EWA: Exact Wiring-Sizing Algorithm[†]

Rony Kay, Gennady Bucheuv and Lawrence T. Pileggi
Carnegie Mellon University
Department of Electrical and Computer Engineering
Pittsburgh, PA 15213

Abstract

The wire sizing problem under inequality Elmore delay constraints is known to be posynomial, hence convex under an exponential variable-transformation. There are formal methods for solving convex programs. In practice heuristics are often applied because they provide good approximations while offering simpler implementation and better efficiency. There are methods for solving related problems, which are comparable to heuristics from efficiency point of view, but they solve a less desirable formulation in terms of the objective function and constraints.

In this paper the EWA algorithm is described. It solves the problem of minimizing the wiring area or capacitance of an interconnect tree subject to constraints on the Elmore delay. EWA is simple to implement and its efficiency is comparable to the available heuristics. No restrictions are placed on the circuit or wire widths, e.g., non-monotone wire widths assignment solutions are feasible. We prove that the optimal wire width assignment for a minimum wiring area objective satisfies all the delay constraint as equalities, when minimum wire width constraints are relaxed. It follows that EWA can be applied also for problems with equality delay constraints such as clock trees. This and other described properties are general enough to permit extensions to higher order delay models in the future.

1.0 Introduction

The interconnect wiring between gates on a VLSI chip can dominate the performance for today's technologies. It has been reported that in some designs up to 70% of the overall path delays are attributed to the interconnect, while the manufacturing trends suggest that the interconnect delay will be even more dominant in the foreseeable future. Minimizing the wiring length has always been a primary objective of routing algorithms to control the chip area, but now it must be done to satisfy tight performance requirements too. Numerous techniques have been developed to construct interconnect trees and optimize their length. When the interconnect resistance and capacitance are dominating the IC performance, it is advantageous to exploit also the wire widths as an additional degree of freedom for performance optimization.

Clock trees were wire width optimized in [14] to demonstrate the improvement in controlling the delay and desentizing the skew to inevitable process variations. The benefits of wire sizing for signal nets was demonstrated in [3]. Various strategies for wire width optimization have followed for both clock trees and signal nets. Most wire sizing techniques optimize either wiring delay or wiring area subject to constraints on the other, or a weighted combination of delay and area. Some heuristic approaches can be applied simultaneously with the tree construction, while others are used for post-processing a given tree topology. Refer to [2][3][14][18][10][21] for examples of each type of approach.

This paper describes a wire-sizing algorithm for post-processing an interconnect tree with efficiency and simplicity that are comparable to the heuristic methods. EWA finds the solution for minimizing the total wiring area subject to constraints on the Elmore path delays, as well as upper and lower bounds on wires width. The delay constraints can be inequalities or equalities, hence EWA can be applied either for signal nets or for clock nets. We propose a two-phase methodology for first targeting delays, then reaching them in terms of minimum area.

EWA is based on several new observations and proofs for properties of RC interconnect trees. We prove that the solution to the problem of minimum area subject to inequality constraints on target delays must be on the delay boundary with respect to all the delay constraints, when minimum wire width constraints are relaxed. Thus the solution subject to inequality constraints on the path delays is equivalent to the solution with equality constraints on the path delays.

We review some related work in Section 2.0. An overview of our two-phase wire sizing methodology is described in Section 3.0. The wire sizing circuit, delay model, and some necessary terminology is outlined in Section 4.0. The formulation of the wire sizing problem is given in Section 5.0. Key theorems and new observations are made in Section 6.0, followed by a description of the EWA algorithm in Section 7.0. Some experimental results are given in Section 8.0. Conclusions are drawn in Section 9.0, along with some comments about extending EWA to handle more accurate delay models.

2.0 Background

This section contains a brief overview of related work. It is not intended as an exhaustive survey. For a detailed review of previous work on wire sizing refer to [2].

2.1 Signal Nets

Cong and Leung [3][4] proposed a wire sizing algorithm for signal nets under the Elmore delay model. Because it lends itself to an elegant solution, they considered a minimization of a weighted sum of critical delays to the leaf nodes (sinks); the critical paths must be given *a-priori*. Special properties of interconnect trees - separability, monotonicity, and dominance - were used to develop an

[†] This work was supported in part by the Semiconductor Research Corporation under contract 97-DC-068.

$O(n^r)$ dynamic programming algorithm, where n is the number of segments and r is the cardinality of a discrete width range. Based on the theoretical analysis of properties of RC trees, greedy heuristics were also proposed. The heuristics can be useful in early phase of the design cycle.

Fishburn and Dunlop showed that the Elmore delay is a posynomial function of transistor widths when applied to transistor sizes optimization in [7] (TILOS). Given that posynomial programs can be cast as convex optimizations under the exponential variable transformation, it was shown by TILOS that heuristics can obtain good approximations using less run-time and requiring simpler implementations. Exact solutions for the continuous sizing are possible if one is willing to resort to geometric programming [5] or to convex programming in the transformed domain.

Sapatnekar *et. al* solved the same problem as TILOS using convex programming in [16]. More recently Sapatnekar applied a similar convex programming approach in [18] to solve the minimization of the critical path delay by wire sizing under the Elmore delay model. A sensitivity-based heuristic was considered in [19] and compared to the results from convex programming. It was empirically observed that the results of the exact method and the heuristic are very well correlated.

