Computer Science Department Technical Report
University of California
Los Angeles, CA 90024-1596

OPTIMAL WIRESIZING UNDER THE DISTRIBUTED ELMORE DELAY MODEL

J. Cong
K. S. Leung

April 1993
CSD-930012

Optimal Wiresizing Under the Distributed Elmore Delay Model

Jason Cong and Kwok-Shing Leung*
Department of Computer Science
University of California, Los Angeles
Los Angeles, CA 90024

Abstract

In this paper, we study the optimal wiresizing problem under the distributed Elmore delay model. We show that the optimal wiresizing solutions satisfy a number of interesting properties, including the separability, the monotone property, and the dominance property. Based on these properties, we have developed a polynomial-time optimal wiresizing algorithm for arbitrary interconnect structures under the distributed Elmore delay model. Extensive experimental results have shown that our wiresizing solution reduces interconnect delay by up to 51% when compared to the uniform-width solution of the same routing topology. Furthermore, compared to the wiresizing solution based on a simpler RC delay model in [7], our wiresizing solution reduces the total wiring area by up to 28% while further reducing the interconnect delays to the timing-critical sinks by up to 12%.

*The current address of Kwok-Shing Leung is: Carnegie Mellon University, Pittsburgh, PA 15213.

1 Introduction

As the VLSI fabrication technology reaches submicron device dimension and gigahertz frequency, interconnect delay has become the dominant factor in determining circuit speed [11, 23]. The analysis in [25] and [7] showed that in the conventional VLSI technology, interconnect delay is determined by the product of the driver resistance and the total wire capacitance. As a result, the minimum interconnect delay is achieved when the routing tree is an optimal Steiner tree with the minimum wire width for each segment (since it has the minimum total wire capacitance). Therefore, conventional global and detailed routers aimed at generating minimum-width Steiner routing trees using the least total wirelength [9, 18, 4]. However, as we reduce the device dimension, the driver resistance becomes smaller and the wire resistance becomes larger, which results in a much smaller resistance ratio (defined to be the ratio of the driver resistance versus the unit wire resistance). In this case, the *distributed* nature of the interconnect structure must be considered, and minimizing the total wire capacitance does not necessarily lead to the minimum interconnect delay. The impact of decreasing resistance ratio in modern VLSI designs was discussed in details in [7].

Most existing work on performance-driven VLSI routing concentrates on optimizing the interconnect topology of the routing trees for reducing interconnect delay. In [12], net priorities are determined based on static timing analysis; nets with high priorities are processed earlier using fewer feedthroughs. In [17], a hierarchical approach to timing-driven routing was outlined. In [20], a timing-driven global router based on the A* heuristic search algorithm was proposed in building-block designs. In [6], a timing-driven global router was proposed to minimize both the cost (i.e. the total wirelength) and the radius (i.e. the longest path from the source to any sink) simultaneously. Other cost-radius tradeoff methods were reported in [1, 3, 19]. Both the maximum performance tree formulation in [5] and the A-tree formulation in [7] aimed at constructing a minimum wirelength routing tree which has the shortest path connection between the source and every sink. Experimental results showed that the algorithm in [7] can construct A-trees which are at most 4% within the optimal, and achieve interconnect delay reduction by as much as 66% when compared to the best-known Steiner routing topology. When the critical-path information is available during iterative timing-driven layout, the critical sink routing approaches in [2, 15] reduce the delays to specified sinks substantially.

Although steady progress has been made in optimizing interconnect topology design for delay minimization, there was very few work on wiresizing optimization for high-performance interconnect designs. In the past, wiresizing was used only for clock nets as in the H-tree clock routing by Fisher and Kung [14], and in the more recent works by Pullela, Menezes and Pillage [21] and by Zhu, Dai, and Xi [27]. Recently, Cong, Leung, and Zhou first used the wiresizing technique for interconnect delay reduction of general signal nets [7]. They developed an optimal wiresizing algorithm minimizing the delay in the upper bound delay model in a distributed RC tree proposed by Rubinstein, Penfield and Horowitz [22]. In their delay model, the signal delay at any node

in a distributed RC circuit is estimated by

$$t = \sum_{all\ nodes\ k} R_k \cdot c_k \tag{1}$$

where R_k is the resistance between the source and the node k and c_k is capacitance at the node k. This upper-bound delay model was chosen in [7] because it simplifies the wiresizing optimization. However, the simplicity of this delay model also results in several drawbacks. First, it provides only an upper bound of the worst-case RC delay in the routing tree and does not distinguish the delays at different sinks. Therefore, it is impossible to optimize the wiresizing solution to reduce the delays to the specific timing-critical sinks. Moreover, since this model tends to over-estimate the delays at many sinks in the routing tree, it often results in unnecessary over-sizing of many wire segments. Oversized wires not only occupy more routing spaces, but also increase the mutual capacitance and inductance between different signal nets. Thus, there is a strong need to develop optimal wiresizing algorithms under more accurate interconnect delay models.

In this paper, we study the optimal wiresizing problem under the distributed Elmore delay model [13, 22]. We have shown that the optimal wiresizing solutions satisfy a number of interesting properties, including the separability, the monotone property, and the dominance property. Based on these properties, we have developed a polynomial-time optimal wiresizing algorithm for arbitrary interconnect structures under the distributed Elmore delay model. Extensive experimental results have shown that our wiresizing solution reduces interconnect delay by up to 51% when compared to the uniform-width solution of the same routing topology. Furthermore, compared to the wiresizing solution in [7], our wiresizing solution reduces the total wiring area by up to 28% while further reducing the interconnect delays to the timing-critical sinks by up to 12%.

The remainder of this paper is organized as follows: In Section 2, we present the general formulation of the wiresizing problems under the distributed Elmore delay model. In Section 3, we study the properties of the optimal wiresizing solutions. In Section 4, we present a polynomial-time optimal wiresizing algorithm. Section 5 shows the experimental results obtained by our wiresizing algorithm. Section 6 extends our basic wiresizing formulation to a more general case, and Section 7 concludes the paper with discussions of future work. An extended abstract of this paper was presented in ICCAD '93 [8].

2 Problem Formulation

Assume that we are given a routing tree T implementing a signal net which consists of a source N_+, and a set of m sinks $\{N_1, N_2, \cdots, N_m\}$. A node refers to the source, or a sink, or a Steiner node in T, and a segment connects two nodes in T. Assume that $\{E_1, E_2, \cdots, E_n\}$ is the set of segments forming the tree T, where n is the total number of segments in the tree. Notice that n is one less than the total number of nodes in the tree.

In order to model a routing tree as a distributed RC circuit accurately, a grid structure is superimposed on the routing plane, and each wire segment in the routing plane is divided into a sequence of wires of unit length as shown in Figure 1. (Adjacent grid points are unit length apart.) For each grid edge ended at u in the tree

T, we use a π-type RC circuit to model the interconnect, where r_u and c_u are the unit interconnect resistance and capacitance, respectively. We use c_u^s to denote the node capacitance at u. If u is a sink, c_u^s represents the loading capacitance at the sink. If there is a via or bend at node u, it can also be formulated by introducing a small capacitance at the node u. For simplicity, we assume that c_u^s is non-zero if u is a sink, and zero otherwise. To correctly model the driver resistance, we introduce an additional node N_0 and connect N_0 to N_+ via an additional segment with resistance R_d (the driver resistance). Since each grid point u is uniquely identified with an incoming grid edge in the routing tree, we also use u to refer to that grid edge in the later discussions. Given a grid point u, we use $Des(u)$ to denote the set of grid points in the subtree rooted at u (excluding u), and $Ans(u)$ to denote the set of grid points $\{v|u \in Des(v)\}$ (again, excluding u). That is, $Des(u)$ is the set of "descendant" grid points of u, and $Ans(u)$ is the set of "ancestor" grid points of u. Also, we use $sink(u)$ to denote the set of sinks in the subtree rooted at u, and C_u to denote the *total* capacitance in the subtree rooted at u (including both the wire capacitances and the sink capacitances). Furthermore, we use $P(u, v)$ to denote the unique path from u to v for any grid point u, v in the routing tree.

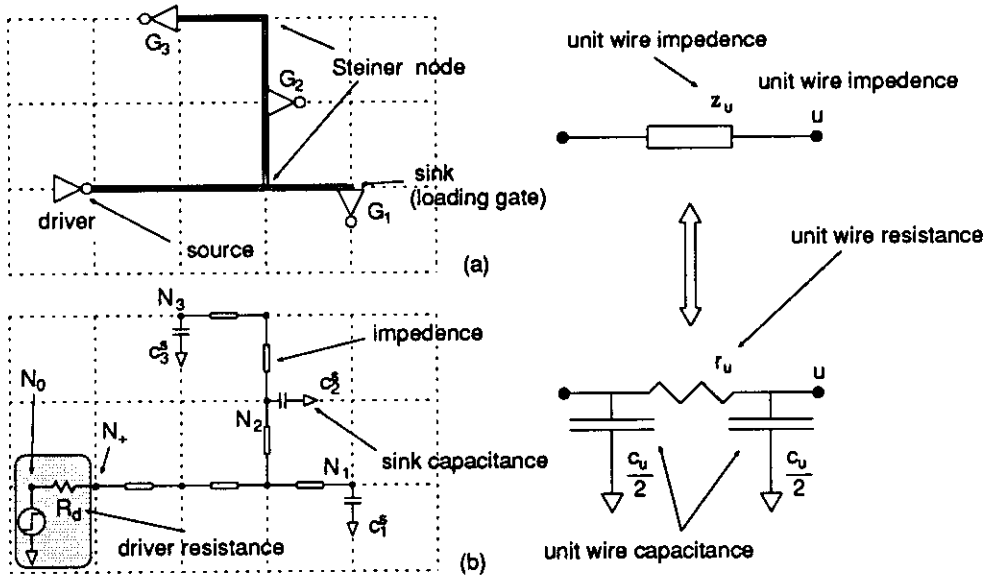

Figure 1: A grid structure for the distributed RC interconnect model. (a) The layout of an interconnect tree T with 3 sinks N_1, N_2, and N_3. (b) The corresponding distributed RC interconnect model of T. Each grid edge in T connecting two adjacent nodes is modeled as a π-type RC circuit containing a resistor of r_u and two capacitors of $\frac{c_u}{2}$ each, where u is the farther end of the grid edge from the source. Each sink has an extra loading capacitance.

We assume that each wire segment has a set of discrete choices of wire widths $\{W_1, W_2, \cdots, W_r\}$ ($W_1 < W_2 < \cdots < W_r$), and the wire width within the same segment does not change. This segment-based wiresizing model resembles more closely to the realistic design style and reflects the actual technological constraint where arbitrary width variation *within a segment* is usually undesirable. Nevertheless, this segment-based formulation can be generalized to handle the case where variable wire width is allowed within a segment, simply by

introducing artificial degree-2 Steiner nodes in the segment. Given a node u, we use w_u to denote the width of the grid edge u, and w_E and l_E to denote the width and length of the segment E, respectively. Assume that a unit-width unit-grid-length wire has wire resistance r_0 and wire capacitance c_0, then $r_u = \frac{r_0}{w_u}$ and $c_u = c_0 \cdot w_u$ for any grid edge u.

In this paper, we use the Elmore delay model [13] as the objective function for delay optimization. Given a distributed RC circuit tree T, the signal delay at a particular node N_i, denoted as $t(N_i)$, is computed as follows:

$$t(N_i) = \sum_{u \in P(N_0, N_i)} r_u \cdot (\frac{c_u}{2} + C_u) \tag{2}$$

where the summation is taken over all the grid points on the path from the driver N_0 to the node N_i. An elegant hierarchical algorithm was presented in [24] for computing the Elmore delay in any RC tree in linear time.

