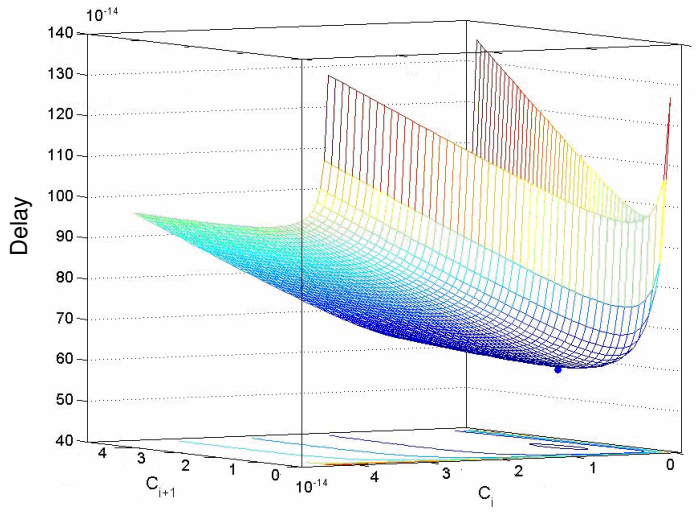


Fig. 3. Delay profile as a function of gate capacitance for 100 µm long wires. The minimum delay can be used to determine the optimal gate sizes C_i and C_{i+1}.

Substituting $C_{i+1} = h_i \cdot C_i$ into (9) in the presence of resistive interconnect, the delay can be expressed in terms of h_i as

$$d = g_i \cdot \left(h_i + \frac{C_{w_i}}{C_i}\right) + p_i + \frac{R_{w_i} \cdot \left(0.5 \cdot C_{w_i} + h_i \cdot C_i\right)}{R_0 \cdot C_0} + g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{h_i \cdot C_i}\right) + p_{i+1} + p_{w_{i+1}}. \tag{10}$$

The optimal condition is determined by equating the derivative of the delay to zero,

$$\frac{\partial d}{\partial h_i} = g_i + \frac{R_{w_i} \cdot C_i}{R_0 \cdot C_0} - g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{h_i^2 \cdot C_i}\right) = 0, \tag{11}$$

$$h_i = \sqrt{\frac{g_{i+1}}{g_i + \dfrac{R_{w_i} \cdot C_i}{R_0 \cdot C_0}} \cdot \left(\frac{C_{i+2}}{C_i} + \frac{C_{w_{i+1}}}{C_i}\right)}. \tag{12}$$

Substituting $C_{i+2}/C_i = h_i \cdot h_{i+1}$ and $C_{w_{i+1}}/C_i = h_i \cdot h_{w_{i+1}}$,

$$h_i = \sqrt{\frac{g_{i+1}}{g_i + \dfrac{R_{w_i} \cdot C_i}{R_0 \cdot C_0}} \cdot \left(h_i \cdot h_{i+1} + h_i \cdot h_{w_{i+1}} \right)} \,. \tag{13}$$

The general condition of the electrical effort for minimal delay of logic stage i with *RC* interconnect is

$$\left(g_i + \frac{R_{w_i} \cdot C_i}{R_0 \cdot C_0} \right) \cdot h_i = g_{i+1} \cdot \left(h_{i+1} + h_{w_{i+1}} \right). \tag{14}$$

In order to supply an intuitive interpretation of the expression, it can be rewritten as follows:

$$\left(g_i + \frac{R_{w_i} \cdot C_i}{R_0 \cdot C_0} \right) \cdot \frac{C_{i+1}}{C_i} = g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{C_{i+1}} \right). \tag{15}$$

When both sides of the equation are multiplied by $R_0 \cdot C_0$ it transforms to:

$$R_0 \cdot C_0 \cdot g_i \cdot \frac{C_{i+1}}{C_i} + R_{w_i} \cdot C_{i+1} = R_0 \cdot C_0 \cdot g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{C_{i+1}} \right). \tag{16}$$

The expression can be further simplified by using the relationships $C_i = C_0 \cdot g_i \cdot x_i$ and $C_{i+1} = C_0 \cdot g_{i+1} \cdot x_{i+1}$ (where x_i and x_{i+1} are the sizing factors of gate i and $i+1$, respectively):

$$\begin{aligned} R_0 \cdot C_0 \cdot g_i \cdot \frac{C_{i+1}}{C_0 \cdot g_i \cdot x_i} + R_{w_i} \cdot C_{i+1} &= R_0 \cdot C_0 \cdot g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{C_0 \cdot g_{i+1} \cdot x_{i+1}} \right) \\ \frac{R_0}{x_i} \cdot C_{i+1} + R_{w_i} \cdot C_{i+1} &= \frac{R_0}{x_{i+1}} \cdot \left(C_{i+2} + C_{w_{i+1}} \right) \end{aligned} \,. \tag{17}$$

By applying the relationships $R_i = \dfrac{R_0}{x_i}$ and $R_{i+1} = \dfrac{R_0}{x_{i+1}}$ we obtain the final optimum condition:

$$\left(R_i + R_{w_i} \right) \cdot C_{i+1} = R_{i+1} \cdot \left(C_{i+2} + C_{w_{i+1}} \right). \tag{18}$$

The meaning of (18) *is that the optimum sizing of gate* $i+1$ *is achieved when the delay component* $\left(R_i + R_{w_i} \right) \cdot C_{i+1}$ *due to the gate capacitance is equal to the delay component* $R_{i+1} \cdot \left(C_{i+2} + C_{w_{i+1}} \right)$ *due to the effective resistance of the gate.*

The scheme with the related delay components is shown in Fig. 8. Note that the delay components $R_i \cdot C_{w_i}$, $0.5 \cdot R_{w_i} \cdot C_{w_i}$ and $R_{w_{i+1}} \cdot \left(0.5 \cdot C_{w_{i+1}} + C_{i+2} \right)$ are fixed and do not influence the sizing optimum.

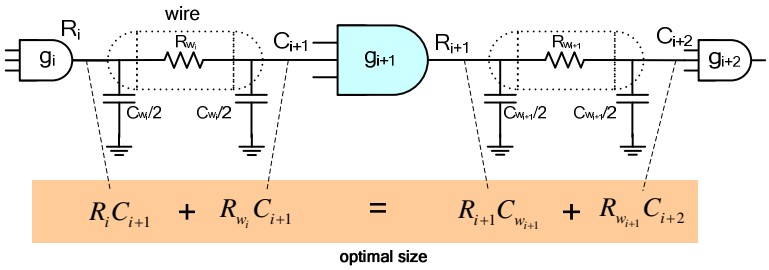


Fig. 4. Delay components in characterization of ULE optimum for long wires

Note that the expression (18) can be used for explanation of logical effort optimization without wires, by applying $R_{w_i} = 0,\ C_{w_{i+1}} = 0$. The resulting optimum condition is:

$$R_i \cdot C_{i+1} = R_{i+1} \cdot C_{i+2}\,. \tag{19}$$

The expression (19) corresponds with the optimum $g_i \cdot h_i = g_{i+1} \cdot h_{i+1}$ as was defined in the LE method

The optimum condition represented by (14) addresses resistive interconnect, which usually refers to long and intermediate length wires. In the case of short local wires, the interconnect effort may be simply analyzed by only considering the capacitive component of the interconnect impedance. The same analysis can be applied to the short branches of the logic gates. The condition of the electrical effort for minimum delay (14) reduces to the case of capacitive interconnect, as described by the following expression,

$$g_i \cdot h_i = g_{i+1} \cdot \left(h_{i+1} + h_{w_{i+1}}\right). \tag{20}$$

For a logic path without wires ($h_{w_i} = 0$, $R_{w_i} = 0$), the optimum conditions of ULE, (14) and (20), converge to the optimum of LE [1]: $g_i \cdot h_i = g_{i+1} \cdot h_{i+1}$. Note that the primary difference between the LE and ULE optima for capacitive wires is the presence of the capacitive interconnect effort $h_{w_{i+1}}$. The right part of the condition expresses the effort applied by the load on gate $i+1$. The left part represents the relative driving ability. Thus, the factor $h_{w_{i+1}}$ includes the contribution of the wire to the load of gate $i+1$.