Menezes *et. al* showed in [10] that also higher order moments can be cast as posynomials, which allowed them to optimize transfer functions under higher order delay models. Later, a Sequential Quadratic Programming (SQP) approach was applied in [11] to optimize driver and wire sizes that satisfy the target delays of critical leaf nodes. The SQP approach converges to the global minimum in terms of the Elmore delay model, and to a local minimum in terms of a higher order delay model using RICE[16]. It was shown that applying higher order delay model yields superior results.

2.2 Clock Trees

Unfortunately, geometric (posynomial) programming does not apply directly to clock tree problems because the associated formulation includes only inequality constraints. Pullela, Menezes and Pileggi proposed a wire sizing approach for clock trees in [14] that used a simple heuristic based on one-wire-at-a-time downhill improvements. Later, Zhu, Dai and Xi proposed a least-squares approach which considered several wires concurrently [21], but in spite of the increased runtime and complexity they could not prove convergence to the global optimum either. Since least-squares approaches can be costly in terms of runtime due to building and factoring a matrix of sensitivities, heuristics were used to sparsify the sensitivity matrix in [15].

It is generally more difficult to achieve exact solutions for equality constraints, hence clock tree problems. It should be noted that such constraint formulations can have advantages for signal nets too. The advantages should become more clear in the subsequent sections.

3.0 Proposed Wire Sizing Methodology

The formulations for RC tree wire-sizing optimization generally fall into one of three categories: (i) minimize the maximum path delay subject to restrictions on the available area resources; (ii) minimize the wiring area subject to upper bound constraints on the path delays; (iii) minimize a combination (weighted product or weighted sum) of delays and area. Each of those formulations captures important aspects of the overall design goals.

Clearly, there is a trade off between delay and area of the interconnects [3][18][11], a qualitative illustration of which is shown in Fig. 1. From the figure it is apparent that to achieve the minimum delay possible it is necessary to use extensive area resources. Analogously, too much concern about area resources results in a poor performance. A weighted sum or a weighted product of delay and area is able to capture a trade off relation somewhat, however, assigning good weights is based on a trial and error.

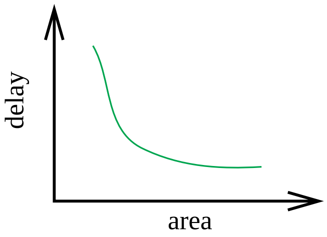


FIGURE 1. Typical curve of delay versus area tradeoff.

The designer’s goal is to comply with the design specification while minimizing resources utilization, yet the design specification is driven by the best performance that can be achieved in a reasonable cost. For this reason we believe that the best methodology for width optimization (of interconnects or transistors) is based on a two-phase approach. Phase I is comprised of a delay minimization step, not necessarily to the optimal solution but to provide a measure of how well we can do. Phase II is the more important phase, where delay bounds that are larger than the delays achieved at Phase I are imposed as constraints for an area minimization.

The purpose of Phase I is to ensure that there is a feasible solution for Phase II and to provide orientation with respect to the quantitative behavior of the area-delay curve (Fig. 1). Further trade-off analysis can be made by varying the delay targets. An important advantage of the two phase approach is that it supports the engineering decision of trading area resources for performance in a more practical and intuitive way then a weighted objective function.

4.0 Circuit and Delay Model

4.1 Overview

In this paper the interconnect is assumed to have an electrical model in the form of an RC tree, as shown in Fig. 2. The Elmore delay[6] is used to estimate path delays. The Elmore path delay from each leaf node to the root is calculated by summing up the individual contributions of wire segments along the path. The contribution to the delay of path P_i by a segment j (on the path) is the product of the segment’s resistance R_j and the down stream capacitance Cds_j. Cds_j is the total capacitance of the wire segments and external capacitance loads which are descendants of j. Denoting *ds(j)* as the set of segments and leaf nodes downstream of j, we can calculate the delay D_i of path P_i as,

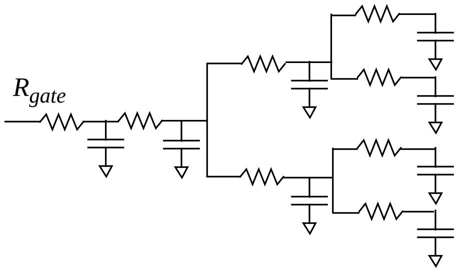


FIGURE 2. Lumped equivalent RC tree model.

$$D_i = \sum_{j \in P_i} R_j \cdot Cds_j = \sum_{j \in P_i} R_j \cdot \sum_{k \in ds(j)} C_k \qquad (1)$$

We use a lumped RC tree model of the interconnect as illustrated in Fig. 2. A Π-model is used to represent any *Uniform RC* (URC) wire segment. While a more accurate delay model may benefit from chopping the URC into multiple lumped sub-sections, it will not change the value of the Elmore delay. That is, *a Π-model yields the precise Elmore delay when used to replace any URCs*. This statement stems from examining the Elmore delay of an URC that drives a downstream load *Cds*; Dividing an URC with resistance *R* and capacitance *C* into *n* Π-model segments, the incremental contribution (ΔD) of the wire to the overall Elmore-path-delay is

$$\Delta D = R \cdot Cds + \frac{R \cdot C}{n^2} \cdot \sum_{i=1}^{n} \left(i - \frac{1}{2} \right) = R \cdot Cds + \frac{RC}{2} \qquad (2)$$

From (2), it is apparent that ΔD is independent of the number of Π sections used to model the URC.