2.1 Delay Minimization Formulation with Single Critical Sink

We shall first study the case where there is only one critical sink N_i in the net. According to (2), the signal delay $t(N_i)$ at N_i under a given wiresizing solution $\mathcal{W}$ is:

$$\begin{aligned}
t_i(\mathcal{W}) &= \sum_{u \in P(N_0, N_i)} r_u \cdot (\frac{c_u}{2} + C_u) \\
&= \sum_{u \in P(N_+, N_i)} \frac{r_0}{w_u} \cdot \frac{c_0 \cdot w_u}{2} + \sum_{u \in P(N_0, N_i)} r_u \cdot C_u \\
&= \sum_{u \in P(N_+, N_i)} \frac{r_0 \cdot c_0}{2} + R_d \cdot C_{N_0} + \sum_{u \in P(N_+, N_i)} r_u \cdot C_u \\
&= \sum_{u \in P(N_+, N_i)} \frac{r_0 \cdot c_0}{2} + \left\{ R_d \cdot \sum_{u \in T} c_u^s + R_d \cdot c_0 \cdot \sum_{u \in T} w_u \right\} + \\
&\quad \left\{ \sum_{u \in P(N_+, N_i)} r_u \cdot \sum_{v \in Des(u)} c_v + \sum_{u \in P(N_+, N_i)} r_u \cdot \sum_{v \in sink(u)} c_v^s \right\} \\
&= \sum_{u \in P(N_+, N_i)} \frac{r_0 \cdot c_0}{2} + R_d \cdot \sum_{u \in T} c_u^s + R_d \cdot c_0 \cdot \sum_{u \in T} w_u + \\
&\quad r_0 \cdot c_0 \cdot \sum_{u \in P(N_+, N_i)} \sum_{v \in Des(u)} \frac{w_v}{w_u} + r_0 \cdot \sum_{u \in P(N_+, N_i)} \sum_{v \in sink(u)} c_v^s \cdot \frac{1}{w_u}
\end{aligned}$$

Let

$$\begin{aligned}
\mathcal{K}_1 &= \sum_{u \in P(N_+, N_i)} \frac{r_0 \cdot c_0}{2} \\
\mathcal{K}_2 &= R_d \cdot \sum_{u \in T} c_u^s \\
\mathcal{K}_3 &= R_d \cdot c_0 \\
\mathcal{K}_4 &= r_0 \cdot c_0 \\
\mathcal{K}_5 &= r_0
\end{aligned} \tag{3}$$

Note that $\mathcal{K}_1$, $\mathcal{K}_2$, $\mathcal{K}_3$, $\mathcal{K}_4$, and $\mathcal{K}_5$ are all constants. Furthermore, let

$$f_i(u,v) = \begin{cases} 1 & \text{if } u \in P(N_+, N_i) \text{ and } v \in Des(u) \\ 0 & \text{otherwise} \end{cases}$$

$$g_i(u,v) = \begin{cases} c_v^s & \text{if } u \in P(N_+, N_i) \text{ and } v \in sink(u) \\ 0 & \text{otherwise} \end{cases}$$

That is, $f_i(u,v) = 1$ if and only if u is on the path from the driver to the critical sink N_i and v is in the subtree rooted at u. Similarly, $g_i(u,v) = c_v^s$ if and only if u is on the path from the driver to the critical sink N_i and v is a sink in the subtree rooted at u. Then, we have:

$$t_i(\mathcal{W}) = \mathcal{K}_1 + \mathcal{K}_2 + \mathcal{K}_3 \cdot \sum_{u \in T} w_u + \mathcal{K}_4 \cdot \sum_{u,v \in T} f_i(u,v) \cdot \frac{w_v}{w_u} + \mathcal{K}_5 \cdot \sum_{u,v \in T} g_i(u,v) \cdot \frac{1}{w_u}. \quad (4)$$

where u and v refer to the grid points in T.

Now, we shall transform the grid-based formulation into the segment-based formulation. We first extend the definition of ancestors and descendants to the domain of segments. That is, $Des(E)$ is the set of segments in the subtree "rooted" at E (excluding E), and $Ans(E)$ is the set $\{E' | E \in Des(E')\}$ (again, excluding E). Given two distinct wire segments E and E', for any grid edges $u, u' \in E$, and $v, v' \in E'$, we have $f_i(u,v) = f_i(u',v')$. Therefore, we use $f_i(E,E')$ to denote $f_i(u,v)$ when $u \in E$ and $v \in E'$ (provided that $E \neq E'$). Similarly, since $\sum_{v \in T} g_i(u,v) = \sum_{v \in T} g_i(u',v)$ for any pairs of grid edges u and u' in the same segment E, we use $g_i(E)$ to refer to $\sum_{v \in T} g_i(u,v)$ when $u \in E$. Therefore, we can rewrite the last three terms in (4) as follows:

$$\mathcal{K}_3 \cdot \sum_{u \in T} w_u = \mathcal{K}_3 \cdot \sum_{E \in T} l_E \cdot w_E$$

$$\begin{aligned} \mathcal{K}_4 \cdot \sum_{u,v \in T} f_i(u,v) \cdot \frac{w_u}{w_v} &= \mathcal{K}_4 \cdot \sum_{E \in T} \sum_{u,v \in E} f_i(u,v) \cdot \frac{w_E}{w_E} + \mathcal{K}_4 \cdot \sum_{E,E' \in T, E \neq E'} \left\{ \sum_{u \in E, v \in E'} f_i(u,v) \cdot \frac{w_{E'}}{w_E} \right\} \\ &= \mathcal{K}_4 \cdot \sum_{E \in T} \sum_{u,v \in E} f_i(u,v) + \mathcal{K}_4 \cdot \sum_{E,E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot f_i(E,E') \cdot \frac{w_{E'}}{w_E} \end{aligned}$$

$$\begin{aligned} \mathcal{K}_5 \cdot \sum_{u,v \in T} g_i(u,v) \cdot \frac{1}{w_u} &= \mathcal{K}_5 \cdot \sum_{u \in T} \left\{ \sum_{v \in T} g_i(u,v) \cdot \frac{1}{w_u} \right\} \\ &= \mathcal{K}_5 \cdot \sum_{E \in T} l_E \cdot g_i(E) \cdot \frac{1}{w_E} \end{aligned}$$

where $\mathcal{K}_3$, $\mathcal{K}_4$, and $\mathcal{K}_5$ are constants depending only on the technology and interconnect topology as shown in (3). Note that $\mathcal{K}_4 \cdot \sum_{E \in T} \sum_{u,v \in E} f_i(u,v)$ is a constant for a given routing tree because f_i's are independent of the wiresizing solution. Therefore, our objective is find a wiresizing assignment $\mathcal{W}$ which minimizes:

$$T_i(\mathcal{W}) = \mathcal{K}_3 \cdot \sum_{E \in T} l_E \cdot w_E + \mathcal{K}_4 \cdot \sum_{E,E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot f_i(E,E') \cdot \frac{w_{E'}}{w_E} + \mathcal{K}_5 \cdot \sum_{E \in T} l_E \cdot g_i(E) \cdot \frac{1}{w_E} \quad (5)$$

Note that the first term minimizes the total wiring area of T. Moreover, a careful study of f_i's and g_i's reveals that

$$\begin{aligned} f_i(E_1, E_2) &\geq f_i(E_1, E_2') && \text{if } E_2 \in Des(E_2') \\ f_i(E_1, E_2) &\geq f_i(E_1', E_2) && \text{if } E_1 \in Ans(E_1') \\ g_i(E_1) &\geq g_i(E_1') && \text{if } E_1 \in Ans(E_1') \end{aligned}$$

Note that given a routing tree and a specified critical sink N_i, all $f_i(E_1, E_2)$'s can be precomputed and stored in a two-dimensional $n \times n$ matrix before wiresizing optimization. Similarly, all $g_i(E)$'s can be precomputed and stored in a linear array.

2.2 Delay Minimization Formulation with Multiple Critical Sinks

Let $sink(T)$ denote the set of sinks in T. When there are several critical sinks of different priorities in the routing tree, the previous formulation can be generalized to optimize:

$$T(\mathcal{W}) = \sum_{N_i \in sink(T)} \lambda_i \cdot T_i(\mathcal{W}) \tag{6}$$

where λ_i is the weight of the delay penalty to sink N_i. The larger λ_i is, the more critical sink N_i is. We normalize λ_i's such that $\sum_{N_i \in sink(T)} \lambda_i = 1$. We can rewrite (6) as follows:

$$\begin{aligned}
& \sum_{N_i \in sink(T)} \lambda_i \cdot T_i(\mathcal{W}) \\
= & \sum_{N_i \in sink(T)} \lambda_i \cdot \mathcal{K}_3 \cdot \sum_{E \in T} l_E \cdot w_E \; + \\
& \sum_{N_i \in sink(T)} \lambda_i \cdot \mathcal{K}_4 \cdot \sum_{E, E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot f_i(E, E') \cdot \frac{w_{E'}}{w_E} \; + \; \sum_{N_i \in sink(T)} \lambda_i \cdot \mathcal{K}_5 \cdot \sum_{E \in T} l_E \cdot g_i(E) \cdot \frac{1}{w_E} \\
= & \; \mathcal{K}_3 \cdot \sum_{E \in T} l_E \cdot w_E \; + \\
& \mathcal{K}_4 \cdot \sum_{E, E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot \frac{w_{E'}}{w_E} \cdot \left\{ \sum_{N_i \in sink(T)} \lambda_i \cdot f_i(E, E') \right\} \; + \\
& \mathcal{K}_5 \cdot \sum_{E \in T} l_E \cdot \frac{1}{w_E} \cdot \left\{ \sum_{N_i \in sink(T)} \lambda_i \cdot g_i(E) \right\} \quad .
\end{aligned}$$

Let $F(E, E') = \sum_{N_i \in sink(T)} \lambda_i \cdot f_i(E, E')$, and $G(E) = \sum_{N_i \in sink(T)} \lambda_i \cdot g_i(E)$. Then, our new objective is to find a wiresizing solution $\mathcal{W}$ which minimizes:

$$T(\mathcal{W}) \;=\; \mathcal{K}_3 \cdot \sum_{E \in T} l_E \cdot w_E \; + \; \mathcal{K}_4 \cdot \sum_{E, E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot F(E, E') \cdot \frac{w_{E'}}{w_E} \; + \; \mathcal{K}_5 \cdot \sum_{E \in T} l_E \cdot G(E) \cdot \frac{1}{w_E} \tag{7}$$

Again, $\mathcal{K}_3$, $\mathcal{K}_4$, and $\mathcal{K}_5$ are constants given in (3). Note that the objective function for multiple critical sinks is very similar to that for a single critical sink, except that f and g in (5) are replaced by F and G in (7), respectively. In fact, we can show that F and G behave in the same way as f and g, since F and G are convex combinations of f and g:

$$\begin{aligned}
F(E_1, E_2) \;&\geq\; F(E_1, E_2') \qquad \text{if } E_2 \in Des(E_2') \\
F(E_1, E_2) \;&\geq\; F(E_1', E_2) \qquad \text{if } E_1 \in Ans(E_1') \\
G(E_1) \;&\geq\; G(E_1') \qquad \text{if } E_1 \in Ans(E_1')
\end{aligned} \tag{8}$$

Moreover, given a routing tree and the weights of the delay penalty to all the sinks, we can precompute the two-variable function F and the one-variable function G by taking weighted sum of f_i's and g_i's. Since computing

each f_i and g_i takes $O(n^2)$ time (n is the number of wire segments in the routing tree), the computation of F and G takes $O(n^3)$ time, and it can be carried out before the wiresizing optimization. Except that the complexity of computing F and G is slightly higher than that of computing f and g, for all other practical purposes, wiresizing optimization algorithm for multiple critical sinks behaves exactly the same way as that for a single critical sink. In the remaining sections, we shall use the multiple-critical-sink formulation in our wiresizing algorithms.