The driving ability of a gate is related to the size of the gate and can be represented by a ratio of input capacitances [1]. The optimum condition in (14) can be rewritten in order to develop the expression for the optimal input capacitance of each gate using the ULE model,

$$\left(g_{i-1} + \frac{R_{w_{i-1}} \cdot C_{i-1}}{R_0 \cdot C_0}\right) \cdot \frac{C_i}{C_{i-1}} = g_i \cdot \left(\frac{C_{i+1}}{C_i} + \frac{C_{w_i}}{C_i}\right), \tag{21}$$

$$C_{i_{opt}} = \sqrt{\frac{g_i}{g_{i-1} + \dfrac{R_{w_{i-1}} \cdot C_{i-1}}{R_0 \cdot C_0}} \cdot C_{i-1} \cdot \left(C_{i+1} + C_{w_i}\right)} = \underbrace{\sqrt{C_{i-1} \cdot C_{i+1}}}_{\text{LE}} \cdot \underbrace{\sqrt{\left(1 + \frac{C_{w_i}}{C_{i+1}}\right)}}_{\text{wire capacitance}} \cdot \underbrace{\sqrt{\frac{g_i}{g_{i-1} + \dfrac{R_{w_{i-1}} \cdot C_{i-1}}{R_0 \cdot C_0}}}}_{\text{logical efforts and wire resistance}}. \tag{22}$$

Note that the first part of the resulting expression is similar to the condition described by the LE model for a path of identical gates. The second component expresses the influence of the interconnect capacitance.

The last component is related to the resistance of the wire and the difference between the individual logical efforts (types of logic gates) along the path.

In the case of a capacitive wire or branch, the expression for the optimal gate capacitance reduces to

$$C_{i_{opt}} = \sqrt{C_{i-1} \cdot C_{i+1}} \cdot \sqrt{\left(1 + \frac{C_{w_i}}{C_{i+1}}\right)} \cdot \sqrt{\frac{g_i}{g_{i-1}}} \ . \tag{23}$$

The expression for the optimal capacitance in (22) and (23) illustrates the quadratic relation between the size of the neighboring gates. The optimal gate size based on ULE can be determined by solving a set of N polynomial expressions for N gates along the path.

In order to reduce the complexity of the solution, a relaxation method can be used. The technique is based on an iterative calculation along the path while applying the optimum conditions:

a. (Initialization) Set the gate capacitances along the path to arbitrary values (only the first and last values are given).

b. (Iteration) Replace each capacitance by the value determined from applying the optimum expression (22) *or* (23) *on two neighboring logic gates.*

c. (Stop check) If any of the new values differ by more than a given precision from the previous value, reiterate step b.

The application of the algorithm generally produces the optimal size, converging to 5% accuracy after three iterations. The gates in the last few stages of the path are the first to converge, since the accuracy increases while propagating along the path from the leaf to the root of the path. Consequently, fewer calculations are performed in each successive iteration.

V. EXAMPLE LOGIC PATHS

The ULE technique is applied here to several example logic paths to demonstrate the properties of optimal gate sizing. The demonstration has been performed based on parameters from [20] for a 65 nm CMOS technology. The examples include the special case where all of the gates have the same logic function and all of the wires have the same length L to demonstrate the relationship of ULE to existing optimization techniques. These examples are followed by more general examples of logic paths with different logic gates and wire segments.

The first example logic path is shown in Fig. 5 and consists of nine identical stages. The example path illustrates a typical case used in LE, where there is a high total electrical effort. The input capacitance of the first and last gates are $10 \cdot C_0$ and $100 \cdot C_0$, respectively. The optimal size of the logic gates along the path is shown in Fig. 5 for several values of wire length L. The ULE optimization process provides an optimal solution over any range of wire length, where the special cases of long and short interconnects converge to existing techniques. All of the solutions range between two limits (the bold lines in the plot): (a) for zero wire lengths, the solution converges to LE optimization [1], and (b) for long wires, the gate size in the middle stages of the path converges to an equal value, $x_{opt} = 49.6$ (the dashed line), similar to repeater insertion methods [3],[16]. The phenomenon of equal optimal sizing x_{opt} for long wires is explained in the following section. While the LE solution (without wires) exhibits uniform gate size tapering, the long wire

solution exhibits different tapering at the first and last stages, depending upon the input and load capacitances.

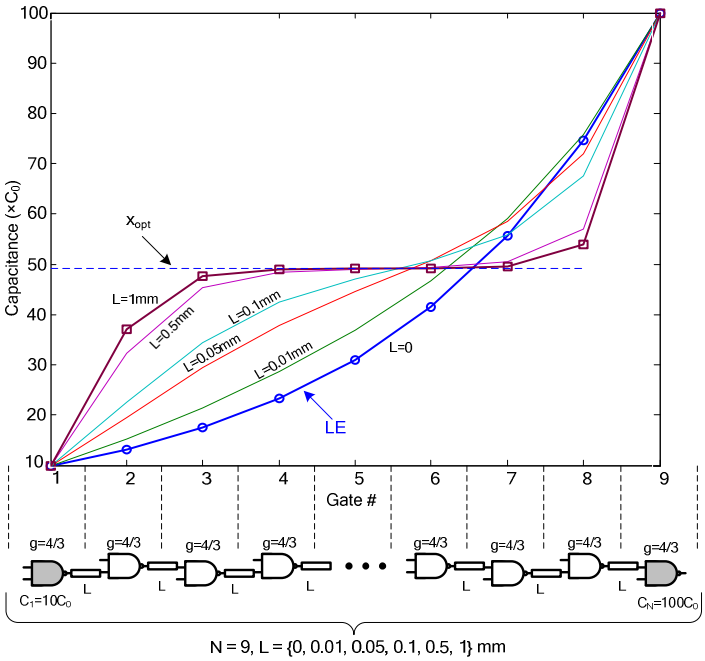


Fig. 5. Optimal ULE sizing (normalized with respect to C_0) in a NAND chain of nine gates with equal wire segments for a variety of lengths. For zero wire length, the solution converges to LE optimization. For long wires, the solution converges to equal sizing x_{opt}. The Parameters of 65 nm process include R_0 = 8800 Ω, C_0 = 0.74 fF, intermediate wires - r_w = 1.0 Ω/μm, c_w = 0.15 fF/μm, global wires - r_w = 0.04 Ω/μm, and c_w = 0.23 fF/μm.

