Optimizing Dominant Time Constant in RC Circuits

Lieven Vandenberghe, Stephen Boyd, and Abbas El Gamal

***Abstract*— Conventional methods for optimal sizing of wires and transistors use linear resistor-capacitor (RC) circuit models and the Elmore delay as a measure of signal delay. If the RC circuit has a tree topology, the sizing problem reduces to a convex optimization problem that can be solved using geometric programming. The tree topology restriction precludes the use of these methods in several sizing problems of significant importance to high-performance deep submicron design, including for example, circuits with loops of resistors, e.g., clock distribution meshes and circuits with coupling capacitors, e.g., buses with crosstalk between the wires. In this paper, we propose a new optimization method that can be used to address these problems.**

The method is based on the dominant time constant as a measure of signal propagation delay in an RC circuit instead of Elmore delay. Using this measure, sizing of any RC circuit can be cast as a convex optimization problem and solved using recently developed efficient interior-point methods for semidefinite programming. The method is applied to three important sizing problems: clock mesh sizing and topology design, sizing of tristate buses, and sizing of bus line widths and spacings taking crosstalk into account.

***Index Terms*— Circuit optimization, circuit topology, clocks, crosstalk, delay effects, dominant time constant, integrated circuit interconnections, nonlinear programming, RC circuits, semidefinite programming.**

I. Introduction

THE CLASSICAL approach to optimal sizing of wires and transistors assumes a linear resistor-capacitor (RC) circuit model and uses *Elmore delay* as a measure of signal propagation delay. This approach finds its origins in [1]–[3]. In particular, Fishburn and Dunlop [3] were first to observe that if the resistors form a tree with the input voltage source at its root and all capacitors are grounded, the Elmore delay of a RC circuit is a *posynomial function* of the conductances and capacitances. This observation has the important consequence that convex programming, specifically *geometric programming*, can be used to optimize Elmore delay subject to area and power constraints. Geometric programming forms the basis of the TILOS program and of several extensions and related programs developed since then [3]–[8]. See [9] also for a comprehensive recent survey.

Manuscript received December 18, 1996. This work was supported in part by AFOSR under F49620-95-1-0318, NSF under ECS-9222391 and EEC-9420565, MURI under F49620-95-1-0525, and a gift from Synopsys, Inc. This paper was recommended by Associate Editor K. Mayaram.

L. Vandenberghe is with the Electrical Engineering Department, University of California, Los Angeles, CA 90095 USA (e-mail: vandenbe@ee.ucla.edu).

S. Boyd and A. El Gamal are with the Information Systems Laboratory, Electrical Engineering Department, Stanford University, Stanford, CA 94305 USA (e-mail: boyd@isl.stanford.edu; abbas@isl.stanford.edu).

Publisher Item Identifier S 0278-0070(98)02545-7.

The tree topology restriction, however, precludes the use of these Elmore delay methods in several sizing problems of significant importance to high performance deep submicron design including circuits with capacitive coupling between the nodes, e.g., buses with crosstalk and circuits with loops of resistors, e.g., clock meshes. In this paper, we present a new optimal sizing method that can be used to address these problems. The method uses the *dominant time constant* as a measure of signal delay instead of Elmore delay. The motivation for this choice is that the dominant time constant of a general RC circuit is a quasi-convex function of the conductances and capacitances. In particular, we show that it can be optimized using recently developed efficient methods for *semidefinite programming*. The Elmore delay, on the other hand, has no useful convexity properties except when the RC circuit has a tree topology.

We apply our method to three important sizing problems. The first is the problem of sizing a clock mesh (Section V). This problem is difficult to handle using Elmore delay methods because of the presence of resistor loops. The results also illustrate that, to a certain extent, our method can be used to design the interconnect *topology* (in addition to sizing). The second problem we consider is the sizing of a tristate bus (Section VI). The interconnect network in this example is driven by multiple sources and therefore it does not have the tree topology required by the Elmore delay methods. The third problem is the simultaneous sizing of bus line widths and spacings taking into account coupling capacitances between neighboring bus wires, i.e., crosstalk (Section VII). This problem is particularly important in deep submicron design where the coupling capacitance can be significantly higher than the plate capacitance. The results illustrate that optimizing dominant time constant allows us to control not only the signal propagation delay, but also indirectly the crosstalk between the wires. Since the circuit has nongrounded capacitors, this is not possible using Elmore delay.