The resistance and capacitance of each wire segment are dependent on three technology based parameters: (i) resistance per square r_s ($\Omega/\Box$), (ii) capacitance per unit area c_a (fF/μm^2), and (iii) fringe capacitance per unit length c_f (fF/μm). The resistance R_i and capacitance C_i of a wire segment *i* with length l_i and width w_i are calculated using the formulas:

$$R_i = r_s \cdot \frac{l_i}{w_i}; \quad C_i = c_a \cdot l_i \cdot w_i + c_f \cdot l_i \qquad (3)$$

Wires on the same layer have similar parameters. Each layer is associated with different parameters due to differences in metal thickness, spacing, and dielectric constants. Using a layer assignment and a technology file, equations (1) and (3) are combined to express the Elmore-path-delays as a function of wire widths.

5.0 Formulation

As stated above, our objective (Phase II from Section 3.0) is to minimize the total wiring area subject to upper bound constraints on each path delay, as well as upper and lower bounds on wire widths. Representing the optimization variables in terms of a vector of wire segment widths $\vec{w}$, the optimization is formally stated as:

$$\begin{array}{lll} \textit{minimize} & \left\{ A(\vec{w}) = \sum_{i=1}^{|S|} l_i \cdot w_i \right\} & \\ \textit{subject to:} & ED_i(\vec{w}) \le TD_i & i = 1 \ldots |P| \\ & w^L{}_i \le w_i \le w^U{}_i & i = 1 \ldots |S| \end{array} \qquad (4)$$

Where $|S|$ and $|P|$ denote the number of wire segments and the number of source-sink paths respectively; $ED_i(\vec{w})$ is an estimator of the *i*-th path delay as a function of the widths; TD_i is the target delay for path *i*; $w^U{}_i$ and $w^L{}_i$ are the upper and lower bounds on the width of segment *i*.

Using the Elmore delay model, $ED_i(\vec{w})$ is obtained by substituting (3) into (1):

$$ED_i(\vec{w}) = \sum_{\substack{\forall j,k \\ k \in ds(j)}} \alpha_{jk} \cdot \frac{w_k}{w_j} + \sum_j \beta_j \cdot \frac{1}{w_j} \qquad (5)$$

where α_{jk} and β_j are positive constants and *ds(j)* is the set of wire segments and leafs downstream (descendants) of segment *j*. It is worth noting that the constants depend only on the technology parameters, the fixed capacitance loads, and the lengths of the wire segments.

5.1 Posynomiality

Posynomial programming is a branch of optimization theory [5]. A posynomial is a function *p* of positive vector $\vec{x}$, having the form

$$p(\vec{x}) = \sum_i u_i(\vec{x}) \qquad (6)$$

with

$$u_i(\vec{x}) = b_i \cdot \prod_j (x_j)^{a_j} \qquad (7)$$

where the exponents a_j are real numbers and the coefficients b_i are positive real numbers. A posynomial program is a minimization problem where the objective function is posynomial and the constraints are posynomial inequalities that are less or equal to 1. Using a variable substitution $x_i = EXP(z_i)$, a posynomial program can be easily transformed to a convex program. Consequently, any local minima is a global minima and formal methods of convex programming can be applied to find the minimum.

The Elmore delay (5) is a posynomial. In addition, the range constraints on the wire widths can be expressed as posynomial constraints: $w^L{}_i \le w_i \le w^U{}_i$ is equivalent to $\frac{w_i}{w^U{}_i} \le 1$ and $\frac{w^L{}_i}{w_i} \le 1$.

It follows that under the Elmore delay model the formulation of our problem (Eqn. (4)) is a posynomial program in the original domain, which is equivalent to a convex program in the transformed domain.

We exploit the posynomiality when we describe the algorithm in Section 7.0. In the next section we prove some interesting properties of the problem of interconnect area minimization.

6.0 Theoretical Analysis

In this section we analyze special properties of the wire sizing problem that have practical and theoretical relevance. It is worth noting that the analysis is applicable also in a broader context of resource allocation, *e.g.* transistors sizing and concurrent wire and gate sizing.

6.1 Preliminaries

We begin by defining some necessary terminology:

> Definition 1: *the delay boundary is an arc in the feasible space for which all inequality delay constraints are satisfied as equalities (equivalently stated, all delay constraints are active).*
>
> Definition 2: *a wire segment has negative sensitivity with respect to the area minimization (4) if we can decrease*

its width by an arbitrarily small amount without violating any constraints.

We also formalize a simple observation which is useful for understanding the subsequent proofs.

Lemma 1: By decreasing the width of a segment i, the path delays from the root to all the leaf nodes which are not descendants of i do not increase.

proof: the Elmore expressions of the path delays from the root to the leaf nodes which are not descendants of segment *i* either do not include *i*, or they include *i* as a capacitive load only. Smaller capacitive loads cannot increase the delay (plain algebra). QED

6.2 The boundary property

In cases that the optimal solution has no wires at their minimum allowable width, we prove that at the optimal solution all the delay constraints are active. Observe that if we relax the minimum wire width constraints (set $w^L{}_i$ to zero) in (4), then this property is always true.