An additional example is shown in Fig. 6 for the case of total electrical effort $H = 1$. The contents of the logic chain is similar to the previous case, but the input and output gate capacitances are equal to $10 \cdot C_0$. This case is not typical for regular LE optimization, since no gate scaling should be performed here by LE in the absence of wires. Note that the ULE optimization process provides an optimal solution for a variety of wire lengths, while meeting LE optimization (no scaling) in the case of zero wire length and converging to equal sizing $\left(x_{opt} = 49.6\right)$ for long wires.

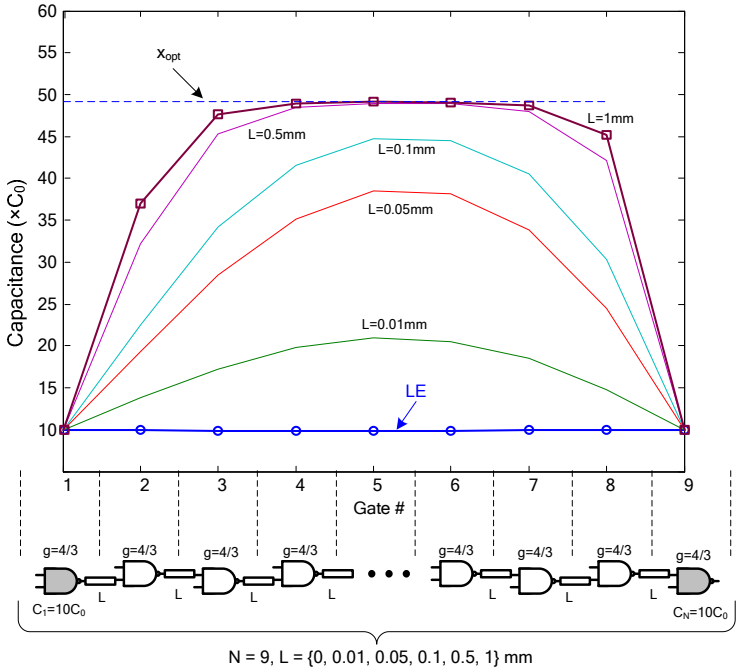


Fig. 6. Optimal ULE sizing (normalized to C_0) for circuit E2 with a varying number of gates along the path. Convergence to optimal equal sizing depends upon the number of gates along the path. The equal sizing x_{opt} is achieved asymptotically.

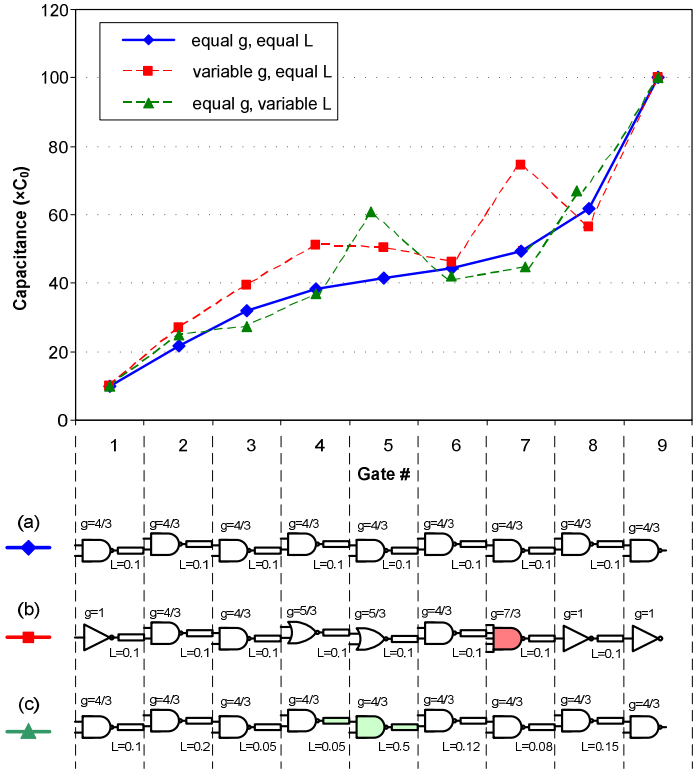


Fig. 7. Optimal ULE sizing (normalized with respect to C_0) (a) similar gates and equal wire lengths, (b) various gates and equal wire lengths, and (c) similar gates and various wire lengths. Those gates with higher logical effort have a higher scaling factor. Circuits with various gates or wires do not converge to a single x_{opt}. The most irregular stages are highlighted.

In Fig. 7, the optimal size is shown for three circuits: (a) all of the gates along the path are of similar type (NAND) with equal logical effort $\left(g = 4/3\right)$ and equal wire length, (b) the path contains different gates but equal wire length between each logic stage, and (c) the path contains similar gates but different wire length between each logic stage (the total wire length is equal in all cases). The optimal size changes as a function of g_i and L_i according to the optimization condition described by (22). As a result of the difference in driving capability, those gates with higher logical effort have a relatively larger scaling factor. The difference in the wire length between the stages has a similar effect on the optimal gate size - a larger size is required for all of the gates to drive longer interconnect segments. Note that due to the difference in gate type and wire length, the solution does not converge to a single equal size for the logic gates within the three circuits.

VI. ULE GATE SIZING FOR LONG WIRES

As was shown in the previous section, in cases of long wire segments, the optimal gate scaling process converges to the optimal scale factor x_{opt} . This scale factor is independent of the wire lengths in case of equal interconnect segments. In this section, the delay model of the logic gate with long wires is investigated in terms of the optimal size.

When long wires are assumed, the components C_{w_i} and $R_{w_{i-1}}$ of (22) become dominant. A schematic model of this case is shown in Fig. 8.

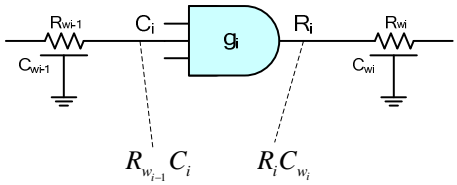


Fig. 8. Delay components of optimum ULE for long wires

Equation (22) can be rewritten for long wires as:

$$C_{i_{opt}} = \sqrt{C_{i-1} \cdot C_{i+1}} \cdot \sqrt{\frac{C_{w_i}}{C_{i+1}}} \cdot \sqrt{\frac{g_i}{\frac{R_{w_{i-1}} \cdot C_{i-1}}{R_0 \cdot C_0}}} = \sqrt{\frac{C_{w_i}}{R_{w_{i-1}}} \cdot g_i \cdot R_0 \cdot C_0} \ . \tag{24}$$

The optimum for scale factor of a general gate is

$$x_{opt_i} = \frac{C_{i_{opt}}}{C_{i_{\min}}} = \frac{C_{i_{opt}}}{C_0 \cdot g_i} = \sqrt{\frac{R_0 \cdot C_{w_i}}{R_{w_{i-1}} \cdot C_0 \cdot g_i}} = \underbrace{\sqrt{\frac{c_w \cdot R_0}{r_w \cdot C_0 \cdot g_i}}}_{\text{constant}} \cdot \sqrt{\frac{L_i}{L_{i-1}}} \ . \tag{25}$$

Note that the optimal scale factor of the gate in the case of long wires only depends upon the ratio of the adjacent wire lengths.