The outline of the paper is as follows. In Section II, we describe the RC circuit model considered in the paper. In Section III, we discuss three definitions of signal propagation delay and define the dominant time constant. In Section IV, we show that sizing problems using the dominant time constant as a measure of delay lead to semidefinite programming problems for which efficient methods have recently been developed. In Sections V–VII, we describe the three applications. We conclude in Section VIII with a discussion of the computational complexity of the method and with an overview of its advantages and disadvantages compared to Elmore delay methods. An extended version of this paper containing additional examples and mathematical background is available as an internal report [22].

0278–0070/98$10.00 © 1998 IEEE

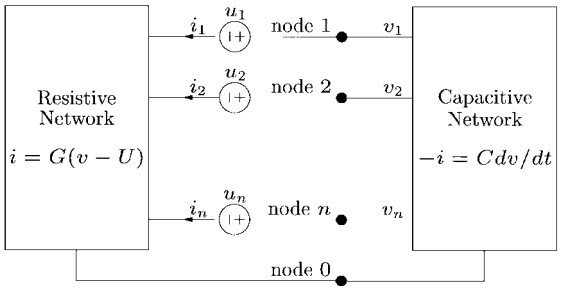

Fig. 1. General RC circuit with $n + 1$ nodes shown as a resistive network, a capacitive network, and voltages sources.

II. Circuit Models

A. General RC Circuit

We consider linear RC circuits that can be described by the differential equation

$$C\frac{dv}{dt} = -G(v(t) - u(t)) \tag{1}$$

where $v(t) \in \boldsymbol{R}^n$ is the vector of node voltages, $u(t) \in \boldsymbol{R}^n$ is the vector of independent voltage sources, $C \in \boldsymbol{R}^{n\times n}$ is the capacitance matrix, and $G \in \boldsymbol{R}^{n\times n}$ is the conductance matrix (see Fig. 1). Throughout the paper we assume that C and G are symmetric and positive definite (i.e., that the capacitive and resistive subcircuits are reciprocal and strictly passive). The case in which C and G are only positive semidefinite, i.e., possibly singular, is considered in Appendix A.

We are interested in design problems in which C and G depend on some design parameters $x \in \boldsymbol{R}^m$. Specifically we assume that the matrices C and G are *affine* functions of x, i.e.,

$$\begin{aligned} C(x) &= C_0 + x_1C_1 + \cdots + x_mC_m \\ G(x) &= G_0 + x_1G_1 + \cdots + x_mG_m \end{aligned} \tag{2}$$

where C_i and G_i are symmetric matrices.

We will refer to a circuit described by (1) and (2) as a *general RC circuit*. We will also consider several important special cases, for example, circuits composed of two-terminal elements, circuits in which the resistive network forms a tree, or all capacitors are grounded. We describe these special cases now.

B. RC Circuit

When the general RC circuit is composed of two-terminal resistors and capacitors (and the independent voltage sources) we will refer to it as an *RC circuit*. More precisely, consider a circuit with N branches and $n+1$ nodes, numbered 0 to n where node 0 is the ground or reference node. Each branch k consists of a capacitor $c_k \geq 0$ and a conductance $g_k \geq 0$ in series with a voltage source U_k (see Fig. 2). Some branches can have a zero capacitance or a zero conductance, but we will assume that both the capacitive subnetwork (i.e., the network obtained by removing all resistors and voltage sources) and the resistive subnetwork (i.e., the network obtained by removing all capacitors) are connected.

We denote the vector of node voltages by $v \in \boldsymbol{R}^n$, the vector of branch voltages by $V \in \boldsymbol{R}^N$, and the vector of branch currents by $I \in \boldsymbol{R}^N$. The relation between branch voltages and currents is

$$I_k = c_k\frac{dV_k}{dt} + g_k(V_k - U_k), \qquad k = 1, \cdots, N. \tag{3}$$

To obtain a description of the form (1), we introduce the reduced node-incidence matrix $A \in \boldsymbol{R}^{n\times N}$ and define C and G as

$$C = A\,\mathrm{diag}(c)A^T, \quad G = A\,\mathrm{diag}(g)A^T. \tag{4}$$

Obviously, C and G are positive semidefinite. Both matrices are nonsingular if the capacitive and resistive subnetworks are connected. It can also be shown that the matrices $R = G^{-1}$ and C^{-1} are elementwise nonnegative.