2.3 Combined Delay and Area Minimization

In general, wiresizing will reduce the signal delay but increase the wiring area. In many cases, we want to minimize both the interconnect delay and the routing area, especially when the global routing solution is very dense and mutual capacitance and inductance cannot be ignored. Therefore, we want to explore the *tradeoff* between area and delay. This tradeoff can be formulated as finding a wiresizing assignment $\mathcal{W}$ which minimizes

$$\alpha \cdot \textbf{Area} + \beta \cdot \textbf{Delay} \tag{9}$$

where α and β are constants.

This formulation is, however, captured by the general formulation in (7) since

$$\begin{aligned}
& \alpha \cdot \textbf{Area} + \beta \cdot \textbf{Delay} \\
= \; & \alpha \cdot \sum_{E \in T} l_E \cdot w_E + \beta \cdot T(\mathcal{W}) \\
= \; & (\alpha + \beta \mathcal{K}_3) \cdot \sum_{E \in T} l_E \cdot w_E + \beta \mathcal{K}_4 \cdot \sum_{E, E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot F(E, E') \cdot \frac{w_{E'}}{w_E} + \beta \mathcal{K}_5 \cdot \sum_{E \in T} l_E \cdot G(E) \cdot \frac{1}{w_E} \\
= \; & \mathcal{K}'_3 \cdot \sum_{E \in T} l_E \cdot w_E + \mathcal{K}'_4 \cdot \sum_{E, E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot F(E, E') \cdot \frac{w_{E'}}{w_E} + \mathcal{K}'_5 \cdot \sum_{E \in T} l_E \cdot G(E) \cdot \frac{1}{w_E}
\end{aligned}$$

where $\mathcal{K}'_3 = \alpha + \beta \mathcal{K}_3$, $\mathcal{K}'_4 = \beta \mathcal{K}_4$, and $\mathcal{K}'_5 = \beta \mathcal{K}_5$ remain as constants. Therefore, we shall concentrate on finding a wiresizing solution which minimizes the objective function in (7).

3 Properties of Optimal Wiresizing Solutions

In this section, we study several interesting properties of optimal wiresizing solutions, including the *separability*, the *monotone property*, and the *dominance property*. These properties are very useful in the development of the wiresizing algorithms in the next section. The proofs of these properties, in a more general context, will be given in the Appendix.

3.1 Separability

Theorem 1 *If the width assignment of a path P originated from the source is given, the optimal width assignment for each subtree branching off P can be carried out independently.* □

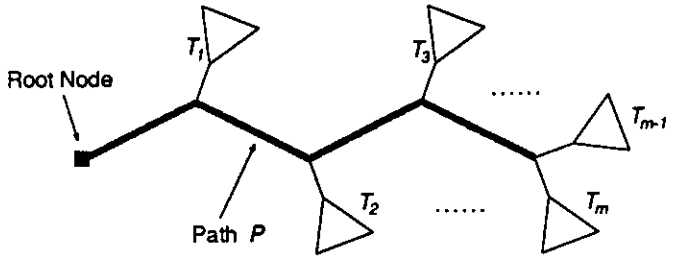

Figure 2: A routing tree can be decomposed into a main path P from the source, and a set $B = \{T_1, T_2, \cdots, T_m\}$ of subtrees branching off the main path.

Figure 2 illustrates the separability, where the optimal width assignemnt for $T_1, T_2, \cdots, T_m$ can be computed independently if the width assignment of path P is given. We shall show later that the separability indeed plays a significant role in the development of our polynomial-time optimal wiresizing algorithm.

3.2 Monotone Property

According to (7), we can see intuitively that wider wire widths should be used for segments near the source (which minimizes the third term), and narrower widths for segments far away from the source (similar to a city water-pipe system). In fact, this turns out to be a very general property of any optimal wire width assignment based on the Elmore delay model.

Definition 1 *Given a routing tree T, a wiresizing solution $\mathcal{W}$ on T is a monotone assignment if $w_E \geq w_{E'}$ for any pair of segments E and E' such that $E \in Ans(E')$.*

Theorem 2 *For any given tree T, there exists a monotone optimal width assignment $\mathcal{W}^*$.* □

According to the monotone property, the optimal wire width assignment $\mathcal{W}^*$ can be represented by a set of "wavefronts" radiating outward from the source N_+. Each wavefront defines the boundary where the segment width decreases, as illustrated in Figure 3. These wavefronts do not intersect, except that they may touch each others at the nodes in the tree, and all the segments enclosed between two wavefronts have the same width.

3.3 Dominance Property

Definition 2 *Given two wire width assignments $\mathcal{W}$ and $\mathcal{W}'$, $\mathcal{W}$* dominates *$\mathcal{W}'$ if for any segment E, the width assignment of E in $\mathcal{W}$ is greater than or equal to that of E in $\mathcal{W}'$.*

Definition 3 *Given a routing tree T, a wire width assignment $\mathcal{W}$ on T, and any particular segment $E \in T$, a* local refinement *on E is the operation to optimize the width of E based on the objective function in (7), subject to the fixed assignment of $\mathcal{W}$ on the other segments.*

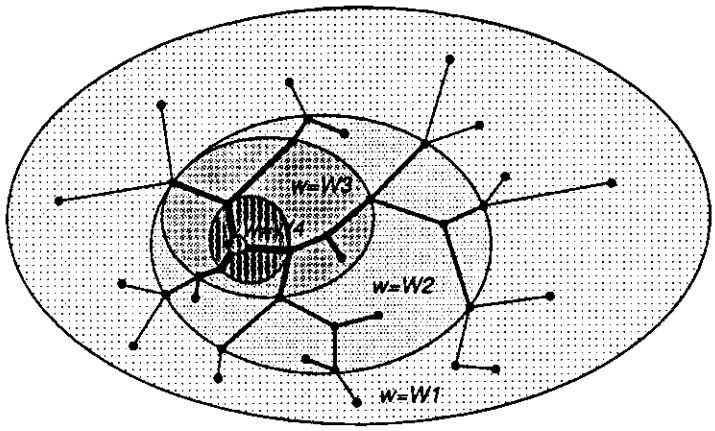

Figure 3: The optimal wire width assignment can be represented by a set of wavefronts.

Note that when w_E is the only variable in (7), we can rewrite (7) as:

$$T(\mathcal{W}) = \mathcal{A} + \mathcal{B} \cdot w_E + \mathcal{C} \cdot \frac{1}{w_E} \tag{10}$$

Local refinement of E in $\mathcal{W}$ can be obtained by minimizing (10) subject to the constraint $W_1 \leq w_E \leq W_r$. Since coefficients $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ can be computed in $O(n)$ time, and minimization of (10) can be carried out in constant time, a local refinement operation takes $O(n)$ time, where n is the number of segments in T. With definitions 2 and 3, we can derive the following theorem which is very useful in the development of a fast wiresizing algorithm in the next section.

Theorem 3 *Let $\mathcal{W}^*$ be an optimal width assignment. If a width assignment $\mathcal{W}$ dominates $\mathcal{W}^*$, then any local refinement of $\mathcal{W}$ still dominates $\mathcal{W}^*$. Similarly, if a width assignment $\mathcal{W}$ is dominated by $\mathcal{W}^*$, then any local refinement of $\mathcal{W}$ is dominated by $\mathcal{W}^*$.* □

The separability, the monotone property, and the dominance property were shown to be true for the optimal wiresizing solution under the upper-bound delay model in [7]. The results in this section show that these three properties are also true for the more complicated distributed Elmore delay model. In fact, these properties also hold in other general delay models as long as the F and G functions satisfy (8).

4 Wiresizing Algorithms

4.1 Optimal Wiresizing Algorithm

We first introduce the notion of a *single-stem tree* used in the following discussion. A single-stem tree is a tree with only one segment (called the *stem segment* of that tree) incident on its root (see Figure 4(a)). We use $SST(E)$ to denote the single-stem tree with stem E.

According to the separability, once E and every segment in $Ans(E)$ are assigned the appropriate widths,

the optimal wire width assignment for the single-stem subtrees $SST(E_{c1})$, $SST(E_{c2})$, $\cdots$, $SST(E_{cb})$ of the tree $SST(E)$ (with respect to the width assignment of E and segments in $Ans(E)$) can be *independently* determined, where the segments $E_{c1}, \cdots, E_{cb}$ are the children of E.

Assume we are given a single-stem tree with stem E, and a set of possible widths $\{W_1, W_2, \cdots, W_r\}$, we can determine the optimal assignment $\mathcal{W}^*$ on $T(E)$ by enumerating all the possible width assignments of E. For each of the possible width assignment W_k of E ($1 \leq k \leq r$), we determine the optimal assignment for each single-stem subtree $SST(E_{ci})$ ($1 \leq i \leq b$) of $SST(E)$ independently by recursively applying the same procedure to each $SST(E_{ci})$ with $\{W_1, W_2, \cdots, W_k\}$ as the set of possible widths (to guarantee the monotone property). The optimal assignment for E is the one which gives the smallest total delay.

If the original routing tree T is not a single-stem tree, however, we can decompose T into b single-stem trees, where b is the degree of the root of T, and apply the algorithm to each individual single-stem tree separately (see Figure 4(b)). The Optimal Wiresizing Algorithm under the Elmore Delay Model (OWSA/ED) is formally described in Table 1. In the algorithm, $\mathcal{W}(E)$ and $\mathcal{W}(SST(E))$ denote the wire width assignment of E, and the wire width assignments of segments in the single-stem subtree with stem E, respectively. All segments in the minimum-width assignment have width W_1.

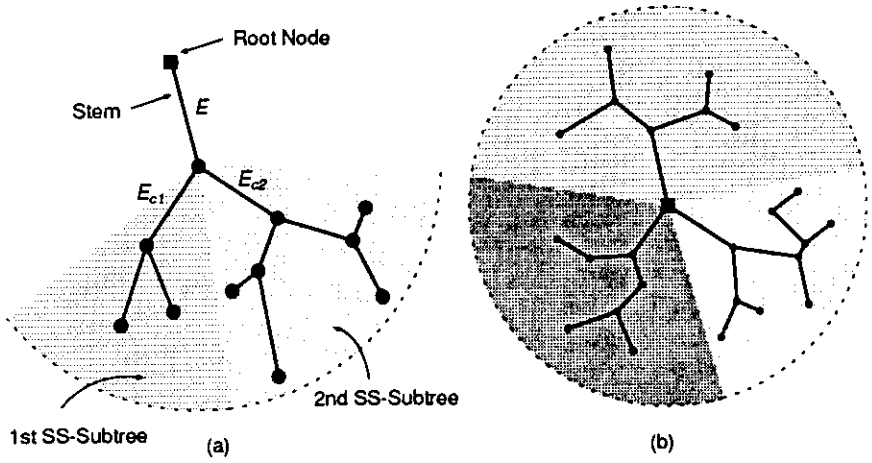

Figure 4: (a) A single-stem tree consists of a stem and a set of single-stem subtrees. In this example, E is the stem of the single-stem tree $SST(E)$, and $SST(E_{c1})$ and $SST(E_{c2})$ are the single-stem subtrees of $SST(E)$ (E_{c1} and E_{c2} are the children of E). (b) Any general tree T can be decomposed into a set of independent single-stem trees.