Theorem 1 (the boundary property): if at the optimal solution no minimum wire width constraint is active, then the solution is on the delay boundary.

proof: assume, for sake of contradiction, that $\vec{w}'$ is the global optimum and there exists a constraint *c* which corresponds to path *i* that has a positive slack (*i.e.* not satisfied as an equality). Let us decrease the width of the leaf wire segment of path *i* until an equality is achieved for the constraint *c*. By lemma 1, all the other root-to-leaf path delays do not increase, therefore the new configuration is feasible. Furthermore, the total area of the tree is smaller then the area of $\vec{w}'$, hence there is a feasible solution with smaller value of the objective function and $\vec{w}'$ is not the optimum, contradiction. QED

It can be shown that the boundary property holds for any reasonable delay model. For example, it can be shown for a higher order dominant pole model[13] using adjoint sensitivities[9].

6.3 Sensitivity Based Properties

The following properties are related to the sensitivity of a solution with respect to wire widths. By definition, as long as some width configuration $\vec{w}$ contains segments with negative sensitivity, an improvement step is possible by decreasing the size of one (or more) wires with negative sensitivity. Moreover, if we are not on the delay boundary, there exists a wire segment (*e.g.* leaf segment) whose width can be decreased to reduce the overall wiring area while maintaining the feasibility with respect to the delay constraints.

It is useful to define not only the sign of the sensitivities but also their magnitude. A change of a specific wire width might either increase or decrease the Elmore delay to the down-stream leaf nodes. However, by inspection of the algebraic expression of the Elmore delay it is evident that increasing (decreasing) the width results always with increased (decreased) delays to the other (*i.e.* not descendant) leaf nodes. Therefore we define the magnitude of the sensitivity to be proportional to the change in the delay to descendant leaf nodes (cone of influence). Since the lengths of wires may differ, we divide it by the wire length in order to get a sensitivity that is normalized with respect to an area unit rather then a width unit. Sometimes it is necessary to minimize capacitance rather then the area. It is worth noting that the only change to the mathematical program being solved is in the constant coefficients of the objective function. For such objective function we divide the magnitude also by the capacitance per area parameter to get the sensitivity of the delay to downstream leaf nodes with respect to a capacitance unit. More formally, for the minimum capacitance objective the magnitude of the sensitivity of segment i is actually the partial derivative: $\frac{1}{l_i \cdot c_{a_i}} \cdot \frac{\partial}{\partial w_i} ED_j(\vec{w})$, where $ED_j(\vec{w})$ is the delay to any down-stream leaf segment.

Let us consider a width configuration on the delay boundary such that no single wire can be decreased while maintaining the feasibility. Assume that the objective is to minimize the total capacitance of the interconnect tree. It can still be possible to reduce the objective function if there is a pair of segments on a path from the root to leaf such that the sensitivity of the upstream segment is larger. It stems from the fact that the sub-tree rooted at the upper segment contains the subtree rooted at the lower segment; increasing the upper wire width is compensated by a larger decrease of the lower wire without violating any delay constraint. Thus a feasible point with lower value of the objective function can be obtained.

The intuition is as follows. Since the transformed domain is convex, we can move between any two feasible points by making subsequent steps in axis parallel directions, *i.e.* resizing a single segment at a time. Good directions involve either a single width decrease, or a pair of segments, or a more complex step such that the objective function is reduced without violating the target delays. An obvious choice would be to increase the width of a wire that has the smallest cost for a given decrease in delay, and to decrease the width of the wire that brings the largest gain for an equivalent increase in delay.

6.4 Geometrical Interpretation

We show why heuristics which consider only a single wire decrements (or increments) at a time are likely to reach a stationary point which is different from the optimum. Observe that the area (capacitance) objective is a separable function whose components are monotone. Consequently the optimal solution cannot be strictly interior and there must be some active constraints at the optimal point.

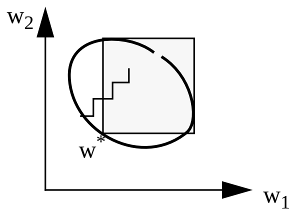


FIGURE 3. Illustration of the feasible domain (oval) and a set (shaded rectangle) that dominates the optimum w*.

A simplified 2D illustration of a feasible convex domain and a set of points that dominate the optimal solution is given in Fig.3. One point dominates another point if each of its element is larger or equal to the corresponding element of the other point. Consider an iterative process that starts from some feasible interior point and improves the value of the objective function only by decrements of a single wire at a time, as sketched in Fig. 3. As soon as a point that does not dominate the optimum is reached, the sequence cannot converge to the optimal solution because it would require an increase of a width.

One way to improve on the short sighted single wire down-hill improvement is to support both increments and decrements, or alternatively support uphill moves when required. While designing such iterative algorithms a care must be taken to avoid cyclic behavior and jamming. An engineering solution can be to consider at least a pair of wires such that (potentially) some are increased and some are decreased. It provides a down-hill improvement of the objective function together with a mechanism to escape from suboptimal stationary points. Unfortunately, counter examples can be contrived to show that for some pathological cases considering at most a pair is not sufficient. Those cases may occur only on the delay boundary when a concurrent resize of more then a pair is necessary to maintain the feasibility. In our experience good results are obtained by considering at most pairs.

7.0 Exact Wire-Sizing Algorithm (EWA)

EWA is implemented following the two phase methodology described in Section 3.0. The results from Phase I ensure a feasible starting point for Phase II. However, any feasible point could be used to initialize Phase II. For example, the delay minimization heuristics proposed in [3] and [19] could provide the starting point for the area minimization phase. Following the proofs and observations from Section 6.0 and the posynomial properties from Section 5.0, EWA converges to the exact solution by solving (4) in the transformed domain.