Using Kirchhoff's laws $AI = 0$ and $V = A^Tv$, it is straightforward to write the branch equations (3) as (1) with $u = G^{-1}A\ \mathrm{diag}(g)U$.

From the expressions for the matrices G and C (4), we see that they are affine functions of the design parameters x if each of the conductances g_k and capacitances c_k is.

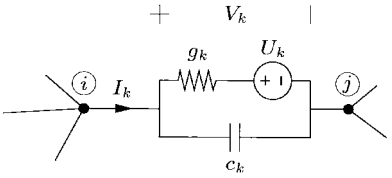

Fig. 2. Orientation of the kth branch voltage V_k and branch current I_k in an RC circuit. Each branch consists of a capacitor $c_k \geq 0$ and a resistor with conductance $g_k \geq 0$ in series with an independent voltage source U_k.

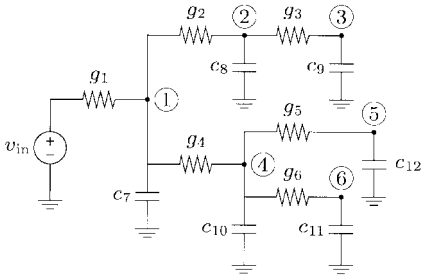

Fig. 3. Example of a grounded capacitor RC tree.

C. Grounded Capacitor RC Circuit

It is quite common that all capacitors in the RC circuit are connected to the ground node. In this case, the matrix C is diagonal and nonsingular if there is a capacitor between every node and the ground. We will refer to circuits of this form as *grounded capacitor RC circuits*.

D. Grounded Capacitor RC Tree

The most restricted class of circuits considered in this paper consists of grounded capacitor RC circuits in which the resistive branches form a tree with the ground node as its root. Moreover, only one resistive branch is connected to the ground node and it contains the only voltage source in the circuit. An example is shown in Fig. 3.

Note that the resistance matrix $R = G^{-1}$ for a circuit of this class can be written down by inspection

$$R_{ij} = \sum \text{ resistances upstream from node } i \text{ and node } j \tag{5}$$

i.e., to find R_{ij} we add all resistances in the intersection of the unique path from node i to the root of the tree and the unique path from node j to the root of the tree. For the example in Fig. 3, we obtain that R is equal to (see the equation located at the bottom of this page) where $r_i = 1/g_i$.

One can also verify that in a grounded capacitor RC tree with input voltage $v_{\text{in}}(t)$, the vector $u(t)$ in (1) is equal to $u(t) = v_{\text{in}}(t)\mathbf{1}$ where $\mathbf{1}$ is the vector with all components equal to one.

E. Applications

Linear RC circuits are often used as approximate models for transistors and interconnect wires. When the design parameters are the physical widths of conductors or transistors, the conductance and capacitance matrices are affine in these parameters, i.e., they have the form (2).

An important example is wire sizing, where x_i denotes the width of a segment of some conductor or interconnect line. A simple lumped model of the segment consists of a π section: a series conductance with a capacitance to ground on each end. Here the conductance is linear in the width x_i and the capacitances are linear or affine. We can also model each segment by many such π sections and still have the general form (1), (2).

Another important example is an MOS transistor circuit where x_i denotes the width of a transistor. When the transistor is "on" it is modeled as a conductance that is proportional to x_i and a source-to-ground capacitance and drain-to-ground capacitance that are linear or affine in x_i.

III. DELAY

We are interested in how fast a change in the input u propagates to the different nodes of the circuit and in how this propagation delay varies as a function of the resistances and capacitances. In this section, we introduce three possible measures for this propagation delay: the threshold delay, which is the most natural measure but difficult to handle mathematically; the Elmore delay, which is widely used in transistor and wire sizing; and the dominant time constant. We will compare the three delay measures in the examples (Sections V–VI) where we will observe that their numerical values are usually quite close. More theoretical details on the relation between these three measures will be presented in Appendix B, including some bounds that they must satisfy.