Theorem 4 *Given a routing tree with n segments and r possible wire widths, the worst case time complexity of OWSA/ED is $O(n^r)$.*

Proof: First, assume that T is a single-stem tree. Let $N(n, r)$ be the *maximum* number of calls to OWSA/ED among all possible single-stem routing trees with n segments and r possible choices of wire widths. For $n = 1$, or $r = 1$, it is easy to see that $N(1, r) = 1$ and $N(n, 1) = 1$. Consider a general n-segment single-stem tree with b single-stem subtrees connected to the stem. Assume that the i^{th} subtrees has n_i segments. For each

Optimal Wire Sizing Algorithm under Elmore Delay (OWSA/ED)

Function SST_OWSA/ED($SST(E), \mathcal{W}, monotone$)

```
/*
 * Given a single-stem subtree SST(E) with stem E, a partial wire width assignment W
 * (where the width assignments of the segments in SST(E) are not specified), the index
 * monotone which sets an upper bound W_monotone of segment widths in SST(E) due to
 * the monotone property, and a set of possible wire widths {W_1, W_2, ..., W_r}, return W
 * which includes the optimal wire width assignment of SST(E) under the distributed Elmore
 * delay model subject to the fixed width assignments of ancestors of E.
 */

W_best ← W;
W_best(SST(E)) ← minimum-width assignment;
for each width W_k, W_1 < W_k ≤ W_monotone do
    W(E) ← W_k;
    for each single stem subtree SST(E_ci) of SST(E) do
        W(SST(E)) ← SST_OWSA/ED(SST(E_ci), W, k);
    if delay(W) < delay(W_best) then
        W_best ← W;
end for;
return W_best;
end Function;

Function OWSA/ED(T)
/*
 * Given a tree T and a set of possible wire widths {W_1, W_2, ..., W_r}, return the optimal
 * wire width assignment of T under the distributed Elmore delay model obtained by applying
 * SST_OWSA/ED() to each single-stem subtree SST(E_i) of T.
 */

W ← minimum-width assignment;
for each single stem subtree SST(E_i) of T do
    W(SST(E_i)) ← SST_OWSA/ED(SST(E_i), W, W_r);
return W;
end Function;
```

Table 1: The Optimal Wiresizing Algorithm under the Elmore Delay Model (OWSA/ED).

width assignment W_j of the stem ($2 \leq j \leq r$), there are at most $\sum_{i=1}^{b} N(n_i, j)$ calls to OWSA/ED. Hence, the maximum number of calls to OWSA/ED would be at most $1 + \sum_{j=2}^{r} \sum_{i=1}^{b} N(n_i, j)$. In short, we can express $N(n, r)$ recursively as follows:

$$N(n,r) \leq \begin{cases} \max_{0<b\leq n-1, n_1+\cdots+n_b=n-1} \left\{1 + \sum_{j=2}^{r} \sum_{i=1}^{b} N(n_i, j)\right\} & n, r \geq 2 \\ 1 & n = 1 \text{ or } r = 1 \end{cases}$$

We proceed to prove that $N(n, r)$ is bounded by n^{r-1} using mathematical induction on n and r: For $n = 1$, we have $N(1, r) = 1 \leq 1^{r-1}$ for $r > 0$. For $r = 1$, we have $N(n, 1) = 1 \leq n^{1-1}$ for $n > 0$. Assume that $N(k, r) \leq k^{r-1}$ for $r > 0$ and $k \leq K$. Then,

$$N(K+1,r) \leq 1 + \max_{b, n_i>0, n_1+\cdots+n_b=K} \left\{ \sum_{j=2}^{r} \sum_{i=1}^{b} N(n_i, j) \right\}$$

$$
\begin{aligned}
&\leq 1 + \max_{b,n_i>0,n_1+\cdots+n_b=K} \left\{ \sum_{j=2}^{r} \sum_{i=1}^{b} n_i^{j-1} \right\} \\
&\leq 1 + \max_{b,n_i>0,n_1+\cdots+n_b=K} \left\{ \sum_{j=2}^{r} \left(\sum_{i=1}^{b} n_i \right)^{j-1} \right\} \\
&= 1 + \max_{b,n_i>0,n_1+\cdots+n_b=K} \left\{ \sum_{j=2}^{r} (K^{j-1}) \right\} \\
&= 1 + \sum_{j=1}^{r-1} K^j \leq 1 + \sum_{j=1}^{r-1} \binom{r-1}{j} \cdot K^j = (K+1)^{r-1}
\end{aligned}
$$

By induction, $N(n,r) \leq n^{r-1}$ for any n and r.

If T is not a single-stem tree, however, we can consider T as a union of b single-stem trees where b is the degree of the root (see Figure 4(b)), and apply OWSA/ED to determine the optimal wire width assignment for each single-stem subtree independently. The overall complexity is $O(n_1^{r-1}) + O(n_2^{r-1}) + \cdots + O(n_b^{r-1}) = O(n^{r-1})$, where n_i is the number of segments in the i^{th} single-stem tree, and $\sum_{i=1}^{b} n_i = n$. Each call to OWSA/ED requires at most $O(B \cdot r \cdot n)$ time (including linear time delay computation but excluding recursive calls to OWSA/ED), where B is the maximum degree in the tree. Both B and r can be considered as constants (in the Manhattan plane, $B \leq 4$, and r is a small constant in practice). Hence the overall complexity of OWSA/ED is $O(n^r)$. □

In essence, OWSA/ED enumerates *all* the possible combinations of monotone wire width assignments along every source-to-leaf path in the routing tree[1]. The complexity indeed can grow exponentially with respect to r (which is usually a small constant in practice). This is the case when the tree is simply a chain of segments, where the total number of possible assignments evaluated by OWSA/ED equals to $\binom{n+r-1}{r-1} = \Omega(n^{r-1})$.

Nevertheless, our optimal wiresizing algorithm is a significant improvement over the brute-force enumeration method which has complexity $O(r^n)$. In the next two subsections, we shall show how to further improve the runtime of the OWSA/ED algorithm. Note that although we assume that each segment has the same set of width choices $\{W_1, W_2, \cdots, W_r\}$, it is clear to see that the OWSA/ED algorithm can also handle the case where each segment width w_E has to satisfy a lower and upper bound constraint $W_{L(E)} \leq w_E \leq W_{U(E)}$ $(L(E) \leq U(E))$. In fact, this general version of the algorithm will be used in the FOWSA/ED algorithm in Section 4.3.

We would like to point out that a simple ***bottom-up dynamic programming*** approach, where the width assignment of each subtree is determined independent of its ancestors, does not produce optimal solutions in general. This is due to the fact that the optimal width assignment of any particular segment E depends on the wire width assignment of both its descendants and ancestors. In fact, our experimental results indicate that the wire width assignments generated by such an approach are in general relatively poor in quality.

[1] OWSA/ED, however, does not enumerate all the possible monotone assignments for the entire tree.

Greedy Wire Sizing Algorithm under Elmore Delay (GWSA/ED)

```
Function GWSA/ED(T, W_init)
  /*  Given a tree T, an initial wire width assignment W_init, and a set of possible wire widths
   *  {W_1, W_2, ..., W_r}, return the greedy wire width assignment under the distributed Elmore
   *  delay model.
   */

  W ← W_init;
  repeat
    progress ← false;
    for each segment E of T do
      w ← local_optimal_width(T, W, E);
      if w ≠ W(E) then
        progress ← true;
        W(E) ← w;
      end if
    end for;
  until progress = false;
  return W;
end Function;
```

Table 2: The Greedy Wiresizing Algorithm under the Elmore Delay Model (GWSA/ED).

4.2 Greedy Wiresizing Algorithm

In this subsection, we present a simple greedy approach based on an iterative refinement technique for efficient wire width assignment, which allow us to achieve significant speedup of the optimal wiresizing algorithm when we incorporate the greedy algorithm into the OWSA/ED algorithm: Starting with an initial wire width assignment (say, all segments have the minimum width), we traverse the tree and perform a local refinement on each segment whenever possible. This process is repeated until no improvement is achieved on any segment in the last round of traversal. The Greedy Wiresizing Algorithm with Elmore Delay Model (GWSA/ED) is described formally in Table 2, where `local_optimal_width`(T, $\mathcal{W}$, E) is a function returning the local optimal width of segment E in T (with respect to the rest of the width assignments in $\mathcal{W}$).

Despite its greedy nature, GWSA/ED performs very well in terms of the quality of assignments and runtime. Given a tree T with n segments, if we start with an assignment that is dominated by the optimal width assignment $\mathcal{W}^*$, say the minimum width assignment, each iteration will generate a better assignment (closer to the optimal) and still remain dominated by $\mathcal{W}^*$. Therefore, GWSA/ED will converge after at most $n \cdot (r-1)$ traversals. During each traversal, each segment is locally refined exactly once, and each refinement takes $O(n)$ time as shown in Section 3.3. As a result, the worst case complexity of GWSA/ED is $O(n^3 \cdot r)$.

Moreover, the dominance property suggests a strategy of using the GWSA/ED algorithm to compute the lower and upper bounds of each segment width of the optimal assignment. If we start with the minimum-width assignment where each segment has the minimum wire width (and is dominated by the optimal solution $\mathcal{W}^*$),

the resulting assignment computed by GWSA/ED gives a lower bound of the optimal width for each segment, since each intermediate assignment computed by GWSA/ED including the last one, is dominated by $\mathcal{W}^*$. Similarly, if we start with the maximum-width assignment, the resulting assignment computed by GWSA/ED gives an upper bound of the optimal width for each segment.

In most circumstances in our experiments, we are able to obtain identical lower and upper bounds of all segments in the tree using the GWSA/ED algorithm, which lead to an optimal assignment.

4.3 The Combined Approach to the Wiresizing Problem

We have presented an $O(n^r)$ time optimal wiresizing algorithm, and a fast $O(n^3 \cdot r)$ time greedy wiresizing algorithm. It is not difficult to see that these two algorithms can be combined into a new algorithm which guarantees the optimal assignment and runs extremely fast. The combined algorithm, called the Fast Optimal Wiresizing Algorithm under the Elmore Delay Model (FOWSA/ED), is described as follows:

First, we obtain the lower and upper bounds $W_{L(E)}$ and $W_{U(E)}$ of each wire segment E using the GWSA/ED algorithm. Then, we run the OWSA/ED algorithm in Section 4.1 with lower and upper bounds on each segment E such that $W_{L(E)} \leq w_E \leq W_{U(E)}$. Since the lower and upper bounds of each segment obtained from the GWSA/ED algorithm are very close or even identical in most cases, the total number of candidate assignments ever generated by FOWSA/ED algorithm is much smaller than that by the OWSA/ED algorithm alone. As a result, the upper and lower bounds obtained by GWSA/ED help to speedup the optimal algorithm significantly. Since in most cases the optimal wiresizing solutions are completely determined by the upper and lower wire width bounds computed by GWSA/ED(i.e. $W_{L(E)} = W_{U(E)}$ for most segments E), FOWSA/ED is competitive with GWSA/ED in runtime while it guarantees the optimality of the wiresizing solution.

5 Experimental Results

We have implemented the optimal OWSA/ED algorithm, the greedy GWSA/ED algorithm, and the combined FOWSA/ED algorithm in ANSI C for the IBM-PC and Sun SPARC station environment. We have tested the wiresizing algorithms on both the MCM and the advanced IC technologies on signal nets of 4 and 8 sinks. The MCM and IC technology parameters are summarized in Table 3. The IC technology parameters are based on the $0.5\mu m$ CMOS process provided by North Carolina Microelectronic Center (MCNC), and the MCM technology parameters were obtained from [10].

5.1 Comparisons Between Different Wiresizing Solutions

We have compared our FOWSA/ED wiresizing solutions with the minimum-width solution (MIN), the maximum-width solution (MAX), and the wiresizing solutions by OWSA based on the upper-bound RC delay model in [7]. In our experiment, the set of wire width allowed is $\{W_1, 2W_1, 3W_1, 4W_1\}$, where W_1 is the minimum width.

Technology:	Integrated Circuits (ICs)	Multi-Chip Modules (MCMs)
Driver Resistance:	156 Ω	25 Ω
Unit Wire Resistance:	0.112 $\Omega/\mu m$	0.008 $\Omega/\mu m$
Loading Capacitance:	1 fF	1000 fF
Unit Wire Capacitance:	0.039 $fF/\mu m$	0.060 $fF/\mu m$
Total Area:	5 mm x 5 mm (200 grids x 200 grids)	100 mm x 100 mm (4000 grids x 4000 grids)

Table 3: Technology parameters based on advanced IC and MCM designs.