An important characteristic of x_{opt_i} can be derived from (25), which is transformed to

$$\frac{R_0}{x_{opt_i}} = \frac{x_{opt_i}}{\left(\frac{C_{w_i}}{R_{w_{i-1}} \cdot C_0 \cdot g_i}\right)} = x_{opt_i} \cdot C_0 \cdot g_i \cdot \frac{R_{w_{i-1}}}{C_{w_i}} . \tag{26}$$

Equation (26) can be rewritten using the expressions $R_i = R_0 / x_{opt_i}$ and $C_i = x_{opt_i} \cdot C_0 \cdot g_i$

$$R_i = C_i \cdot \frac{R_{w_{i-1}}}{C_{w_i}} , \tag{27}$$

and

$$R_i \cdot C_{w_i} = C_i \cdot R_{w_{i-1}} . \tag{28}$$

The meaning of (28) is that the optimum sizing condition is achieved when the delay component $R_{w_{i-1}} \cdot C_i$ *due to the gate capacitance is equal to the delay component* $R_i \cdot C_{w_i}$ *due to the effective resistance of the gate.*

An additional characteristic of x_{opt_i} can be developed from (24),

$$R_{w_{i-1}} \cdot C_i = \sqrt{C_0 \cdot R_0 \cdot g_i \cdot R_{w_{i-1}} \cdot C_{w_i}} = \sqrt{(\tau \cdot g_i) \cdot (r_w c_w)} \cdot \sqrt{L_i \cdot L_{i-1}} . \tag{29}$$

Both conditions (28) and (29) are combined to form a general condition,

$$R_i \cdot C_{w_i} = C_i \cdot R_{w_{i-1}} = \sqrt{(\tau \cdot g_i) \cdot (r_w c_w)} \cdot \sqrt{L_i \cdot L_{i-1}} . \tag{30}$$

The meaning of (30) *is that the optimum sizing condition is achieved when the delay component* $R_{w_{i-1}} \cdot C_i$ *due to the gate capacitance is equal to the delay component* $R_i \cdot C_{w_i}$ *due to the effective resistance of the gate, and is equal to the product of technology constant with the geometric mean of wire lengths.*

The dependence of the delay on the sizing factor is exemplified in Fig. 9. Observe that choosing sizing factors different from x_{opt} will increase the delay. The total delay D_{tot} is comprised of four components: the constant delays $0.5 \cdot R_{w_{i-1}} C_{w_{i-1}}$ and $0.5 \cdot R_{w_i} C_{w_i}$, and the variable delays $D_r = R_i \cdot C_{w_i}$ and $D_c = C_i \cdot R_{w_{i-1}}$ that are dependant on the sizing factor x. The optimum conditions (30) can be observed by inspecting the curves D_r and D_c, as well as $D_{av} = \sqrt{(\tau \cdot g_i) \cdot (r_w c_w)} \cdot \sqrt{L_i \cdot L_{i-1}}$. The optimal value of the sizing factor x_{opt} is obtained at the intersection point of the three curves D_r, D_c and D_{av}, as described in (30).

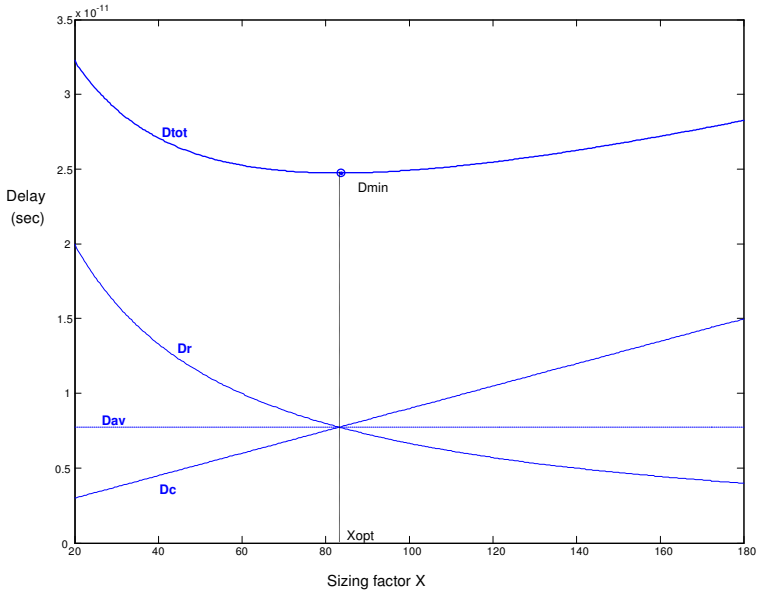


Fig. 9. Dependence of delay on the sizing factor for long wires (for a NAND gate with L_i=100μm and L_{i-1}=1mm).

In the special case of equal wire segments, the capacitance and resistance of all the segments are equal to C_w and R_w, respectively. In this case, the scaling factor x_{opt} is independent of the wire length since the component C_w/R_w is independent of wire length. The optimum condition can be rewritten as a function of capacitance and resistance per length unit c_w and r_w,

$$x_{opt_i} = \sqrt{\frac{R_0 \cdot c_w}{r_w \cdot C_0 \cdot g_i}} \,. \tag{31}$$

For the special case of repeater insertion (inverters with an electrical effort $g = 1$), the condition in (31) reduces to

$$x_{opt} = \sqrt{\frac{R_0 \cdot c_w}{r_w \cdot C_0}} \,. \tag{32}$$

This optimal sizing factor is the same as the optimal repeater scaling described by Bakoglu in [3].

In addition, similar to (28), the optimal sizing condition for a repeater is

$$R_{rep} \cdot C_w = C_{rep} \cdot R_w \,. \tag{33}$$

The optimal sizing of repeater is achieved when the delay component $R_w \cdot C_{rep}$ due to the repeater capacitance is equal to the delay component $R_{rep} \cdot C_w$ due to the effective resistance of the repeater.

The application of ULE to repeater insertion provides solution to some specific design problems. Two examples are presented here:

- *Layout constrain:* given a wire of total length L comprised of two segments of lengths L_1 and L_2, the optimal size of the repeater located between the segments is

$$x_{rep_{opt}} = \sqrt{\frac{c_w \cdot R_0}{r_w \cdot C_0 \cdot g_i}} \cdot \sqrt{\frac{L_2}{L_1}} \,. \tag{34}$$

- *Cell size constrain:* given a repeater of size x_{rep} dividing a wire of total length L into two segments, the optimal segment lengths $L_{1_{opt}}$ and $L_{2_{opt}} = L - L_{1_{opt}}$, are

$$\frac{L_{2_{opt}}}{L_{1_{opt}}} = x_{rep}^2 \Big/ \left(\frac{c_w \cdot R_0}{r_w \cdot C_0 \cdot g_i} \right). \tag{35}$$

VII. ADDITIONAL OPTIMIZATION TECHNIQUES

The ULE optimization technique presented in the previous sections targeted delay minimization. Several additional design objectives are considered in this section based on the ULE model. The proposed methodologies extend the use of ULE to delay minimization by buffers and repeaters, as well as power optimization.