We assume that for $t<0$, the circuit is in static steady state with $u(t) = v(t) = v_-$. For $t \geq 0$, the source switches to the constant value $u(t) = v_+$. As a result, for $t \geq 0$ we have

$$v(t) = v_+ + e^{-C^{-1}Gt}(v_- - v_+) \tag{6}$$

which converges as $t \to \infty$ to v_+ (since our assumption $C>0, G>0$ implies stability). The difference between the node voltage and its ultimate value is given by

$$\tilde{v}(t) = e^{-C^{-1}Gt}(v_- - v_+)$$

and we are interested in how large t must be before this is small.

To simplify the notation, we will relabel $\tilde{v}$ as v and from here on study the rate at which

$$v(t) = e^{-C^{-1}Gt}v(0) \tag{7}$$

becomes small. Note that this v satisfies the autonomous equation $Cdv/dt = -Gv$.

It can be shown that for a grounded capacitor RC circuit the matrix $e^{-C^{-1}Gt}$ is elementwise nonnegative for all $t \geq 0$ (see [23, p. 146]). Therefore, if $v(0) \geq 0$ (meaning, $v_k(0) \geq 0$ for $k = 1, \cdots, n$) in (7), the voltages remain nonnegative, i.e., for $t \geq 0$ we have

$$v(t) \geq 0.$$

Also note that in a grounded capacitor RC tree, the steady-state node voltages are all equal. When discussing RC trees, we will therefore assume without loss of generality that the input switches from zero to one at $t = 0$, i.e., $v_- = 0, v_+ = \mathbf{1}$ in (6), or for the autonomous model, that $v(0) = \mathbf{1}$ in (7).

A. Threshold Delay

In many applications the natural measure of the delay at node k is the first time after which v_k stays below some given threshold level $\alpha > 0$, i.e.,

$$T_k^{\text{thres}} = \inf\{T \mid |v_k(t)| \leq \alpha \text{ for } t \geq T\}.$$

We will call the maximum threshold delay to any node the *critical threshold delay* of the circuit

$$\begin{aligned} T^{\text{thres}} &= \max\{T_1^{\text{thres}}, \cdots, T_n^{\text{thres}}\} \\ &= \inf\{T \mid \|v(t)\|_\infty \leq \alpha \text{ for } t \geq T\} \end{aligned}$$

where $\|\cdot\|_\infty$ denotes the infinity norm, defined by $\|z\|_\infty = \max_i |z_i|$. The critical threshold delay is the first time after which all node voltages are less than α.

$$\begin{bmatrix} r_1 & r_1 & r_1 & r_1 & r_1 & r_1 \\ r_1 & r_1+r_2 & r_1+r_2 & r_1 & r_1 & r_1 \\ r_1 & r_1+r_2 & r_1+r_2+r_3 & r_1 & r_1 & r_1 \\ r_1 & r_1 & r_1 & r_1+r_4 & r_1+r_4 & r_1+r_4 \\ r_1 & r_1 & r_1 & r_1+r_4 & r_1+r_4+r_5 & r_1+r_4 \\ r_1 & r_1 & r_1 & r_1+r_4 & r_1+r_4 & r_1+r_4+r_6 \end{bmatrix}$$

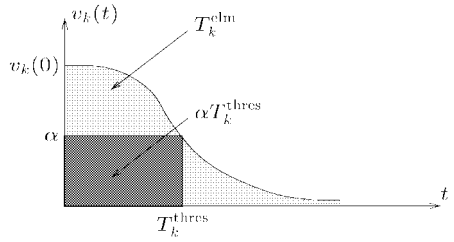

Fig. 4. Graphical interpretation of the Elmore delay at node k. $T_k^{\rm thres}$ is the threshold delay at node k. The area below v_k, which is shaded lightly, is $T_k^{\rm elm}$. The darker shaded box, which lies below v_k, has area $\alpha T_k^{\rm thres}$. From this it is clear that when the voltage is nonnegative and monotonically decaying $\alpha T_k^{\rm thres} \leq T_k^{\rm elm}$.