Hence, every segment in MIN has width W_1, and every segment in MAX has width $4W_1$. The placement of 100 4-sink nets and 100 8-sink nets were generated randomly and routed by the batched 1-Steiner algorithm[16]. For each Steiner tree, a critical sink is chosen randomly. The delay to the critical sink(s) and the total wiring area of the different wiresizing solutions are compared. The signal delay is computed using the two-pole circuit simulator developed by Zhou et al., which was shown to produce results comparable to SPICE in term of delay simulation, but runs much faster [26]. Table 4 summarizes the averages of the delays and areas for the 4-sink and 8-sink nets used in different wiresizing solutions, based on the IC and MCM parasitic parameters, respectively.

IC	Delay (ns)				Normalized Wiring Area	
# Sinks	MIN	MAX	OWSA	FOWSA/ED	OWSA	FOWSA/ED
4	0.238	0.497 (+109.01%)	0.224 (-5.88%)	0.220 (-7.42%)	1.2745	1.2422
8	0.327	0.706 (+116.00%)	0.300 (-8.05%)	0.288 (-12.01%)	1.3599	1.2719
MCM	Delay (ns)				Normalized Wiring Area	
# Sinks	MIN	MAX	OWSA	FOWSA/ED	OWSA	FOWSA/ED
4	7.906	7.259 (-8.18%)	4.777 (-39.57%)	4.391 (-44.50%)	2.3677	1.8565
8	13.899	11.860 (-14.67%)	7.671 (-44.82%)	6.750 (-51.44%)	2.3762	1.7214

Table 4: Comparisons of the average delay (in nanoseconds) and normalized wiring area among different wiresizing algorithms. The normalized wiring areas for MIN and MAX are 1.0000 and 4.0000, respectively.

We can see from Table 4 that our wiresizing solution reduces interconnect delay by up to 51% when compared to the minimum-width solution (MIN) of the same routing topology. Furthermore, compared to the wiresizing solution obtained by OWSA, our wiresizing solution reduces the total wiring area by up to 28% while further reducing the interconnect delays to the timing-critical sinks by up to 12%. Although area minimization was not considered here, one can use the combined delay and area objective shown in Section 2.3 to achieve delay and area tradeoff in wiresizing optimization, which will be demonstrated later in this section.

5.2 Effects of Multiple Critical Sinks

We have studied the effect of multiple critical sinks on the overall quality of the wiresizing solution by FOWSA/ED as compared to the minimum-width solution and the solution obtained by the OWSA algorithm in [7]. Figure 5 shows the different wiresizing solutions for a typical Steiner routing tree under the MCM

technology. The width assignments of the trees are obtained by the following ways: (a) using the minimum-width solution (MIN); (b) by OWSA based on the upper-bound RC delay model[7]; (c) by FOWSA/ED with a single crifical sink X; (d) by FOWSA/ED with a single critical sink Y; (e) by FOWSA/ED with two critical sinks X and Y; (f) by FOWSA/ED with all sinks being critical. The delay and the normalized wiring area are shown in Table 5.

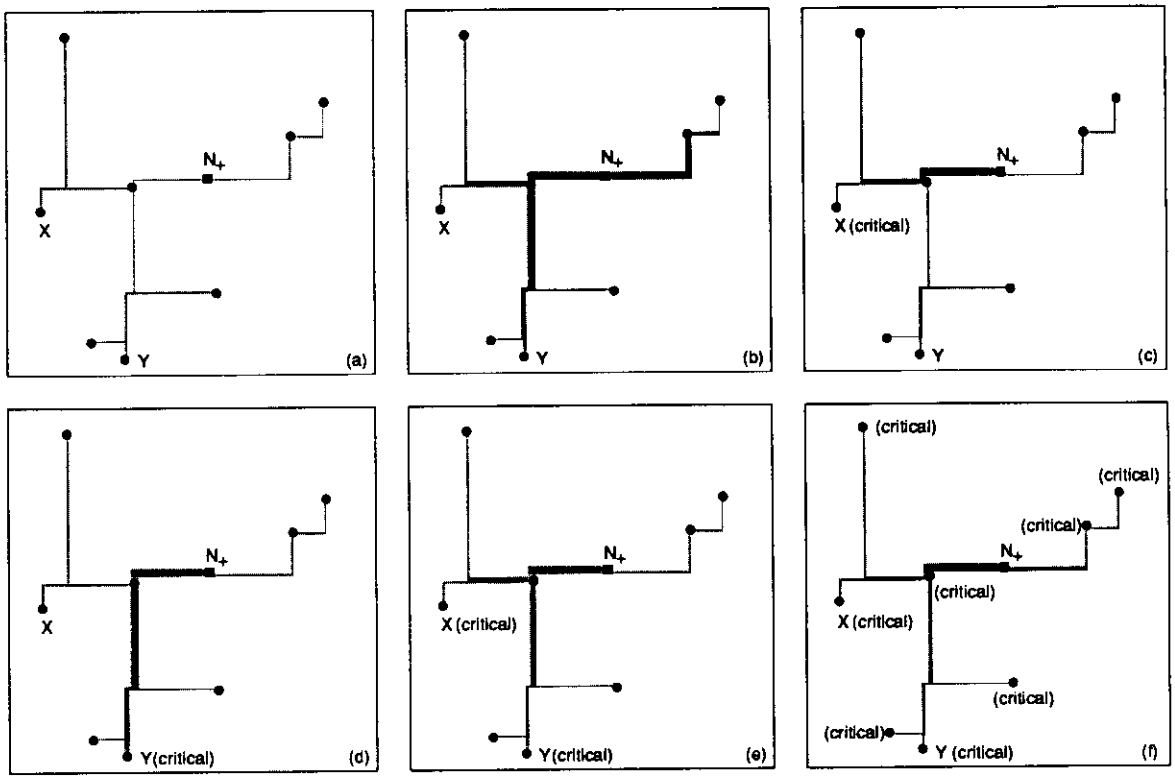

Figure 5: Different wiresizing solutions obtained using the following algorithms: (a) the minimum width; (b) by OWSA based on the RC delay model; (c) by FOWSA/ED with a single critical sink X; (d) by FOWSA/ED with a single critical sink Y; (e) by FOWSA/ED with two critical sinks X and Y; (f) by FOWSA/ED with all sinks being critical.

Comparison	MIN	OWSA	FOWSA/ED (X critical)	FOWSA/ED (Y critical)	FOWSA/ED (X, Y critical)	FOWSA/ED (all critical)
t(X) (ns)	9.679	5.824	* 4.560	6.310	* 4.945	* 5.008
t(Y) (ns)	12.117	6.773	8.116	* 6.141	* 6.651	* 6.715
$\frac{t(X)+t(Y)}{2}$ (ns)	10.898	6.299	6.338	6.226	5.798	5.862
t(ALL) (ns)	8.962	5.611	5.762	5.638	5.331	5.222
Normalized Area	1.0000	2.3267	1.4554	1.7525	1.5050	1.5941

Table 5: Comparisons (delay and normalized wiring area) among wiresizing assignments obtained by different algorithms (with MCM technology). Numbers with an asterisk (*) are critical delays. t(X) and t(Y) are the delays at X and Y, respectively, and t(ALL) is the average delay to all sinks.

This experiment shows that FOWSA/ED outperforms OWSA in terms of both delay and wiring area

reduction even in the presence of multiple critical sinks. Comparing Figure 5(c), Figure 5(d), and Figure 5(e), we can see that FOWSA/ED assigns wider widths only to segments along the critical path(s), and at the same time try to minimize widths of other segments. In general, when the number of critical sinks increases, delay to each sink (such as t(X) and t(Y)) increases, but the average delay to the critical sinks (such as $\frac{t(X)+t(Y)}{2}$) decreases. Furthermore, comparison between Figure 5(b) and 5(f) reveals that the OWSA algorithm, which uses the upper-bound RC delay model, indeed tends to over-size wires: even when every sink is critical, FOWSA/ED still gives smaller delay and wiring area than the OWSA solution.

5.3 Tradeoff Between Area and Delay Minimization

We have studied the tradeoff between area and delay minimization. In our experiment, the same set of 8-sink MCM signal nets (with the same critical sink as chosen before) in the previous subsection is used, and the average areas and average delays under different area/delay tradeoff are compared. Instead of choosing a pair of fixed α and β for all the interconnect structures, we used a scaled objective

$$\alpha \cdot \frac{\textbf{Area}}{\textbf{OptArea}} + \beta \cdot \frac{\textbf{Delay}}{\textbf{OptDelay}}$$

in our studies, where α and β are fixed constants with $\alpha + \beta = 1$, and **OptArea** and **OptDelay** are the optimal area and optimal delay, respectively. Note that **OptArea** and **OptDelay** are constants for a given interconnect structure. The use of the scaled objective avoids the problem of choosing a different pair of α and β in (9) for each interconnect structure. The comparisons are shown in Figure 6, where a smooth tradeoff between area and delay optimization is observed. Notice that the two ends of the tradeoff spectrum are the optimal wiresizing solution ($\alpha = 0, \beta = 1$) and the minimum-width solution ($\alpha = 1, \beta = 0$), respectively.

6 Extension

In actual layout designs, interconnects are often routed using more than one layer, and each layer may have different parasitic characteristics. As a result, unit wire resistance r_0 and unit wire capacitance c_0 are different for segments in different layers. In this section, we will extend our wiresizing algorithms to cope with this problem.

Assume there are a total of M possible routing layers. We represent the values of the unit wire resistance and unit wire capacitance of the l^{th} layer by $q_l^r \cdot r_0$ and $q_l^c \cdot c_0$, respectively. In other words, q_l^r and q_l^c are the *scaling factors* of the unit wire resistance and unit wire capacitance of the l^{th} layer with respect to some reference wire resistance (r_0) and wire capacitance (c_0). Segments in the l^{th} layer has a discrete set of choices of wire widths $\{W_1^l, W_2^l, \cdots, W_{r(l)}^l\}$. Given an interconnect tree T routed in M layers, our objective is again to compute the optimal wire width assignments of all the segments in T to minimize the weighted delay to the critical sinks as in Section 2. Notice that when there is only a single layer, $M = 1$ and $q_1^r = q_1^c = 1$.

We can carry out a similar sequence of simplifications and transformations as previously done. Given a

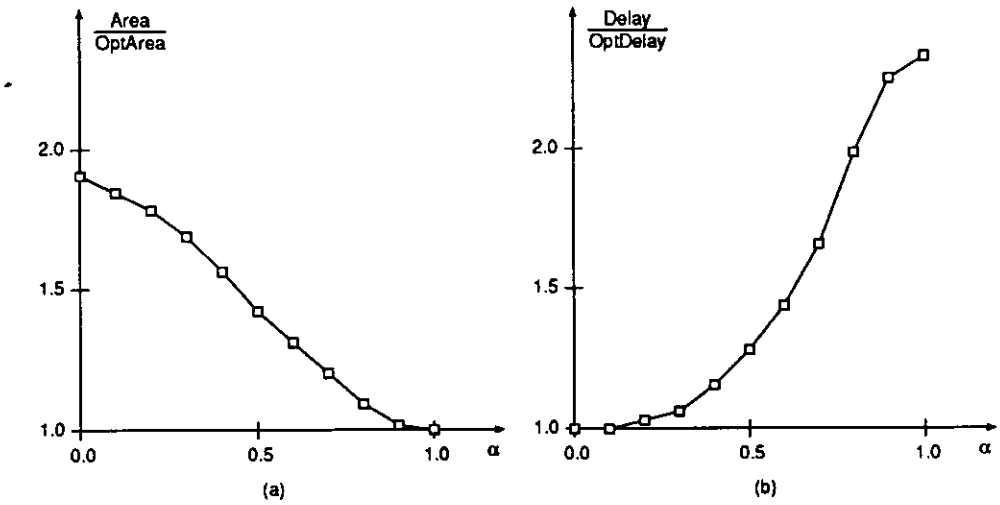

Figure 6: Tradeoff between area optimization and delay optimization as captured by the combined objective function $\alpha \cdot \frac{\mathbf{Area}}{\mathbf{OptArea}} + \beta \cdot \frac{\mathbf{Delay}}{\mathbf{OptDelay}}$, with $\alpha + \beta = 1$: (a) normalized wiring area as a function of α; (b) normalized delay to the critical sink as a function of α.