7.1 The Algorithm

The following is a high level description of EWA.

***Phase I**:*
[0] Initialize all the wires to their maximum width.
[1] Decrease the width of one (or several) wire(s), such as to maximize the reduction in the critical path delay.
[2] If there exists a wire whose size can be decreased while reducing the critical path delay, goto [1].
[3] End of Phase I.

***Phase II**:*
[0] Initialize the wire widths to the solution of Phase 1 (or any feasible point); Set the delay bounds using the results of Phase I to ensure a feasible starting point.
[1] Decrease the width of one (or several) wire(s) with negative sensitivity such that the product of sum of slacks with respect to the delay constraints (negative slacks are prohibited) and the area reduction is maximal.
[2] If there exists a wire with negative sensitivity goto [1].
[3] If there exists a subset (e.g. pair) of wires that can be resized to reduce the total area without violating the constraints perform a resize of that wires and goto [1].
[4] End of Phase II.

It is straightforward to realize that the algorithm terminates since the area is reduced at each iteration. Recall that the mathematical program is uni-modal and that it is equivalent to a convex program under the exponential (i.e. log) variable transformation, thus general purpose convex programming methods yield an optimal solution. Next we show how to customize a combination of the above with a formal method to suite the needs of a specific application.

7.2 Implementation Considerations

The way to pick a set of segments and to resize them at steps I(1), II(1), and II(3) of the algorithm is somewhat undetermined. Generally speaking, the larger the subset of wires which is considered the less efficient the iterations become, however, jamming is less likely. In practice we obtained very good results while considering at most a pair of wires.

The configuration $\overrightarrow{w_{II}}$ obtained from Phase II is feasible and the value of the objective function is an upper bound of the optimum. In order to bound the error we propose to linearizing the objective function and constraints around point $\overrightarrow{w_{II}}$. The resulting Linear Program is a true relaxation of the nonlinear program. The value of the original objective function at the optimal solution of the linear program (which is not necessarily a feasible solution to the nonlinear program) is a lower bound of the solution.

As an alternative approach to considering large subsets of segments, convergence can be achieved by using Sequential Linear Programming (SLP). Each iteration of Linear Programming can be done very efficiently. At this stage it is far easier then applying a formal method from the outset. Few major difficulties for applying formal methods in the general case are not present. To be more specific: the starting point is feasible and in the neighborhood of the optimal solution; by adjusting the units of the parameters a good scaling is easily obtained; the objective is monotone and separable and therefore lends itself to linearization; building a dense Hessian matrix is not required.

Practically, using the algorithm in section 7.1 in conjunction with SLP covers a broad range of applications that require different settings of implementation simplicity, computational efficiency, and accuracy of results.

7.3 Maximum Wire Width Constraints

Maximum wire width constraints can be represented as posynomial inequalities that do not disturb the convexity of the problem. A wire width may be increased during a pairwise (or multi-way) sizing step. If a max width constraint becomes active, the width is set to w^U_i. In most practical situations, the optimal solution is on the delay boundary even when some maximum width constraints are active.

For example the leaf wire segments are typically less than their maximum width for trees with significant depth, such as clock trees. If at the optimal solution the width of all the leaf wire segments is less than the upper bound and more than the lower bound, it implies that all the delay constraints are active, regardless of the active width constraints of upstream wires.

7.4 Minimum Wire Width Constraints

Like the maximum width constraints, the minimum width constraints are posynomial inequalities that do not disturb the convexity of the problem. When the *i-th* wire is decreased such that its minimum width constraint becomes active, its width is set to w^L_i. It may be increased when there are no wires with negative sensitivity and a step that involves at least a pair of wires is required; *e.g.* one is increased and the other decreased.

When one or more minimum width constraints are active at the optimal solution, some delay constraints may not be active, thus the solution may not be on the delay boundary. This observation has important implications on both signal nets (inequality constraints) and clock nets (equality constraints).

7.4.1 Signal Nets (Inequality Constraints)

At convergence of the algorithm to a configuration $\hat{\vec{w}}$, some minimum wire width constraints may become active, therefore $\hat{\vec{w}}$ does not have to be on the delay boundary $\overrightarrow{TD}$. Note that if we set $\overrightarrow{TD} = \overrightarrow{ED(\overline{w})}$ and re-run the algorithm it will converge to the same solution. Less formally it can be argued that within the set of active constraints there is a trade-off between delay constraints and width constraints.

7.4.2 Clock Nets (Equality Constraints)

Due to active minimum wire width constraints, not all the delay constraints have to be active, hence there may be a skew. Unlike for signal nets, we cannot simply tighten a delay constraint on some paths and accept the minimum area solution. In such cases all of the path-delay targets would have to be decreased, which could cause the clock tree to be over-designed (*i.e.* more resources than necessary used for meeting the design goals). It is reasonable to assume that active minimum-wire-width constraints indicate that there exists a better tree topology or a clustering of loads. Due to space limitations we cannot explore these issues in detail, but we use a simple clock tree example to illustrate them in the next section.

8.0 Experimental Results

Several examples are included in this section to demonstrate the utility of EWA.

8.1 Tapered Line

As a first example of applying EWA we consider a single tapered wire with capacitance load and driver resistance, as shown in Fig. 4. The parameters for the example were taken from a paper by Fishburn and Schevon [8], where a taper function that minimizes the Elmore delay of a single wire is derived. A similar derivation was made independently in [1]. Unlike the analytical derivation, EWA demonstrates the tapering for the minimum area solution while considering fringe capacitance, maximum width constraints, and minimum width constraints.