The critical threshold delay $T^{\rm thres}$ depends on the design parameters x through (7), i.e., in a very complicated way. Methods for direct optimization of $T^{\rm thres}$ are inefficient and also local, i.e., not guaranteed to find a globally optimal design.

B. Elmore Delay

In [1], Elmore introduced a measure of the delay to a node that depends on C and G (hence, x) in a simpler way than the threshold delay and often gives an acceptable approximation to it. The Elmore delay to node k is defined as

$$T_k^{\rm elm} = \int_0^\infty v_k(t)\, dt.$$

While $T_k^{\rm elm}$ is always defined, it can be interpreted as a measure of delay only when $v_k(t) \geq 0$ for all $t \geq 0$, i.e., when the node voltage is nonnegative. (Which is the case, as we mentioned, in grounded capacitor RC circuits with $v(0) \geq 0$.)

In the common case that the voltages decay monotonically, i.e., $dv_k(t)/dt \leq 0$ for all $t \geq 0$, we have the simple bound

$$\alpha T_k^{\rm thres} \leq T_k^{\rm elm}$$

which can be derived as follows. Assuming v_k is positive and nonincreasing, we must have $v_k(t) > \alpha$ for $t < T_k^{\rm thres}$. Hence the integral of v_k must exceed $\alpha T_k^{\rm thres}$ (see Fig. 4). The monotonic decay property holds, for example, for grounded capacitor RC trees [2, Appendix C]. (See also [24] for sharper bounds between the threshold delay and the Elmore delay in grounded capacitor RC trees.)

We can express the Elmore delay in terms of G, C, and $v(0)$ as

$$T_k^{\rm elm} = e_k^T G^{-1} C v(0)$$

where e_k is the kth unit vector. Thus the vector of Elmore delays is given by the simple expression $RCv(0)$ where $R = G^{-1}$ is the resistance matrix. We define the *critical Elmore delay* as the largest Elmore delay at any node, i.e., $T^{\rm elm} = \max_k\ T_k^{\rm elm}$.

For a grounded capacitor RC circuit with $v(0) \geq 0$, we can express the critical Elmore delay as

$$T^{\rm elm} = \|G^{-1} C v(0)\|_\infty$$

by noting that the matrix $G^{-1}C = RC$ is elementwise nonnegative. If $v(0) = \mathbf{1}$ (as in a grounded-capacitor RC tree) we can also write

$$T^{\rm elm} = \max_k (e_k^T G^{-1} C e) = \|G^{-1}C\|_\infty. \tag{8}$$

(For a matrix $A \in \boldsymbol{R}^{n\times n}$, $\|A\|_\infty$ is the maximum row sum of A, i.e., $\|A\|_\infty = \max_{i=1,\cdots,n}\ \Sigma_{j=1}^n\ |A_{ij}|$.)

C. Dominant Time Constant

In this paper, we propose using the *dominant time constant* of the RC circuit as a measure of the delay. We start with the definition. Let $\lambda_1, \cdots, \lambda_n$ denote the eigenvalues of the circuit, i.e., the eigenvalues of $-C^{-1}G$ or equivalently, the roots of the characteristic polynomial $\det(sC+G)$. They are real and negative since they are also the eigenvalues of the symmetric, negative definite matrix

$$C^{1/2}(-C^{-1}G)C^{-1/2} = -C^{-1/2}GC^{-1/2}$$

(which is similar to $-C^{-1}G$). We assume they are sorted in decreasing order, i.e.,

$$0 > \lambda_1 \geq \cdots \geq \lambda_n.$$

The largest eigenvalue λ_1 is called the *dominant eigenvalue* or *dominant pole* of the RC circuit.

Each node voltage can be expressed in the form

$$v_k(t) = \sum_{i=1}^n \alpha_{ik} e^{\lambda_i t} \tag{9}$$

which is a sum of decaying exponentials with rates given by the eigenvalues. We define the *dominant time constant* at the kth node as follows. Let p denote the index of the first nonzero term in the sum (9), i.e., $\alpha_{ik} = 0$ for $i < p$ and $\alpha_{ip} \neq 0$. (Thus, the slowest decaying term in v_k is $\alpha_{ip} e^{\lambda_p t}$.) We call λ_p the *dominant eigenvalue* at node k, and the dominant time constant at node k is defined as

$$T_k^{\rm dom} = -1/\lambda_p.$$

In most cases, v_k contains a term associated with the largest eigenvalue λ_1 in which case we simply have $T_k^{\rm dom} = -1/\lambda_1$.