segment E routed on the k^{th} layer, we define $layer(E) = k$, $q^r(E) = q_k^r$, and $q^c(E) = q_k^c$. It is easy to see that our new objective is to find a wiresizing solution $\mathcal{W}$ which minimizes:

$$T(\mathcal{W}) = \mathcal{K}_3 \cdot \sum_{E \in T} l_E \cdot q^c(E) \cdot w_E + \mathcal{K}_4 \cdot \sum_{E,E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot F(E,E') \cdot q^r(E) \cdot q^c(E') \cdot \frac{w_{E'}}{w_E} + \mathcal{K}_5 \cdot \sum_{E \in T} l_E \cdot G(E) \cdot q^r(E) \cdot \frac{1}{w_E} \quad (11)$$

6.1 Separability, Monotone Property, and Dominance Property

We can show that the separability and the dominance property can be directly extended to the generalized wiresizing problem. However, to generalize the monotone property, we need to properly extend the definition of *monotone assignment*:

Definition 1 *(Generalized Monotone Assignment) Given a routing tree T, a wiresizing solution $\mathcal{W}$ on T is a monotone assignment if $w_E \geq w_{E'}$ for any pair of segments E and E' in the same routing layer with $E \in Ans(E')$.*

In essense, monotonicity is maintained within each routing layer. With this extension, we can show that the following three theorems in Section 3 still hold (the proofs of which are given in the Appendix):

Theorem 1 *(Separability) For the generalized wiresizing problem, if the width assignment of a path P originated from the source is given, the optimal width assignment for each subtree branching off P can be carried out independently.* □

Theorem 2 *(Monotone Property) For any given tree T routed in multiple layers, there exists a monotone optimal width assignment $\mathcal{W}^*$.* □

Theorem 3 *(Dominance Property) Let $\mathcal{W}^*$ be an optimal width assignment for the generalized wiresizing problem. If a width assignment $\mathcal{W}$ dominates $\mathcal{W}^*$, then any local refinement of $\mathcal{W}$ still dominates $\mathcal{W}^*$. Similarly, if a width assignment $\mathcal{W}$ is dominated by $\mathcal{W}^*$, then any local refinement of $\mathcal{W}$ is dominated by $\mathcal{W}^*$.* □

6.2 Extended Wiresizing Algorithms

With the generalized separability and monotone property, we can extend the OWSA/ED algorithm properly to handle multiple routing layers. In the original OWSA/ED algorithm, the separability guarantees the correctness of a dynamic-programming-like assignments of wire widths, while the monotone property is used to limit the search space. We can extend these ideas to the construction of the generalized OWSA/ED algorithm, named EOWSA/ED, as follows: Instead of passing the index of the maximum allowed width (*monotone*) as a parameter as in the recursive call to FOWSA/ED, we pass an *array* of indices of maximum allowed widths (*monotone*$[1..M]$) as a parameter to EOWSA/ED. The l^{th} element in the array is the maximum allowed width for the l^{th} layer, which is used to enforce the monotone requirement in *all* layers. It should be clear that when $M = 1$ (a single layer), EOWSA/ED is the same as OWSA/ED.

On the other hand, the GWSA/ED algorithm can be used to handle multiple routing layers without any change, except replacing the objective function (7) by (11).

Again, we can combine the optimal algorithm and the greedy algorithm to get a wiresizing algorithm which guarantees the optimality of the assignment solution and runs extremely fast. The new algorithm, called the Extended Fast Optimal Wiresizing Algorithm under the Elmore Delay Model (EFOWSA/ED) is constructed from EOWSA/ED and EGWSA/ED in exactly the same fashion as FOWSA/ED is constructed from OWSA/ED and GWSA/ED. The optimal EOWSA/ED algorithm, the greedy EGWSA/ED algorithm, and the combined EFOWSA/ED algorithm are described in Table 6–8, respectively. As before, $\mathcal{W}(E)$ and $\mathcal{W}(SST(E))$ denote the wire width assignment of E, and the wire width assignments of segments in the single-stem subtree with stem E, respectively. In the minimum-width assignment, segments in the l^{th} layer has width W_1^l.

6.3 Combined Delay and Area Minimization

Although the same delay/area tradeoff formulation in (9) can be generalized to handle multiple wiring layers, it is no longer captured by the generalized delay formulation in (11). Instead, we have the following combined formulation:

$$\alpha \cdot \textbf{Area} + \beta \cdot \textbf{Delay}$$

Extended Optimal Wire Sizing Algorithm under Elmore Delay (EOWSA/ED)

Function SST_EOWSA/ED($SST(E), \mathcal{W}, \mathcal{W}_{lower}, \mathcal{W}_{upper}, monotone[1..M]$)

```
/*
 * Given a single-stem subtree SST(E) with stem E, a partial wire width assignment W
 * (where the width assignments of the segments in SST(E) are not specified), the lower-
 * bound width assigment W_lower, the upper-bound width assigment W_upper, the array
 * monotone[1..M] of indices which sets an upper bound W^l_{monotone[l]} of segment widths
 * in SST(E) for each layer l (1 <= l <= M) due to the monotone property, and a set of
 * possible wire widths {W^l_1, W^l_2, ..., W^l_{r(l)}} for each layer l (1 <= l <= M), return W which
 * includes the optimal wire width assignment of SST(E) under the distributed Elmore delay
 * model subject to the fixed width assignments of ancestors of E.
 */

l ← layer(E);
W_best ← W;
W_best(SST(E)) ← minimum-width assignment;
for each width W^l_k, W_lower(E) < W^l_k ≤ min{W^l_{monotone[l]}, W_upper(E)} do
    W(E) ← W^l_k;
    monotone[l] ← k;
    for each single stem subtree SST(E_ci) of SST(E) do
        W(SST(E_ci)) ← SST_EOWSA/ED(SST(E_ci), W, W_lower, W_upper, monotone[1..M]);
    if delay(W) < delay(W_best) then
        W_best ← W;
end for;
return W_best;
end Function;
```

Function EOWSA/ED($T, \mathcal{W}_{lower}, \mathcal{W}_{upper}$)

```
/*
 * Given a tree T, the lower-bound width assigment W_lower, the upper-bound width assigment
 * W_upper, and a set of possible wire widths {W^l_1, W^l_2, ..., W^l_{r(l)}} for each layer l (1 <= l <= M),
 * return the optimal wire width assignment of T under the distributed Elmore delay model
 * obtained by applying SST_EOWSA/ED() to each single-stem subtree SST(E_i) of T.
 */

W ← minimum-width assignment;
for each single stem subtree SST(E_i) of T do
    for each 1 <= l <= M do
        monotone[l] ← r(l)
    W(SST(E_i)) ← SST_EOWSA/ED(SST(E_i), W, W_lower, W_upper, monotone[1..M]);
return W;
end Function;
```

Table 6: The Extended Optimal Wiresizing Algorithm under the Elmore Delay Model (EOWSA/ED).

$$
\begin{aligned}
= \quad & \left\{ \alpha \cdot \sum_{E \in T} l_E \cdot w_E \; + \; \beta \mathcal{K}_3 \cdot \sum_{E \in T} l_E \cdot q^c(E) \cdot w_E \right\} \; + \\
& \beta \mathcal{K}_4 \cdot \sum_{E, E' \in T, E \neq E'} l_E \cdot l_{E'} \cdot q^r(E) \cdot q^c(E') \cdot F(E, E') \cdot \frac{w_{E'}}{w_E} \; + \\
& \beta \mathcal{K}_5 \cdot \sum_{E \in T} l_E \cdot G(E) \cdot q^r(E) \cdot \frac{1}{w_E}
\end{aligned}
$$

Fortunately, it can be shown that optimization under this objective satisfies the separability, the monotone property, and the dominance property. As a result, we can still apply the EFOWSA/ED algorithm to compute

Extended Greedy Wire Sizing Algorithm under Elmore Delay (EGWSA/ED)

```
Function EGWSA/ED(T, W_init)
  /*
   *  Given a tree T, an initial wire width assignment W_init, and a set of possible wire widths
   *  {W_1^l, W_2^l, ..., W_r(l)^l} for each layer l (1 ≤ l ≤ M), return the greedy wire width assignment
   *  under the distributed Elmore delay model.
   */

  W ← W_init;
  repeat
    progress ← false;
    for each segment E of T do
      w ← local_optimal_width(T, W, E);
      if w ≠ W(E) then
        progress ← true;
        W(E) ← w;
      end if
    end for;
  until progress = false;
  return W;
end Function;
```

Table 7: The Extended Greedy Wiresizing Algorithm under the Elmore Delay Model (EGWSA/ED).

Extended Fast Optimal Wire Sizing Algorithm under Elmore Delay (EFOWSA/ED)

```
Function EFOWSA/ED(T)
  /*
   *  Given a tree T and a set of possible wire widths {W_1^l, W_2^l, ..., W_r(l)^l} for each layer l
   *  (1 ≤ l ≤ M), call EGWSA/ED() to find the lower-bound and upper-bound wire width
   *  assignments, and then return the optimal wire width assignment under the distributed
   *  Elmore delay model obtained by calling EOWSA/ED() with the lower-bound and upper bound
   *  information.
   */

  for each segment E ∈ T do
    l ← layer(E);
    W_lower(E) ← W_1^l;
    W_upper(E) ← W_r(l)^l;
  end for;
  W_lower ← EGWSA/ED(T, W_lower);
  W_upper ← EGWSA/ED(T, W_upper);
  return EOWSA/ED(T, W_lower, W_upper);
end Function;
```

Table 8: The Extended Fast Optimal Wiresizing Algorithm under the Elmore Delay Model (EFOWSA/ED).

the optimal wiresizing solution which gives the best tradeoff between area and delay.

7 Conclusion and Future Work

The results in this paper have shown convincingly that proper sizing of the wire segments in a routing tree can lead to significant reduction in the interconnect delay. Although the Elmore delay model is more complicated than the upper-bound delay model used in [7], our study has shown that optimal wiresizing under the distributed Elmore delay model can still be achieved in polynomial time very efficiently when the separability, the monotone property, and the dominance property are used to prune suboptimal wiresizing solutions. The use of the distributed Elmore delay model successfully avoids the over-sizing problem in [7] and leads to additional reduction of the delays to the timing-critical sinks. We have further extended the problem formulation to handle wiresizing in multiple routing layers, and proved that the separability, the monotone property, and the dominance property are still valid in the generalized formulation. As a result, the general wiresizing problem for multiple routing layers can also be solved efficiently.

Allowing variable-width routing considerably complicates the global and detailed routing steps in VLSI layout design. The focus of our research in the next step will be on the development of channel routers and general area routers which can generate compact routing solutions of non-uniform wire width. Furthermore, our currently wiresizing algorithm optimizes each routing tree independently. It will be of practical interest to develop efficient wiresizing algorithms and detailed routers which can take into consideration the mutual capacitance and inductance between different routing trees.

Acknowledgments

This work is partially supported by the National Science Foundation under grant MIP-9110450 and the NSF Young Investigator Award, and by ARPA/CSTO under Contract J-FBI-93-112. The authors would like to thank Prof. Dian Zhou and Dr. David Gao for their helpful discussions, and Cheng-Kok Koh for his comments on the paper.

References

[1] C. J. Alpert, T. C. Hu, J. H. Huang, and A. B. Kahng, "A Direct Combination of the Prim and Dijkstra Constructions for Improved Performance-Driven Routing", *Proc. IEEE Int'l Symp. on Circuits and Systems*, May 1993, pp. 1869-1872.

[2] K. D. Boese, A. B. Kahng, and G. Robins, "High-Performance Routing Trees With Identified Critical Sinks", *Proc. ACM/IEEE Design Automation Conf.*, 1993, pp. 182-187.