ULE Sizing of Repeaters for Delay Minimization

The proposed ULE technique decreases the delay of a logic path composed of a number of sized stages connected by wires. A further decrease in delay can be achieved by changing the number of stages by inserting additional repeaters. This approach is similar to adding inverters to a logic path in LE to obtain an optimal number of stages, as in repeater insertion [3],[16]. In many existing repeater insertion methodologies, the inverters can be placed anywhere along the wire; however, practically, the wire segments are fixed along the path after placement and routing (see Fig. 10a).

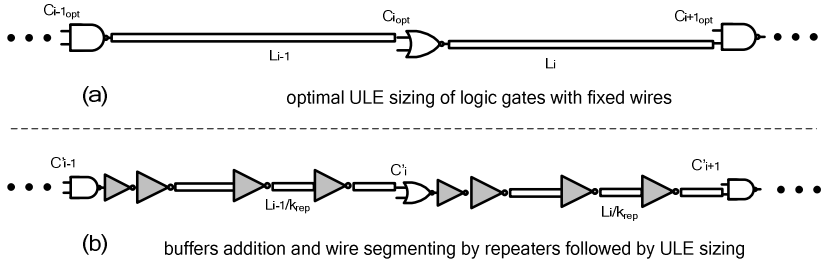


Fig. 10. Optimal sizing of number of stages: (a) ULE sizing is performed on gates with fixed wire length, (b) repeaters can be inserted along wires, while cascaded buffers are added and sized to maintain small logic gates.

The first optimization heuristic incorporated into ULE is based on repeater insertion within long wire segments along a path (see Fig. 10b). In this case, the wires are divided by repeaters into shorter segments prior to ULE sizing. For example, the number of repeaters can be determined from [3]. After repeater insertion, ULE optimization is performed to determine the optimal size of the individual logic gates and repeaters.

Another optimization step can be performed in addition to repeater insertion. The optimal size of those logic gates that drive long interconnect can be large, expending excessive area and power. This effect is most significant in multiple input logic gates where all of the transistors are scaled to obtain the optimum size if one of the inputs is along the critical path. Buffers are inserted to drive the long wire segments, as

shown in Fig. 10b, to solve this problem. The optimal number of inverters in each cascaded buffer is determined from the LE model [1] since the wire impedance between the cascaded inverters is negligible. Sizing the entire path is determined from ULE.

Note that the resulting size of the repeaters after ULE optimization can be different than the equal size as described in [3]. As exemplified in previous sections, the ULE solution converges to the equal sizing factor in the case of long wires, but produces different results when shorter wire segments are considered (such as in repeaters insertion) and when sizing is influenced by neighboring gates. Thus, after repeater insertion, the ULE optimization process produces optimal sizing, resulting in a shorter delay as compared to equal sizing in [3].

ULE Gate Sizing for Power-Delay Product Minimization

Sizing the gates for minimum delay can result in a large size with significant power dissipation. However, a power-efficient optimization is often desirable. In these cases, a power-delay product as the minimization goal will result in a smaller gate size while trading of delay and power.

The delay of a two stage chain (see Fig. 2) is described in (10) and is a function of h_i. The dynamic power is represented by the capacitance of the gate $i+1$ and the wire capacitance:

$$P \propto \left(C_{i+1} + C_{w_{i+1}}\right) = C_i \cdot h_i + C_{w_{i+1}} . \tag{36}$$

The optimal condition is determined by setting the derivative of the power-delay product to zero, resulting in the following expression (see Appendix A for the complete derivation) for the optimal input capacitance C_i,

$$\begin{aligned}
&C_i^3 \cdot a_1 + C_i^2 \cdot a_2 + C_i \cdot a_3 + a_4 = 0 \\
&a_1 = 2\left(g_{i-1} + \frac{R_{w_{i-1}} \cdot C_{i-1}}{\tau}\right) \\
&a_2 = \left(g_{i-1} \cdot \left(C_{w_{i-1}} + C_{w_i}\right) + \frac{R_{w_{i-1}} \cdot C_{i-1} \cdot \left(0.5 \cdot C_{w_{i-1}} + C_{w_i}\right)}{\tau} + p_{w_i} \cdot C_{i-1}\right) \\
&a_3 = 0 \\
&a_4 = -\left(g_i \cdot C_{w_i} \cdot C_{i-1} \cdot \left(C_{i+1} + C_{w_i}\right)\right) .
\end{aligned} \tag{37}$$

According to Descartes Rule of Signs [19] the polynomial has a single positive real root, which can be solved iteratively, similarly to the ULE delay minimization technique.

VIII. RESULTS

Verification of the proposed ULE model and optimization techniques is achieved by several benchmark circuits and a comparison of the ULE results with an industrial CAD optimizer. The ULE optimization are compared to the results of Cadence Virtuoso® Analog Optimizer. The Analog Optimizer uses LSQ (least square) and CFSQP (C version Feasible Sequential Quadratic Programming) algorithms to determine the value of the design variables that satisfy the design goals [21]. Optimal solution is achieved by detecting the sensitivity of the goal expression to each design variable, and iteratively changing the variables and

performing circuit simulations. The design variables used by Analog Optimizer are the size of the gates in the critical path where the design goal is either minimizing the delay or the power-delay product.

Two benchmark circuits are considered – (a) a four-bit carry-lookahead adder and (b) a four-bit ripple-carry adder. The critical paths of the two circuits are shown in Fig. 12 and Fig. 11, respectively. The simulations are performed using a 65 nm CMOS technology [20].

The circuits are characterized in terms of delay, area and power consumption for different wire lengths. For each scenario, optimization is performed by sizing the gates along the critical path according to the ULE technique. The results of each optimization are compared with the results of the Analog Optimizer tool.

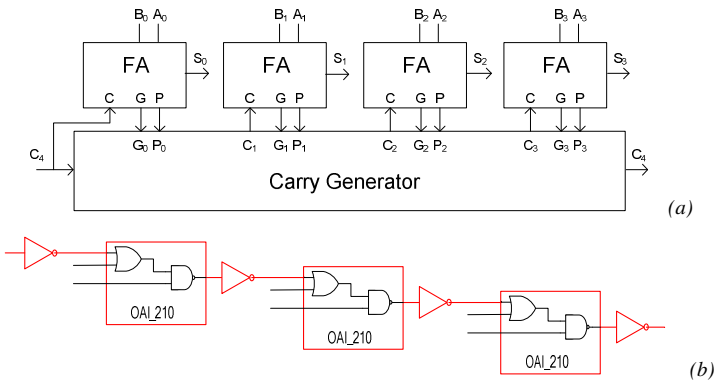


Fig. 11. The structure of a four-bit carry-lookahead adder (a) and a seven-gate critical path. For a P/N scaling ratio k=1.72, the logical effort is ginv=1 and gOAI210=2.

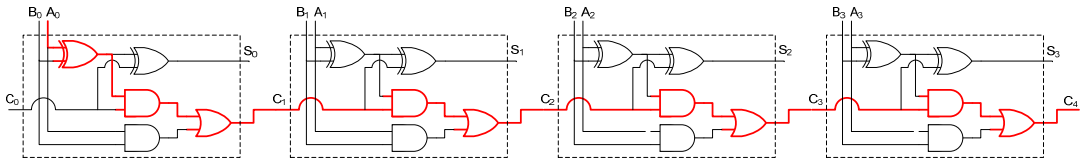


Fig. 12. The structure of a four-bit ripple-carry adder. The nine-gate critical path is the carry signal propagating through the full-adder units (highlighted). For a P/N scaling ratio k=1.72, the logical effort is gXOR2=3.63, gAND2=1.37 and gOR2=1.63.