The dominant time constant $T_k^{\rm dom}$ measures the asymptotic rate of decay of $v_k(t)$ and there are several ways to interpret it. For example, $T_k^{\rm dom}$ is the smallest number T such that

$$|v_k(t)| \leq \beta e^{-t/T}$$

holds for some β and all $t \geq 0$.

The *(critical) dominant time constant* is defined as $T^{\rm dom} = \max_k\ T_k^{\rm dom}$. Except in the pathological case when $v(0)$ is deficient in the eigenvector associated with λ_1, we have

$$T^{\rm dom} = -1/\lambda_1. \tag{10}$$

In the sequel we will assume this is the case. Note that the dominant time constant $T^{\rm dom}$ is a very complicated function of G and C, i.e., the negative inverse of the largest zero of the polynomial $\det(sC+G)$.

The dominant time constant can also be expressed in another form that will be more useful to us.

$$T^{\rm dom} = \min\{T | TG - C \geq 0\}. \tag{11}$$

This form has another advantage: it provides a reasonable measure of delay in the case when C and G are only positive semidefinite (i.e., possibly singular). The details are given in Appendix A.

IV. Dominant Time Constant Optimization

In this section, we show how several important design problems involving dominant time constant, area, and power can be cast as convex or quasi-convex optimization problems that can be solved efficiently.

A. Dominant Time Constant Specification as Linear Matrix Inequality

As a consequence of (11), we have

$$T^{\mathrm{dom}}(x) \le T_{\max} \Longleftrightarrow T_{\max}G(x) - C(x) \ge 0. \tag{12}$$

This type of constraint is called a *linear matrix inequality* (LMI): the left-hand side is a symmetric matrix, the entries of which are affine functions of x. It can be shown that the set of vectors x that satisfy (12) is convex regardless of the topology of the circuit. In other words, an upper bound on the dominant time constant is a convex constraint in the variables x.

This means that T^{dom} is a *quasi-convex* function of x, i.e., the sublevel sets

$$\{x | T^{\mathrm{dom}}(x) \le T_{\max}\}$$

are convex sets for all $T_{\max}$. Quasiconvexity can also be expressed as: for $\theta \in [0,1]$

$$T^{\mathrm{dom}}(\theta x + (1-\theta)\tilde{x}) \le \max\{T^{\mathrm{dom}}(x), T^{\mathrm{dom}}(\tilde{x})\}$$

i.e., as the design parameters vary on a segment between two values, the dominant time constant is never any more than the larger of the two dominant time constants at the endpoints.

Linear matrix inequalities have recently been recognized as an efficient and unified representation of a wide variety of nonlinear convex constraints. They arise in many different fields such as control theory and combinatorial optimization (for surveys, see [26]–[30]). Most importantly for us, many convex and quasi-convex optimization problems that involve LMI's can be solved with great efficiency using recently developed interior-point methods.

B. Semidefinite Programming

The most common optimization problem involving LMI's is the *semidefinite programming* problem (SDP) in which we minimize a linear function subject to a linear matrix inequality

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & A(x) \ge 0 \end{aligned} \tag{13}$$

where $A(x) = A_0 + x_1 A_1 + \cdots + x_m A_m, A_i = A_i^T$, and the inequality means $A(x)$ is positive semidefinite. Semidefinite programs are convex optimization problems and can be solved very efficiently (see, e.g., [27]–[29]). We can also handle multiple LMI constraints in the SDP (13) by representing them as one big block diagonal matrix.