[3] K. D. Boese, A. B. Kahng, Bernard A. McCoy, and G. Robins, "Toward Optimal Routing Trees", *4th ACM/SIGDA Physical Design Workshop*, 1993.

[4] C. Chiang, M. Sarrafzadeh, and C. K. Wong, "Global Routing based on Steiner Min-Max Trees", *Proc. IEEE Int'l. Conf. on Computer-Aided Design*, 1989, pp. 2-5.

[5] J. P. Cohoon and L. J. Randall, "Critical Net Routing", *Proc. IEEE Int'l. Conf. on Computer Design*, 1991, pp. 174-177.

[6] J. Cong, A. B. Kahng, G. Robins, M. Sarrafzadeh, and C. K. Wong, "Provably Good Performance-Driven Global Routing", *IEEE Trans. on CAD*, 11(6), June 1992, pp. 739-752.

[7] J. Cong, K. S. Leung, and D. Zhou, "Performance-Driven Interconnect Design Based on Distributed RC Delay Model", *Proc. ACM/IEEE Design Automation Conf.*, 1993, pp. 606-611.

[8] J. Cong and K. S. Leung, "Optimal Wiresizing Under the Distributed Elmore Delay Model", *Proc. IEEE Int'l. Conf. on Computer-Aided Design*, 1993, pp. 634-639.

[9] J. Cong and B. Preas, "A New Algorithm for Standard Cell Global Routing", *Proc. IEEE Int'l. Conf. on Computer-Aided Design*, 1988, pp. 176-179.

[10] W. Dai, Private Communication, 1992.

[11] W. E. Donath, R. J. Norman, B. K. Agrawal, and S. E. Bello, "Timing Driven Placement Using Complete Path Delays", *Proc. ACM/IEEE Design Automation Conf.*, 1990, pp. 84-89.

[12] A. E. Dunlop, V. D. Agrawal, D. N. Deutsh, M. F. Jukl, P. Kozak, and M. Wiesel, "Chip Layout Optimization Using Critical Path Weighting", *Proc. ACM/IEEE Design Automation Conf.*, 1990, pp. 133-136.

[13] W. C. Elmore, "The Transient Response of Damped Linear Network with Particular Regard to Wideband Amplifier", *J. Applied Physics*, 19(1948), pp. 55-63.

[14] A. L. Fisher and and H. T. Kung, "Synchronizing Large Systolic Arrays", *Proc. SPIE* 341, May 1982, pp. 44-52.

[15] X. Hong, T. Xue, E. S. Kuh, C. K. Cheng, and J. Huang, "Performance-Driven Steiner Tree Algorithms For Global Routing", *Proc. ACM/IEEE Design Automation Conf.*, 1993, pp. 177-181.

[16] A. B. Kahng and G. Robins, "A New Class of Iterative Steiner Tree Heuristics with Good Performance", *Proc. IEEE Int'l. Conf. on Computer-Aided Design*, July 1992, pp. 893-902.

[17] E. Kuh, M. A. B. Jackson, and M. Marek-Sadowska, "Timing-Driven Routing for Building Block Layout", *Proc. IEEE International Symposium on Circuits and Systems*, 1987, pp. 518-519.

[18] K. W. Lee and C. Sechen, "A New Global Router for Row-Based Layout", *Proc. IEEE Int'l. Conf. on Computer-Aided Design*, 1988, pp. 180-183.

[19] A. Lim, S. W. Cheng, and C. T. Wu, "Performance Oriented Rectilinear Steiner Trees", *Proc. ACM/IEEE Design Automation Conf.*, 1993, pp. 171-176.

[20] S. Prastjutrakul and W. J. Kubitz, "A Timing-Driven Global Router for Custom Chip Design", *Proc. IEEE Int'l. Conf. on Computer-Aided Design*, 1990, pp. 48-51.

[21] S. Pullela, N. Menezes, and L. T. Pillage, "Reliable Non-Zero Skew Clock Trees Using Wire Width Optimization", *Proc. ACM/IEEE Design Automation Conf.*, 1993, pp. 165-170.

[22] J. Rubinstein, P. Penfield, and M. A. Horowitz, "Signal Delay in RC Tree Networks", *IEEE Trans. on CAD*, 2(3) (1983) pp. 202-211.

[23] S. Sutanthavibul and E. Shragowitz, "An Adaptive Timing-Driven Layout for High Speed VLSI", *Proc. ACM/IEEE Design Automation Conf.*, 1990, pp. 90-95.

[24] R. S. Tsay, "Exact Zero Skew", *Proc. IEEE Int'l Conf. on Computer-Aided Design*, 1991, pp. 336-339.

[25] D. Zhou, F. P. Preparata, and S. M. Kang, "Interconnection Delay in Very High-speed VLSI", *IEEE Trans. on Circuits and Systems* 38(7), 1991.

[26] D. Zhou, S. Su, F. Tsui, D. S. Gao, and J. Cong, "A Distributive RCL-Model for MCM Layout", *Proc. of IEEE Multichip Module Conf.*, March 1993, pp. 191-197.

[27] Q. Zhu, W. W. M. Dai, and J. G. Xi, "Optimal Sizing of High-Speed Clock Networks Based on Distributed RC and Lossy Transmission Line Models", *Proc. IEEE Int'l. Conf. on Computer-Aided Design*, 1993, pp. 628-633.

Appendix

We shall prove the *Separability*, the *Monotone Property*, and the *Dominance Property* for the multi-layer wiresizing problem, in which different physical routing layers have different resistive and capacitive characteristics. The corresponding properties for the original wiresizing problem stated in Section 3 (with uniform unit wire resistance r_0 and unit wire capacitance c_0), being a special case when the number of routing layer is one, can be derived once we have established the proofs of the general case.

1 Separability

Theorem 1 *For the generalized wiresizing problem, if the width assignment of a path P originated from the source is given, the optimal width assignment for each subtree branching off P can be carried out independently.*

Proof: Let P be a path originated from the source, and $B = \{T_1, T_2, \cdots T_m\}$ be the set of subtrees branching off P (as illustrated in Figure 2). To simplify the notations, we define:

$$\begin{aligned}
\lambda(E) &= \mathcal{K}_3 \cdot l_E \cdot q^c(E) \\
\mu(E, E') &= \mathcal{K}_4 \cdot l_E \cdot l'_E \cdot F(E, E') \cdot q^r(E) \cdot q^c(E) \\
\nu(E) &= \mathcal{K}_5 \cdot l_E \cdot G(E) \cdot q^r(E)
\end{aligned}$$

We then rewrite (11) as a function of the subtrees in B and the path P:

$$\begin{aligned}
T(\mathcal{W}) = & \sum_{E \in P} \lambda(E) \cdot w_E + \sum_{b \in B} \left\{ \sum_{E \in b} \lambda(E) \cdot w_E \right\} + \\
& \sum_{E, E' \in P, E \neq E'} \mu(E, E') \cdot \frac{w_{E'}}{w_E} + \\
& \sum_{b \in B} \left\{ \sum_{E \in b, E' \in P} \mu(E, E') \cdot \frac{w_{E'}}{w_E} \right\} + \sum_{b \in B} \left\{ \sum_{E \in P, E' \in b} \mu(E, E') \cdot \frac{w_{E'}}{w_E} \right\} + \\
& \sum_{b \in B} \left\{ \sum_{E, E' \in b, E \neq E'} \mu(E, E') \cdot \frac{w_{E'}}{w_E} \right\} + \sum_{b, b' \in B, b \neq b'} \left\{ \sum_{E \in b, E' \in b'} \mu(E, E') \cdot \frac{w_{E'}}{w_E} \right\} + \\
& \sum_{E \in P} \nu(E) \cdot \frac{1}{w_E} + \sum_{b \in B} \left\{ \sum_{E \in b} \nu(E) \cdot \frac{1}{w_E} \right\}
\end{aligned}$$

We define:

$$\begin{aligned}
\mathcal{H}_{P,1} &= \sum_{E \in P} \lambda(E) \cdot w_E + \sum_{E, E' \in P, E \neq E'} \mu(E, E') \cdot \frac{w_{E'}}{w_E} + \sum_{E \in P} \nu(E) \cdot \frac{1}{w_E} \\
\mathcal{H}_{P,2}(b) &= \sum_{E \in b} \lambda(E) \cdot w_E + \sum_{E \in b, E' \in P, E \neq E'} \mu(E, E') \cdot \frac{w_{E'}}{w_E} + \sum_{E \in P, E' \in b, E \neq E'} \mu(E, E') \cdot \frac{w_{E'}}{w_E} + \\
& \sum_{E, E' \in b, E \neq E'} \mu(E, E') \cdot \frac{w_{E'}}{w_E} + \sum_{E \in b} \nu(E) \cdot \frac{1}{w_E}
\end{aligned}$$

We then have:

$$T(\mathcal{W}) = \mathcal{H}_{P,1} + \sum_{b \in B} \mathcal{H}_{P,2}(b) + \sum_{b,b' \in B, b \neq b'} \left\{ \sum_{E \in b, E' \in b'} \mu(E, E') \cdot \frac{w_{E'}}{w_E} \right\} \quad (12)$$

where $\mathcal{H}_{P,1}$ is a constant because it depends only on the width assignment of the path P. For any $b \in B$, $\mathcal{H}_{P,2}(b)$ is a function whose only variables are the widths of the segments in the subtree b (assuming that the segment widths in P are given).

Given two segments $E \in b_i$ and $E' \in b_j$ such that $i \neq j$, we have $F(E, E') = 0$ and hence $\mu(E, E') = 0$ (see definition of $F(E, E')$ in Section 2.2) since E' is not a descendent of E. Therefore, the third term in (12) is zero, and we conclude that:

$$T(\mathcal{W}) = \mathcal{H}_{P,1} + \sum_{b \in B} \mathcal{H}_{P,2}(b) + 0 = \mathcal{H}_{P,1} + \sum_{b \in B} \mathcal{H}_{P,2}(b)$$

Since the contribution from each subtree b to the summation is independent of each other, $T(\mathcal{W})$ is optimized if and only if for each $b \in B$, $\mathcal{H}_{P,2}(b)$ is optimized. Separability follows as a consequence. □

2 Monotone Property

Theorem 2 *For any given tree T routed in multiple layers, there exists a monotone optimal width assignment $\mathcal{W}^*$.*

Proof: Assume that the monotone property fails for a tree T, that is, all the optimal width assignments are not monotone. Then for any optimal assignment $\mathcal{W}^*$, there must exist two edges E_1 and E_2 such that $layer(E_1) = layer(E_2) = l$, $E_1 \in Ans(E_2)$, and $w_{E_1} < w_{E_2}$ (see Figure 7(a)). Note that $q^r(E_1) = q^r(E_2) = q_l^r$ and $q^c(E_1) = q^c(E_2) = q_l^c$. Since $\mathcal{W}^*$ is optimal, the increase in the cost function when we change the width of E_2 in $\mathcal{W}^*$ from w_{E_2} to w_{E_1} (see Figure 7(b)) is:

$$\begin{aligned}
& \Delta T(\mathcal{W}^* / E_2 : w_{E_2} \rightarrow w_{E_1}) \\
= \; & \mathcal{K}_3 \cdot l_{E_2} \cdot q_l^c \cdot (w_{E_1} - w_{E_2}) + \mathcal{K}_4 \cdot \sum_{E' \in Ans(E_2)} l_{E_2} \cdot l_{E'} \cdot F(E', E_2) \cdot q_l^c \cdot q_l^r \cdot \frac{w_{E_1} - w_{E_2}}{w_{E'}} + \\
& \mathcal{K}_4 \cdot \sum_{E' \in Des(E_2)} l_{E_2} \cdot l_{E'} \cdot F(E_2, E') \cdot q_l^c \cdot q_l^r \cdot \left(\frac{w_{E'}}{w_{E_1}} - \frac{w_{E'}}{w_{E_2}}\right) + \\
& \mathcal{K}_5 \cdot l_{E_2} \cdot G(E_2) \cdot q_l^r \cdot \left(\frac{1}{w_{E_1}} - \frac{1}{w_{E_2}}\right) \\
\geq \; & 0
\end{aligned}$$