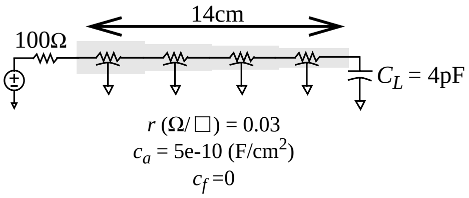


FIGURE 4. Fishburn's tapered line example.

Fishburn's model achieves a delay of 3.72ns with exponential tapering of the width from 30.7 to 7.8 microns, as shown in Fig. 5(a). The EWA result for Phase I is also plotted for comparison, where the wire was broken into 20 segments and the total wiring area was 2.32mm^2. Next we applied Phase II of the EWA approach targeting a delay of 4.25ns that is 15% above the minimum possible delay. The total metal area was reduced by 50% to 1.17mm^2. The widths are shown plotted in Fig. 5(a).

Of further interest is to consider what the optimal tapering looks like when fringe capacitance or wire width constraints are imposed. The tapering is no longer a perfect exponential, but it did not change dramatically for this example. The results are summarized in Fig. 5(b).

8.2 Small Signal Net with Non-monotone Wire Width Assignment

As mentioned in Section 1.0, EWA does not rely on any restrictions on the wire width assignment, however some other methods [3] [19] rely on a monotone assignment of wire widths that increase from the leaf nodes toward the root node. The monotonicity makes sense and can be proved for a single layer wiring; it does not hold for deep submicron technologies and multi-layer wiring, where the unit resistivity and unit-capacitance vary significantly between layers.

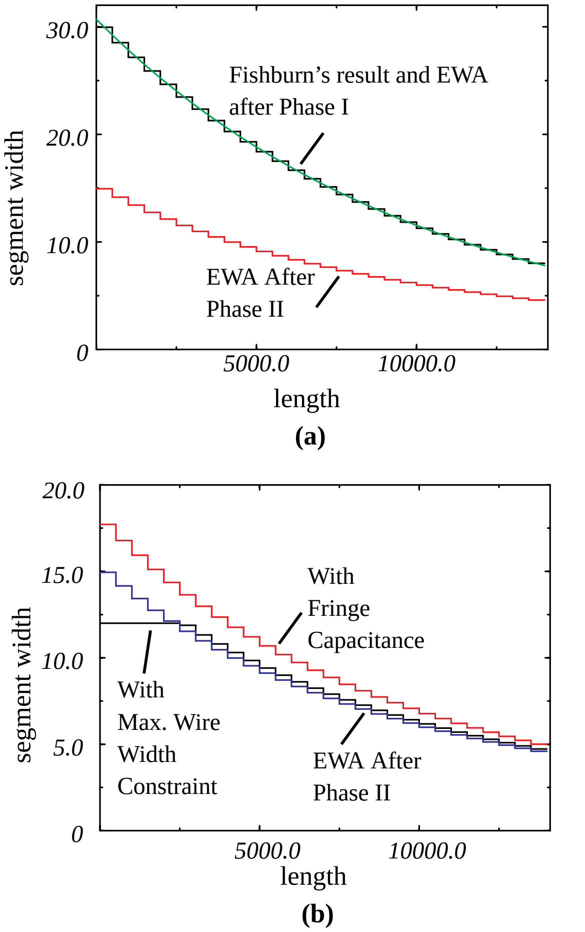


FIGURE 5. Tapered line example from Fig. 4.
(a) Comparison with Fishburn's analytical solution after phase I, and minimum area solution after phase II.
(b) Change in exponential tapering function due to fringe capacitance (c_f=0.05(fF/micron)) and maximum wire width constraints (12 microns).

To demonstrate this point, consider the simple example circuit in Fig. 6 for a typical 0.5 micron CMOS process. All of the horizontal segments are routed on metal 1 (M1) and all of the vertical segments are routed on metal 4 (M4). Phase I of EWA produced a delay of 273ps for the critical path to the 0.5pF load. For Phase II we targeted the delay at this node to be 314ps which is 15% larger. Notice that the wire widths assignment in Fig. 6 is non-monotone,

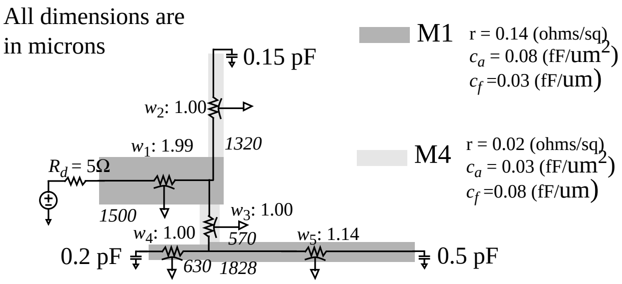


FIGURE 6. Single-stage driver and wire sizing example. Non-monotone wire width assignment for a target delay of 314ps and max and min wire width constraints of 2 and 1 micron respectively. The total metal area is 7,588 um^2

since segment 5 is wider than the upstream segment 3.

8.3 Clock Tree

We have tried EWA on some of the Tsay benchmark circuits [20] with the minimum wire width constraints set to zero. As could be expected, many of the downstream wires became less then the minimum feature size of today's technologies. It indicates that we must either reduce the target delays so that the wires are required to be wider, or modify the tree topology. Both are important considerations that can be most easily discussed in terms of a simpler clock tree example.