As an example, suppose the area of the circuit described by (2) is an affine function of the variables x_i. This occurs when the variables represent the widths of transistors or conductors (with lengths fixed as l_i) in which case the circuit area has the form

$$a_0 + x_1 l_1 + \cdots + x_m l_m$$

where a_0 is the area of the fixed part of the circuit. We can minimize the area subject to a bound on the dominant time constant $T^{\mathrm{dom}} \le T_{\max}$ and subject to upper and lower bounds on the widths by solving the SDP

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{m} l_i x_i \\ \text{subject to} \quad & T_{\max}G(x) - C(x) \ge 0 \\ & x_{\min} \le x_i \le x_{\max}, \qquad i = 1, \cdots, m. \end{aligned} \tag{14}$$

The solutions of (14) are on the globally optimal tradeoff curve, i.e., they are *Pareto optimal* for area and dominant time constant. By solving this SDP for a sequence of values of $T_{\max}$, we can compute the exact globally optimal tradeoff between area and dominant time constant.

In Section V we will see that tradeoffs between power dissipation and dominant time constant can be computed in a similar way by solving a series of SDP's.

C. Generalized Eigenvalue Minimization

Another common problem involving LMI's has the form

$$\begin{aligned} \text{minimize} \quad & \lambda \\ \text{subject to} \quad & \lambda B(x) - A(x) \ge 0 \\ & B(x) > 0, \quad C(x) \ge 0 \end{aligned} \tag{15}$$

where A, B, and C are symmetric matrices that are affine functions of x and the variables are x and $\lambda \in \boldsymbol{R}$. This problem is called the *generalized eigenvalue minimization problem* (GEVP). GEVP's are quasiconvex and can be solved efficiently. See [25], [31], [27], [32], and [33] for details on specialized algorithms.

As an example, the problem of minimizing the dominant time constant, subject to an upper bound on the area and upper and lower bounds on the variables, can be cast as a GEVP

$$\begin{aligned} \text{minimize} \quad & T \\ \text{subject to} \quad & TG(x) - C(x) \ge 0 \\ & \sum_{i=1}^{m} l_i x_i \le c \\ & x_{\min} \le x_i \le x_{\max}, \qquad i = 1, \cdots, m \end{aligned}$$

with variables T and x.

V. Sizing of Clock Meshes

The possibility of optimizing RC circuits with loops of resistors is of importance to high-performance microprocessor design where the clock signal is distributed using a mesh instead of a tree. In [34], Desai, Cvijetic, and Jensen describe the design of the clock distribution network on a DEC-alpha processor and note, "there is a need for algorithms for sizing large nontree networks." Minimizing the dominant time

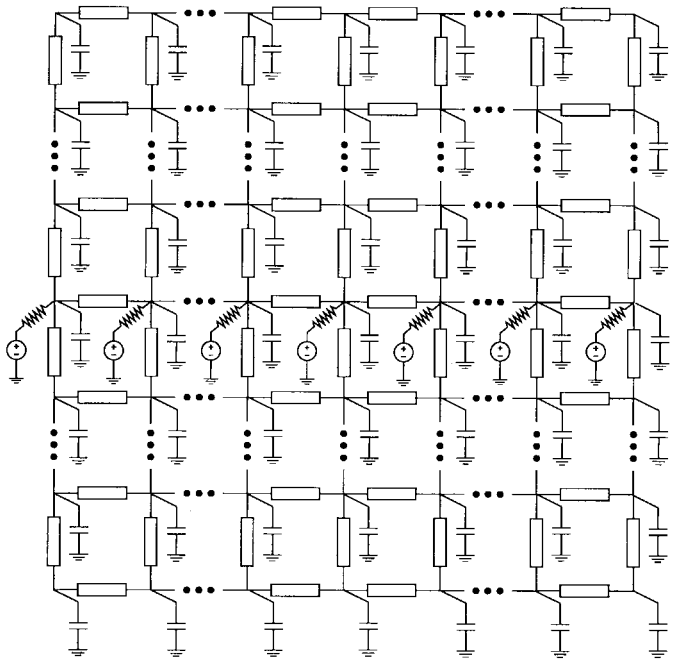

Fig. 5. Clock distribution network modeled as an RC mesh. Each rectangular element represents a wire, which we model as a single π-segment as in Fig. 6. The drivers switch simultaneously. We are interested in the tradeoff between delay (dominant time constant) and total dissipated power. The variables are the widths of the N^2 segments, where N is the number of segments in each column and row.