Delay Minimization

ULE delay minimization is performed in both benchmark circuits according to (22). In each circuit, the optimization techniques are applied for different wire lengths and physical parameters.

In a carry-lookahead adder, the optimization techniques were applied for three scenarios:

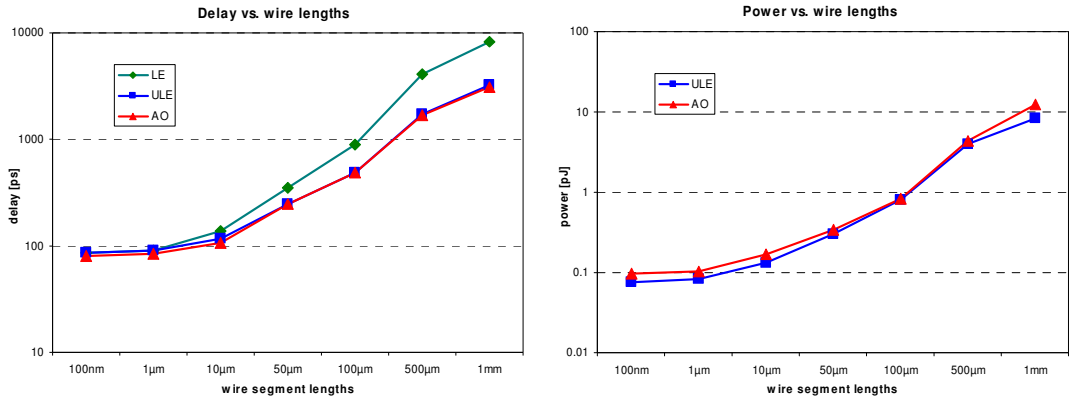
- Equal interconnect segments of local wires with lengths of 100 nm, 1 µm, 10 µm, and 50 µm.
- Equal interconnect segments of global wires with lengths of 100 µm, 500 µm, and 1mm.
- Unequal interconnect segments with different wire lengths.

A comparison of results for the delay is presented below (Fig. 13). At the first stage the results of ULE and Analog Optimizer are compared with the regular LE technique which does not consider the wires.

The comparative results of delay after the ULE, Analog Optimizer and LE optimizations are shown in Fig. 13a. Note that the delay after the ULE optimization is close to the results achieved by Analog

Optimizer tool. Also note that the standard LE technique becomes increasingly inaccurate as the wire lengths grows.

The resulting optimal circuits created by the ULE and Analog Optimizer techniques are also compared in terms of energy consumption, as shown in Fig. 13b. In the case of equal RC wire segments for the carry-lookahead adder, the ULE technique results in lower energy consumption than the Analog Optimizer. This behavior is achieved by smaller gate sizes allocated during the ULE optimization process.



(a) *(b)*

Fig. 13. Delay (a) and power consumption (b) in a carry-lookahead adder for various wire segment lengths after LE, ULE, and Analog Optimizer.

A comparison of the ULE and Analog Optimizer results is listed in Table 1. The ULE technique outperforms the Analog Optimizer tool in terms of energy consumption, where the delay is within 9% as compared to the Analog Optimizer.

Wire lengths	100nm	1µm	10µm	50µm	100µm	500µm	1mm
Δ Delay	6.64%	7.11%	8.44%	1.56%	1.04%	1.51%	3.67%
Δ Energy	-15.6%	-18.0%	-23.5%	-9.5%	-3.6%	-7.9%	-33.2%

Table 1. Difference between delay and energy from application of ULE and Analog Optimizer for a carry-lookahead adder.

The carry-lookahead adder circuit is also optimized for unequal wore segments, where two sets of wire length were used, *S1= [100, 0.1, 10, 50, 100, 500] µm* and *S2 = [10, 500, 0.1, 10, 0.1, 1000] µm*. The resulting delay and energy characteristics are summarized in Table 2. The results of ULE are close to Analog Optimizer in both delay and energy.

	Delay [ps]			Energy [pJ]		ULE vs. AO	
scenario	ULE	AO	LE	ULE	AO	Δ delay	Δ energy
S1	400.1	394.7	556.5	0.773	0.798	1.35%	-3.42%
S2	321.9	310.0	847.7	1.58	1.64	3.70%	-3.88%

Table 2. Optimization results for different scenarios of unequal wire segments in carry-lookahead adder circuit.

Run Time Comparison in ULE vs. Analog Optimizer

The low complexity and fast run time of ULE could make it a competitive alternative for integration in the EDA toolset for optimization of long and complex logic structures with interconnect. The ULE and

Analog Optimizer are compared in Fig. 14 in terms of the run time as a function of the logic path length. Both techniques are used to optimize the critical path in a ripple carry adder with varying number of full adder stages. The Analog Optimizer run time is orders of magnitude longer than the run time of ULE.

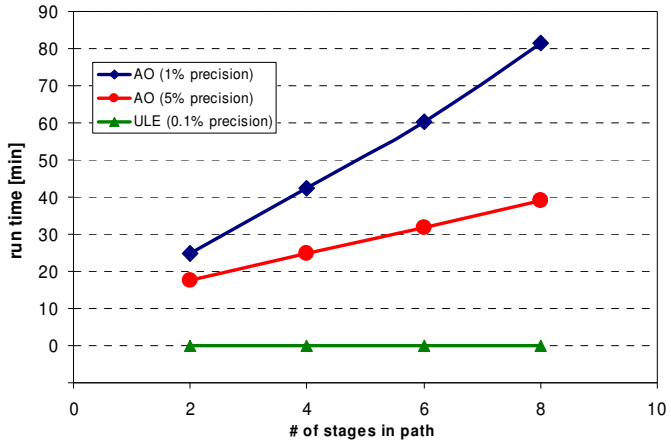


Fig. 14. Comparison of optimization run time of Analog Optimizer and ULE for various numbers of full-adders in ripple carry adder path. ULE run time is shorter than 1 second.

Delay Minimization using Repeaters with ULE sizing

The delay of a logic path with long wires can be further reduced by inserting inverters along the path. As a test case, repeaters are inserted into the critical path of a ripple-carry adder, assuming that the adder cells are connected by wire segments, and the wire length is a parameter. In the first numerical experiment, a repeater is inserted in the middle of each wire. In the second experiment each wire is split into three equal parts by inserting two repeaters. ULE optimization is performed on the combined circuit to produce the optimal gate sizes. The scaling factor of each repeater is determined by ULE, contrary to standard fixed-size repeater insertion. The delay of the path as function of wire length is illustrated in Fig. 15. Repeater insertion significantly reduces the delay for long wires. However, when too many repeaters are used per wire, the delay increases (e.g. two repeaters produce higher delay than a single repeater for a 100 μm wire as shown in Fig. 15).