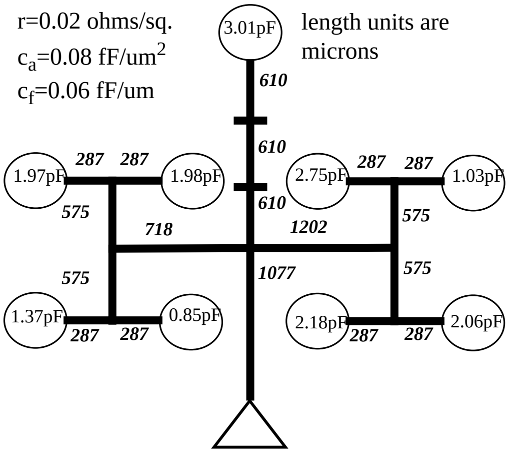
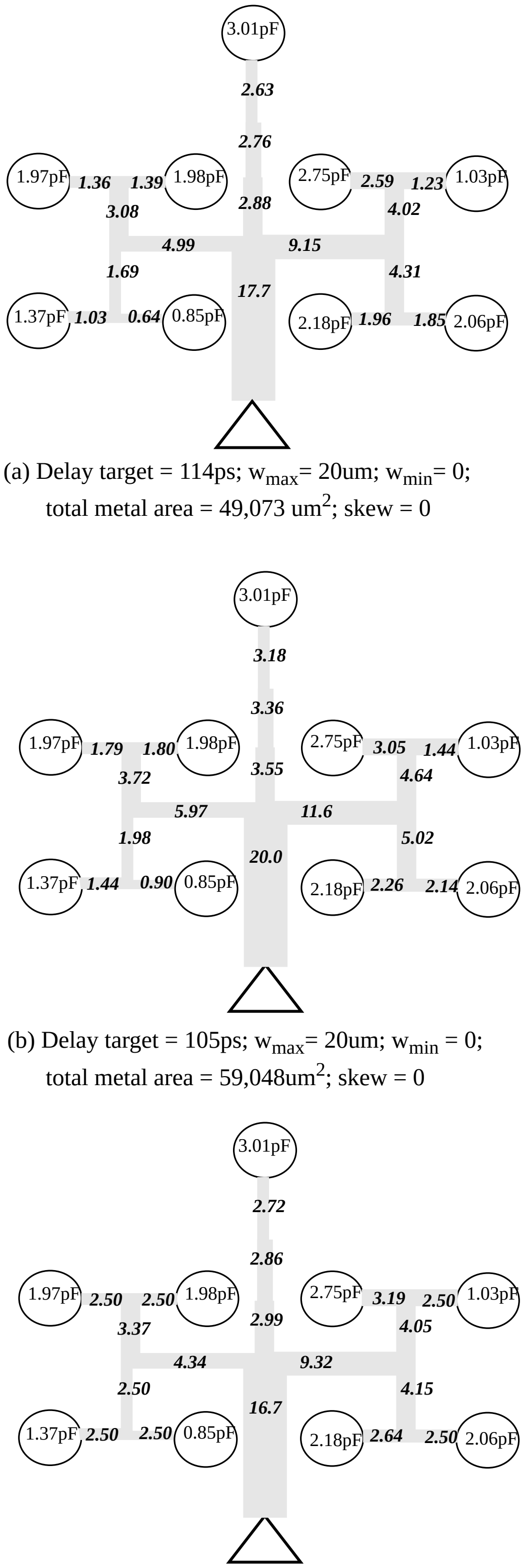


FIGURE 7. Clock tree distribution - driver resistance is 2 ohms.

Fig. 7 is a sketch of the top-level clock distribution for a microprocessor. The load capacitances represent clusters of local repeaters and the associated interconnect. The design objective is to size the wires, such as to balance the delays (zero the skew) to all of the clusters. Using the two phase approach, the delay for Phase II was first targeted at 114ps. The maximum allowable width was 20 microns for all the wires. The sized clock tree is shown in Fig. 8(a). Note that the skew is zero, but one of the wire widths is 0.64 microns which is less than the smallest manufacturable width of 0.9 microns. There are two options to proceed, either modify the tree or target the delay more aggressively. For this design we considered retargeting the delay to 105ps, which tends to increase the smallest wire width, as shown in Fig. 8(b). The narrowest wire became manufacturable at a width of 0.9 microns, while the total metal area was increased by 20%.

If the clock is distributed on a metal layer with minimum feature sizes greater than 0.9 microns, we should retarget the delays even more aggressively, as shown in Fig. 8(c) for a minimum allowable width of 2.5 microns. The area is decreased as we trade area for less aggressive delay and more skew. The skew became 5.7ps due to the delay at the 0.85pF load which is less then the target. Such small skew is probably acceptable. It could be brought to zero by changing the tree topology or by adding dummy loads.

Observe what the minimum wire width violations indicate for clock trees. While the tree in Fig. 7 was distributed in somewhat of an H-tree fashion, the imbalance of the cluster loads made a balanced H-tree non-optimal. In other words, there exists a better tree topology and/or a clustering of loads as a starting point for wire sizing. While it is beyond the scope of this paper to discuss such design trade-offs, we shall explore this problem as part of the future work associated with EWA.