Moreover, the increase in the cost function when we change the width of E_1 from w_{E_1} to w_{E_2} (see Figure 7(c)) is:

$$\begin{aligned}
& \Delta T(\mathcal{W}^* / E_1 : w_{E_1} \rightarrow w_{E_2}) \\
= \; & \mathcal{K}_3 \cdot l_{E_1} \cdot q_l^c \cdot (w_{E_2} - w_{E_1}) + \mathcal{K}_4 \cdot \sum_{E' \in Ans(E_1)} l_{E_1} \cdot l'_E \cdot F(E', E_1) \cdot q_l^r \cdot q_l^c \cdot \frac{w_{E_2} - w_{E_1}}{w_{E'}} +
\end{aligned}$$

$$\mathcal{K}_4 \cdot \sum_{E' \in Des(E_1)} l_{E_1} \cdot l_{E'} \cdot F(E_1, E') \cdot q_l^r \cdot q_l^c \cdot (\frac{w_{E'}}{w_{E_2}} - \frac{w_{E'}}{w_{E_1}}) +$$
$$\mathcal{K}_5 \cdot l_{E_1} \cdot G(E_1) \cdot q_l^r \cdot (\frac{1}{w_{E_2}} - \frac{1}{w_{E_1}})$$

If we compare $\frac{\Delta T(W^*/E_2 : w_{E_2} \to w_{E_1})}{l_{E_2}}$ with $-\frac{\Delta T(W^*/E_1 : w_{E_1} \to w_{E_2})}{l_{E_1}}$ term-by-term and use the properties of the F

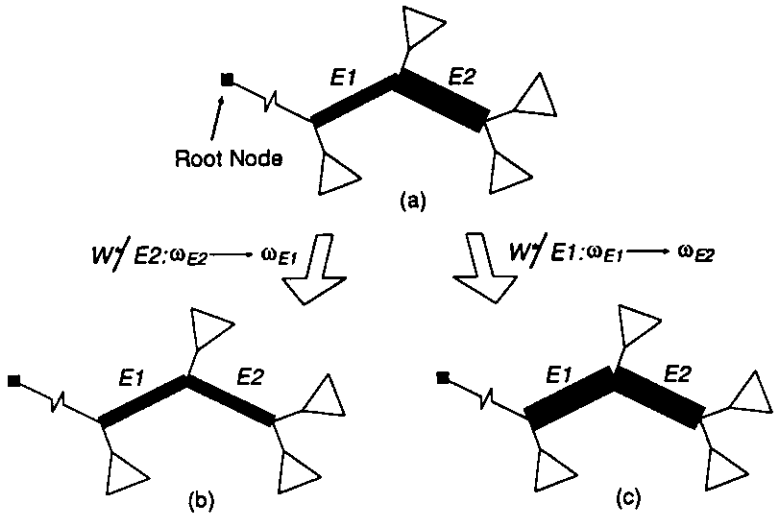

Figure 7: (a) If the monotone property fails for an interconnect tree T, for any optimal assignment W^*, there must exist two edges E_1 and E_2 such that $layer(E_1) = layer(E_2) = l$, $E_1 \in Ans(E_2)$, and $w_{E_1} < w_{E_2}$; (b) The width assignments obtained by replacing the width of E_2 with that of E_1; (b) The width assignments obtained by replacing the width of E_1 with that of E_2.

and G functions in (8), we have:

$$\begin{aligned}
\mathcal{K}_3 \cdot q_l^c \cdot (w_{E_1} - w_{E_2}) &= \mathcal{K}_3 \cdot q_l^c \cdot (w_{E_1} - w_{E_2}) \\
\mathcal{K}_4 \cdot \sum_{E' \in Ans(E_2)} l_{E'} \cdot F(E', E_2) \cdot q_l^r \cdot q_l^c \cdot \frac{w_{E_1} - w_{E_2}}{w_{E'}} &< \mathcal{K}_4 \cdot \sum_{E' \in Ans(E_1)} l_{E'} \cdot F(E', E_1) \cdot q_l^r \cdot q_l^c \cdot \frac{w_{E_1} - w_{E_2}}{w_{E'}} \\
\mathcal{K}_4 \cdot \sum_{E' \in Des(E_2)} l_{E'} \cdot F(E_2, E') \cdot q_l^r \cdot q_l^c \cdot (\frac{w_{E'}}{w_{E_1}} - \frac{w_{E'}}{w_{E_2}}) &< \mathcal{K}_4 \cdot \sum_{E' \in Des(E_1)} l_{E'} \cdot F(E_1, E') \cdot q_l^r \cdot q_l^c \cdot (\frac{w_{E'}}{w_{E_1}} - \frac{w_{E'}}{w_{E_2}}) \\
\mathcal{K}_5 \cdot G(E_2) \cdot q_l^r \cdot (\frac{1}{w_{E_1}} - \frac{1}{w_{E_2}}) &\leq \mathcal{K}_5 \cdot G(E_1) \cdot q_l^r \cdot (\frac{1}{w_{E_1}} - \frac{1}{w_{E_2}})
\end{aligned}$$

Summing up the inequalities, we have:

$$\begin{aligned}
\frac{\Delta T(W^*/E_2 : w_{E_2} \to w_{E_1})}{l_{E_2}} &< -\frac{\Delta T(W^*/E_1 : w_{E_1} \to w_{E_2})}{l_{E_1}} \\
\Rightarrow \Delta T(W^*/E_1 : w_{E_1} \to w_{E_2}) &< -\frac{l_{E_1}}{l_{E_2}} \cdot \Delta T(W^*/E_2 : w_{E_2} \to w_{E_1}) \leq 0
\end{aligned}$$

Therefore, the delay $T(W^*)$ will be reduced if the width of segment E_1 is increased from w_{E_1} to w_{E_2}, which contradicts to the assumption that W^* is already optimal. □

3 Dominance Property

Theorem 3 *Let $\mathcal{W}^*$ be an optimal width assignment for the generalized wiresizing problem. If a width assignment $\mathcal{W}$ dominates $\mathcal{W}^*$, then any local refinement of $\mathcal{W}$ still dominates $\mathcal{W}^*$. Similarly, if a width assignment $\mathcal{W}$ is dominated by $\mathcal{W}^*$, then any local refinement of $\mathcal{W}$ is dominated by $\mathcal{W}^*$.*

Proof: Define

$$\begin{aligned}
\Psi(\mathcal{W},\overline{E}) &= \mathcal{K}_3 \cdot \sum_{E' \in T-\{E\}} l_{E'} \cdot q^c(E') \cdot w_{E'} \; + \\
&\quad \mathcal{K}_4 \cdot \sum_{E',E'' \in T-\{E\}, E' \neq E''} l_{E'} \cdot l_{E''} \cdot q^r(E') \cdot q^c(E'') \cdot F(E',E'') \cdot \frac{w_{E''}}{w_{E'}} \; + \\
&\quad \mathcal{K}_5 \cdot \sum_{E' \in T-\{E\}} l_{E'} \cdot G(E') \cdot q^r(E') \cdot \frac{1}{w_{E'}} \\
\Phi(\mathcal{W},\overline{E}) &= \mathcal{K}_3 \cdot l_E \cdot q^c(E) \; + \; \mathcal{K}_4 \cdot \sum_{E' \in Ans(E)} l_E \cdot l_{E'} \cdot F(E',E) \cdot q^r(E') \cdot q^c(E) \cdot \frac{1}{w_{E'}} \\
\Theta(\mathcal{W},\overline{E}) &= \mathcal{K}_4 \cdot \sum_{E' \in Des(E)} l_E \cdot l_{E'} \cdot F(E,E') \cdot q^r(E) \cdot q^c(E') \cdot w_{E'} \; + \; \mathcal{K}_5 \cdot l_E \cdot G(E) \cdot q^r(E)
\end{aligned}$$

We can then rewrite (11) as follows:

$$T(\mathcal{W}) = \Psi(\mathcal{W},\overline{E}) + \Phi(\mathcal{W},\overline{E}) \cdot w_E + \Theta(\mathcal{W},\overline{E}) \cdot \frac{1}{w_E} \tag{13}$$

Since $\Psi(\mathcal{W},\overline{E})$, $\Phi(\mathcal{W},\overline{E})$, and $\Theta(\mathcal{W},\overline{E})$ are independent of w_E, they are considered as constant for local refinement of E. Minimizing (13) gives the local refinement $\tilde{w_E}$ of segment E. Notice that for any pair of wiresizing solutions $\mathcal{W}$ and $\mathcal{W}'$ such that $\mathcal{W}'$ dominates $\mathcal{W}$. We have $\Phi(\mathcal{W},\overline{E}) \geq \Phi(\mathcal{W}',\overline{E})$ and $\Theta(\mathcal{W},\overline{E}) \leq \Theta(\mathcal{W}',\overline{E})$.

Let w_E^* be the width for segment E in the optimal assignment $\mathcal{W}^*$. We have:

$$\Psi(\mathcal{W},\overline{E}) + \Phi(\mathcal{W},\overline{E}) \cdot \tilde{w_E} + \Theta(\mathcal{W},\overline{E}) \cdot \frac{1}{\tilde{w_E}} \leq \Psi(\mathcal{W},\overline{E}) + \Phi(\mathcal{W},\overline{E}) \cdot w_E^* + \Theta(\mathcal{W},\overline{E}) \cdot \frac{1}{w_E^*} \tag{14}$$

Since w_E^* is also the *locally* optimal width assignment for edge E with respect to the rest of the width assignment in $\mathcal{W}^*$, we have:

$$\Psi(\mathcal{W}^*,\overline{E}) + \Phi(\mathcal{W}^*,\overline{E}) \cdot w_E^* + \Theta(\mathcal{W}^*,\overline{E}) \cdot \frac{1}{w_E^*} \leq \Psi(\mathcal{W}^*,\overline{E}) + \Phi(\mathcal{W}^*,\overline{E}) \cdot \tilde{w_E} + \Theta(\mathcal{W}^*,\overline{E}) \cdot \frac{1}{\tilde{w_E}} \tag{15}$$

Summing up (14) and (15), we obtain:

$$\begin{aligned}
&\{\Phi(\mathcal{W},\overline{E}) - \Phi(\mathcal{W}^*,\overline{E})\} \cdot \{\tilde{w_E} - w_E^*\} + \{\Theta(\mathcal{W},\overline{E}) - \Theta(\mathcal{W}^*,\overline{E})\} \cdot \left\{\frac{1}{\tilde{w_E}} - \frac{1}{w_E^*}\right\} \leq 0 \\
&\Rightarrow \left\{\Phi(\mathcal{W},\overline{E}) - \Phi(\mathcal{W}^*,\overline{E}) + \frac{\Theta(\mathcal{W}^*,\overline{E}) - \Theta(\mathcal{W},\overline{E})}{\tilde{w_E} \cdot w_E^*}\right\} \cdot \{\tilde{w_E} - w_E^*\} \leq 0
\end{aligned}$$

If $\mathcal{W}$ dominates $\mathcal{W}^*$, we have $\Phi(\mathcal{W},\overline{E}) \leq \Phi(\mathcal{W}^*,\overline{E})$ and $\Theta(\mathcal{W}^*,\overline{E}) \leq \Theta(\mathcal{W},\overline{E})$, and therefore $\tilde{w_E} - w_E^* \geq 0$ (i.e. the refinement of $\mathcal{W}$ still dominates $\mathcal{W}^*$). Similarly, if $\mathcal{W}^*$ dominates $\mathcal{W}$, then $\mathcal{W}^*$ dominates the local refinement of $\mathcal{W}$. □