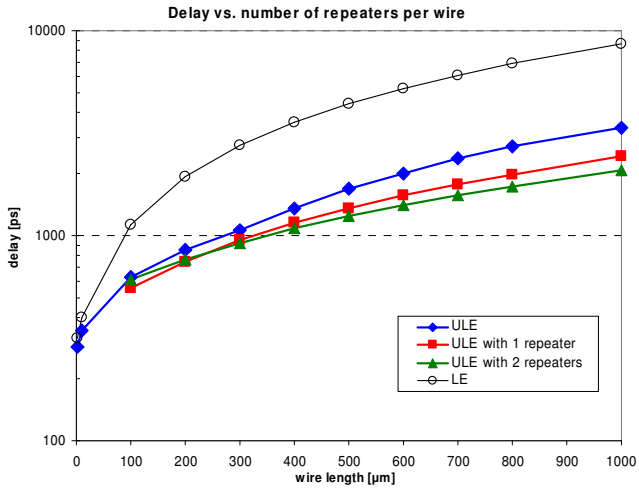


Fig. 15. Delay minimization in a ripple carry adder. Circuit simulation results are shown, validating the effectiveness of the ULE model and the optimization method. The horizontal axis depicts the length of each wire connecting every pair of adder cells. The curves correspond to LE sizing (gates sizes determined by the standard Logical Effort method), ULE sizing (gate sizes determined by equation (22)), ULE sizing with a single repeater per wire, and ULE sizing with two repeaters per wire.

ULE Gate Sizing for Power-Delay Product Minimization

Power-delay optimization is performed in ULE based on (37). The results of the optimization are compared with the results of Analog Optimizer in which power-delay product minimization is the design goal. In each circuit, the simulations are performed for equal 1 mm wires and for unequal wires. The comparative results of power delay optimization in a carry-lookahead adder are listed in Table 3. ULE maintains delays close to the Analog Optimizer, while exhibiting a significant advantage in terms of power-delay product.

	Delay [ps]		Power-Delay [$\times 10^{-16}$]		Area-Delay [$\times 10^{-24}$]	
Wires	ULE	AO	ULE	AO	ULE	AO
1 mm	3340	3217	4830	9410	899	2350
unequal	457	511	244	262	22.4	22.0

Table 3. Power-delay optimization for different scenarios of wire segments in a carry-lookahead adder circuit using ULE and Analog Optimizer. Unequal wire segments L=[100, 200, 150, 80, 120, 50] µm.

A comparison of the power-delay optimization in a ripple-carry adder is listed in Table 4. The ULE results are close to the results of Analog Optimizer, while showing a significant advantage in the power-delay product for unequal wire segments.

	Delay [ps]		Power-Delay [$\times 10^{-16}$]		Area-Delay [$\times 10^{-24}$]	
Wire segments	ULE	AO	ULE	AO	ULE	AO
1 mm	3800	4410	2158	1760	9750	9960
unequal	3480	3316	1120	3192	3200	4650

Table 4. Power-delay optimization for different wire segments in a ripple-carry adder using ULE and Analog Optimizer. Unequal wire segments L=[1000, 100, 150, 300, 800, 200, 400, 150, 50] µm.

The power reduction obtained in ULE is due to the reduced gate size from (37) as compared to the size of from (22). An example graph comparing the sizing profiles resulting from ULE delay minimization and ULE power-delay minimization in a ripple-carry adder with 1 mm wires is shown in Fig. 16.

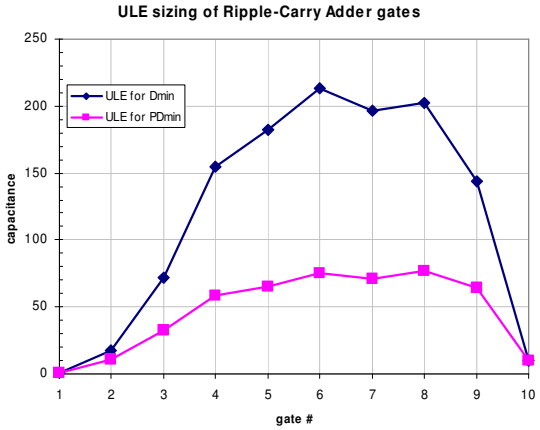


Fig. 16. Optimal gate sizing of a ripple-carry adder with 1 mm wire segments resulting from ULE delay minimization and from ULE power-delay minimization.

IX. SUMMARY

Delay minimization in logic paths with wires is an important issue in the VLSI circuit design process. The interconnect is now the dominant factor in performance-driven circuits and must be explicitly considered throughout the design process. The characteristics of the wires are not correlated with the characteristics of the gates; thereby not permitting the use of the standard logical effort model. In fact, optimal gate sizing in the presence of interconnect does not correspond to equal effort of all of the stages along the path.

The Unified Logical Effort (ULE) method is proposed here for delay evaluation and minimization of logic paths with general gates and *RC* wires. The ULE method provides closed-form conditions to achieve minimum delay. The ULE method converges to the LE conditions in the case of zero interconnect, yielding equal sized gates when long wires are considered, as in repeater insertion methodologies.

Optimal gate sizing is provided by the ULE method, making the proposed ULE method suitable for both manual calculations as well as for integration into CAD tools. The technique is applied to several example logic paths, permitting the influence of the wire length, gate type, and technology parameters to be evaluated.

The ULE optimization is compared with the industrial Analog Optimizer tool, showing close results in terms of delay and power consumption. Due to its simplicity, the computational run time of the ULE optimization is several orders shorter than the reference tool. This enhanced efficiency with similar accuracy demonstrates the high potential of ULE for integration into EDA tools.

The ULE method can be combined with known heuristics for buffering and repeater insertion. This combination is effective due to the fixed wire lengths dictated in many design flows. Further research is required to develop closed-form solutions that combine simultaneous optimal gate sizing with wire segmentation.

ACKNOWLEDGEMENTS

The authors thank Nathanaelle Gibrat-Wormser and David Pedahel for contributing to the ULE evaluation.

REFERENCES

[1] I. Sutherland, B. Sproull and D. Harris, *Logical Effort - Designing Fast CMOS Circuits*, Morgan Kaufmann Publishers, 1999.

[2] ——, Section 10.4, Interconnect, p. 175, 1999.

[3] H.B. Bakoglu, Circuits, *Interconnections and Packaging for VLSI*, Adison-Wesley, pp. 194-219, 1990.

[4] H.B. Bakoglu and J.D. Meindl, "Optimal Interconnection Circuits for VLSI," *IEEE Transactions on Electron Devices*, pp. 903-909, 1985.

[5] A. Nalamalpu and W.Burleson, "Repeaters Insertion in Deep Submicron CMOS: Ramp-based Analytical Model and Placement Sensitivity Analysis," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 766-769, May 2000.

[6] V. Adler and E.G. Friedman, "Repeater Design to Reduce Delay and Power in Resistive Interconnect," *IEEE Transactions on Circuits and Systems – II*, vol. 45, no. 5, pp. 607-616, 1998.

[7] B. Lasbouygues et al., "Logical Effort Model Extension to Propagation Delay Representation," *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 2006.

[8] A. Kabbani, D. Al-Khalili and A.J. Al-Khalili, "Delay Analysis of CMOS Gates Using Modified Logical Effort Model," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 6, pp. 937-947, June 2005.

[9] S. Srinivasaraghavan and W. Burleson, "Interconnect Effort - A Unification of Repeater Insertion and Logical Effort," *Proceedings of the IEEE Computer Society Annual Symposium on VLSI*, pp. 55-61, February 2003.