(a) Delay target = 114ps; w_{max}= 20um; w_{min}= 0;
total metal area = 49,073 um^2; skew = 0

(b) Delay target = 105ps; w_{max}= 20um; w_{min} = 0;
total metal area = 59,048um^2; skew = 0

(c) Delay target = 114ps; w_{max}=20um; w_{min} =2.5um;
total metal area = 51,600 um^2; skew = 5.7ps

FIGURE 8. Clock tree example that demonstrates the design trade-offs offered by changing the delay targets and wire width constraints. (Width units are microns).

9.0 Conclusion

The algorithm presented in this paper converges to the global optimum of the wire-sizing problem under the Elmore delay model. The algorithm is simple and efficient making it useful for applications where one might be tempted to use heuristics. Several new observations were made to facilitate the algorithm which are of both theoretical and practical value. Most of the observations hold for more exact delay models, but we can not at this time prove that the delay bounds form a convex set. It can be shown empirically that using higher order delay models yield better results. Future work will be focused on extending EWA to include higher order delay models and developing efficient schemes for selecting the best set of wires to be sized in each step of the algorithm.

10.0 References

[1] C-P. Chen, Y-P. Chen and D.F. Wong, "Optimal Wire-Sizing Formula Under the Elmore Delay Model," In Proc. *Design Automation Conference*, pp. 487-490, June 1996.

[2] J. Cong, L. He, C-K Koh, and P.H. Madden, "Performance Optimization of VLSI Interconnect Layout," to appear in *Integration, the VLSI Journal.*

[3] J. Cong and K.-S. Leung, "Optimal wiresizing under the distributed Elmore delay model," *Proc. of the Intl. Conf. on Computer-Aided Design*, pp. 634–639, Nov. 1993.

[4] J. Cong and K.-S. Leung, "Optimal wiresizing under Elmore delay model," *IEEE Trans. Computer-Aided Design*, **14**(3), pp. 321-336, March 1995.

[5] J. G. Ecker, "Geometric programming: methods, computations and applications," *SIAM Review*, **22**, pp. 338-362, July 1980.

[6] W. C. Elmore, "The transient response of damped linear networks with particular regard to wide-band amplifiers," *J. Applied Physics*, **19**(1), pp. 55-63, Jan. 1948.

[7] J. P. Fishburn and A. E. Dunlop, "TILOS: A posynomial programming approach to transistor sizing," *Proc. of the Intl. Conf. on Computer-Aided Design*, pp. 326–328, Nov. 1985.

[8] J. P. Fishburn and C.A. Schevon, "Shaping a Distributed-RC Line to Minimize Elmore Delay," *IEEE Trans. on Circuits and Systems*, **42**(12), pp. 1020-1022, December 1995.

[9] J. Y. Lee, X Huang, and R. Rohrer, "Pole and zero sensitivity calculation in asymptotic waveform evaluation," *IEEE Trans. Computer-Aided Design*, **11**(5), pp. 586-597, May 1992.

[10] N. Menezes, S. Pullela, F. Dartu, and L. T. Pillage, "RC interconnect synthesis – a moment fitting approach," *Proc. of the Intl. Conf. on Computer-Aided Design*, pp. 418–425, Nov. 1994.

[11] N. Menezes, R. Baldick and L. Pileggi, A Sequential Quadratic Programming Approach to Concurrent Gate and Wire Sizing, *Proceedings of the International Conference on Computer-Aided Design, pp. 144-151*, 1995.

[12] P. Penfield, J. Rubinstein, and M. Horowitz, "Signal delay in RC tree networks," *IEEE Trans. Computer-Aided Design*, **CAD-2**, pp. 202-211, July 1983.

[13] L. T. Pillage and R. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design*, **9**(4), pp. 352-366, April 1990.

[14] S. Pullela, N. Menezes, and L. T. Pillage, "Reliable non-Zero Skew Clock Trees Using Wire Width Optimization," *Proc. 30th ACM/IEEE Design Automation Conference, pp.165-170,* June 1993.

[15] S. Pullela, N. Menezes and L.T. Pileggi, Clock Skew Minimization via Wire Width Optimization, *IEEE Transactions on Computer-Aided Design,* Accepted for Publication.

[16] C. Ratzlaff and L. T. Pillage, "RICE: rapid interconnect circuit evaluation using AWE," *IEEE Trans. Computer-Aided Design*, **13**(6), pp. 763-776, June 1994.

[17] S. S. Sapatnekar, V. B. Rao, P. M. Vaidya, and S.-M. Kang, "An exact solution to the transistor sizing problem for CMOS circuits using convex optimization," *IEEE Trans. Computer-Aided Design*, **12**(11), pp. 1621- 1634, May 1992.

[18] S. S. Sapatnekar, "RC interconnect optimization under the Elmore delay model," *Proc. 31st ACM/IEEE Design Automation Conferenc*e, pp. 387–391, June 1994.

[19] S. S. Sapatnekar, "Wire Sizing as a Convex Optimization Problem: Exploring the Area-Delay Tradeoff," *IEEE Trans. Computer-Aided Design*, **15**(8), pp. 1001- 1011, Aug. 1996.

[20] R-S. Tsay, "Exact Zero-skew", pp. 336-339, *Proc. IEEE International Conference on Computer-Aided Design*, Nov 1991.

[21] Quing Zhu, Wayne W.-M. Dai, and Joe G. Xi, "Optimal Sizing of High-Speed Clock Networks Based on Distributed RC and Lossy Transmission Line Models," pp. 628-633, *Proc. International Conference on Computer Aided Design 1993*.