[10] H.C. Lin and L.W. Linholm, "An Optimized Output Stage for MOS Integrated Circuits," *IEEE Journal of Solid-State Circuits*, vol. SC-10, no. 2, pp.106-109, April 1975.

[11] R.C. Jaeger, "Comments on 'An Optimized Output Stage for MOS Integrated Circuits'," *IEEE Journal of Solid-State Circuits*, vol. SC-10, no. 2, pp.185-186, June 1975.

[12] B.S. Cherkauer and E.G Friedman, "Design of Tapered Buffers with Local Interconnect Capacitance," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 2, pp. 151-155, February 1995.

[13] B.S. Cherkauer and E.G Friedman, "A Unified Design Methodology for CMOS Tapered Buffers," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 3, no. 1, pp. 99-111, March 1995.

[14] S.R. Vemuru and A.R. Thorbjornsen, "Variable-Taper CMOS Buffer," *IEEE Journal of Solid-State Circuits*, vol. 26, no. 9, pp. 1265-1269, September 1991.

[15] K. Venkat, "Generalized Delay Optimization of Resistive Interconnections through an Extension of Logical Effort," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 2106-2109, May 1993.

[16] M. Moreinis, A. Morgenshtein, I. Wagner and A. Kolodny, "Logic Gates as Repeaters (LGR) for Area-Efficient Timing Optimization," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 14, no. 11, pp. 1276- 1281, November 2006.

[17] C. Chu and D. F. Wong, "Closed Form Solution to Simultaneous Buffer Insertion / Sizing and Wire Sizing," *ACM Transactions on Design Automation of Electronic Systems*, vol. 6, no. 3, pp. 343-371, July 2001.

[18] W. C. Elmore, "The Transient Response of Damped Linear Networks with Particular Regard to Wide Band Amplifiers," *Journal of Applied Physics*, vol. 19, no. 1, pp. 55-63, January 1948.

[19] B. Anderson, J. Jackson, and M. Sitharam, "Descartes' Rule of Signs Revisited," *The American Mathematical Monthly*, v. 105, no. 5, p. 447-451, May 1998.

[20] Predictive Technology Model (PTM), http://www.eas.asu.edu/~ptm/ .

[21] Virtuoso Advanced Analysis Tools User Guide, http://www.ece.uci.edu/eceware/cadence/aatoolsuser/chap3.html

APPENDIX A

The power-delay product (while neglecting parasitic impedances) is

$$\begin{aligned} PD &= \left(g_i \cdot \left(h_i + \frac{C_{w_i}}{C_i} \right) + \frac{R_{w_i} \cdot \left(0.5 \cdot C_{w_i} + h_i \cdot C_i \right)}{\tau} + g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{h_i \cdot C_i} \right) + p_{w_{i+1}} \right) \cdot \left(C_i \cdot h_i + C_{w_{i+1}} \right) = \\ &= h_i^2 \cdot \left(g_i \cdot C_i + \frac{R_{w_i} \cdot C_i^2}{\tau} \right) + h_i \cdot \left(g_i \cdot C_{w_i} + \frac{0.5 \cdot R_{w_i} \cdot C_{w_i} \cdot C_i}{\tau} + p_{w_{i+1}} \cdot C_i + g_i \cdot C_{w_{i+1}} + \frac{R_{w_i} \cdot C_i \cdot C_{w_{i+1}}}{\tau} \right) + \\ &+ \left(g_{i+1} \cdot \left(C_{i+2} + C_{w_{i+1}} \right) + g_i \cdot \left(\frac{C_{w_i} \cdot C_{w_{i+1}}}{C_i} \right) + \frac{0.5 \cdot R_{w_i} \cdot C_{w_i} \cdot C_{w_{i+1}}}{\tau} + p_{w_{i+1}} \cdot C_{w_i} \right) + \\ &+ \frac{1}{h_i} \cdot \left(g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{C_i} \right) \cdot C_{w_{i+1}} \right) \quad . \end{aligned} \tag{38}$$

The optimal condition is determined by equating the derivative of (38) to zero,

$$\begin{aligned} &\frac{\partial PD}{\partial h_i} = h_i \cdot 2 \left(g_i \cdot C_i + \frac{R_{w_i} \cdot C_i^2}{\tau} \right) + \left(g_i \cdot C_{w_i} + \frac{0.5 \cdot R_{w_i} \cdot C_{w_i} \cdot C_i}{\tau} + p_{w_{i+1}} \cdot C_i + g_i \cdot C_{w_{i+1}} + \frac{R_{w_i} \cdot C_i \cdot C_{w_{i+1}}}{\tau} \right) - \\ &- \frac{1}{h_i^2} \cdot \left(g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{C_i} \right) \cdot C_{w_{i+1}} \right) = 0 \quad . \end{aligned} \tag{39}$$

$$\begin{aligned} &h_i^3 \cdot 2 \left(g_i \cdot C_i + \frac{R_{w_i} \cdot C_i^2}{\tau} \right) + h_i^2 \left(g_i \cdot C_{w_i} + \frac{0.5 \cdot R_{w_i} \cdot C_{w_i} \cdot C_i}{\tau} + p_{w_{i+1}} \cdot C_i + g_i \cdot C_{w_{i+1}} + \frac{R_{w_i} \cdot C_i \cdot C_{w_{i+1}}}{\tau} \right) - \\ &- \left(g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{C_i} \right) \cdot C_{w_{i+1}} \right) = 0 \\ &\frac{C_{i+1}^3}{C_i^3} \cdot 2 \left(g_i \cdot C_i + \frac{R_{w_i} \cdot C_i^2}{\tau} \right) + \frac{C_{i+1}^2}{C_i^2} \left(g_i \cdot C_{w_i} + \frac{0.5 \cdot R_{w_i} \cdot C_{w_i} \cdot C_i}{\tau} + p_{w_{i+1}} \cdot C_i + g_i \cdot C_{w_{i+1}} + \frac{R_{w_i} \cdot C_i \cdot C_{w_{i+1}}}{\tau} \right) - \\ &- \left(g_{i+1} \cdot \left(\frac{C_{i+2} + C_{w_{i+1}}}{C_i} \right) \cdot C_{w_{i+1}} \right) = 0 \quad . \end{aligned} \tag{40}$$

The final expression for the optimal gate capacitance is described by the following polynomial,

$$C_{i+1}^3 \cdot a_1 + C_{i+1}^2 \cdot a_2 + C_{i+1} \cdot a_3 + a_4 = 0 \,, \tag{41}$$

where the coefficients are:

$$\begin{aligned} a_1 &= 2 \left(\frac{g_i}{C_i^2} + \frac{R_{w_i}}{C_i \cdot \tau} \right) \\ a_2 &= \left(\frac{g_i \cdot \left(C_{w_i} + C_{w_{i+1}} \right)}{C_i^2} + \frac{R_{w_i} \cdot \left(0.5 \cdot C_{w_i} + C_{w_{i+1}} \right)}{C_i \cdot \tau} + \frac{p_{w_{i+1}}}{C_i} \right) \\ a_3 &= 0 \\ a_4 &= - \left(g_{i+1} \cdot C_{w_{i+1}} \cdot \frac{C_{i+2} + C_{w_{i+1}}}{C_i} \right) \quad . \end{aligned} \tag{42}$$

The expression can be simplified and rewritten for the gate capacitance C_i, resulting in (37).