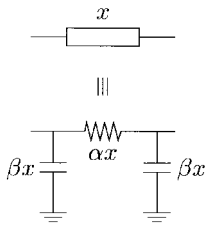

Fig. 6. A segment of an interconnect wire with width x is modeled as a conductance αx and two capacitances to the ground βx.

constant instead of Elmore delay is a promising technique to achieve exactly that goal.

Fig. 5 shows the example that we consider. The circuit consists of a mesh of interconnect wire, with N segments per row and column (so the number of nodes in the circuit is equal to $n = (N+1)^2$). Each interconnect segment (the rectangular elements in Fig. 5) is modeled as a π-segment, as in Fig. 6. Each node of the mesh has a capacitive load C_i. The network is driven by voltage sources with output conductance G_0. Note that this circuit is a grounded capacitor RC circuit, but not an RC tree, since it contains loops of resistors and multiple sources. The optimization variables are the N^2 segment widths x_i (with constraints $0 \leq x_i \leq w_{\max}$).

The sources switch between zero and one simultaneously and at a fixed frequency. Therefore the energy dissipated in once cycle is equal to

$$\mathbf{1}^T C(x)\mathbf{1} = \sum_{i=1}^{n} C_i + 2\beta \sum_{i=1}^{N^2} x_i$$

which is a linear function of the variables x. This means we can minimize the dissipated power subject to a bound on the

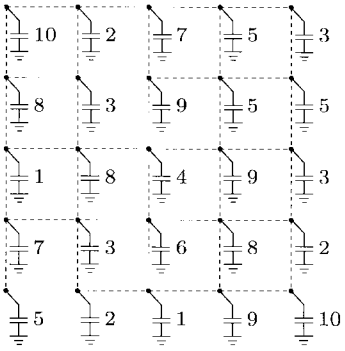

Fig. 7. RC mesh example with 4×4 segments and the numerical values of the load capacitances used in the calculation.

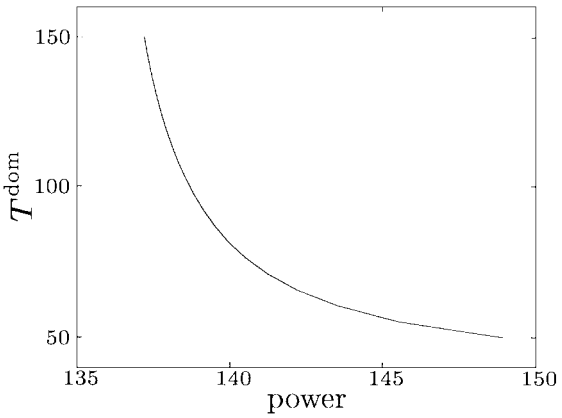

Fig. 8. Tradeoff between dissipated power and dominant time constant.

dominant time constant by solving the SDP

$$\begin{array}{ll} \text{minimize} & \mathbf{1}^T C(x)\mathbf{1} \\ \text{subject to} & T_{\max} G(x) - C(x) \geq 0 \\ & 0 \leq x_i \leq w_{\max}, \qquad i = 1, \cdots, N^2. \end{array} \quad (16)$$

By solving this problem for different values of $T_{\max}$, we can trace the exact globally optimal tradeoff curve between dissipated power and dominant time constant. This tradeoff curve is shown in Fig. 8 for the numerical values

$$N = 4, \quad G_0 = 1, \quad \alpha = 1, \quad \beta = 0.5, \quad w_{\max} = 1$$

and for load capacitances as indicated in Fig. 7. Fig. 9 shows the solution for two different points on the tradeoff curve ($T^{\mathrm{dom}} = 55$ and $T^{\mathrm{dom}} = 100$).

We note that the topology is different in the two cases. More segments are used in the circuit on the left, which has a small dominant time constant and large power consumption (large total capacitance). In the solution on the right, fewer segments are used and they are smaller which reduces the power dissipation but increases the dominant time constant.

Fig. 10 shows the fastest and the slowest step responses in both circuits when a step input is applied simultaneously to the five voltage sources in the middle row. Note in particular that the values of the three delay measures are very close (and in fact, the dominant time constant approximates the 50%-threshold delay better than the Elmore delay). This observation is confirmed by many other examples (see the following sections and the report [22]).

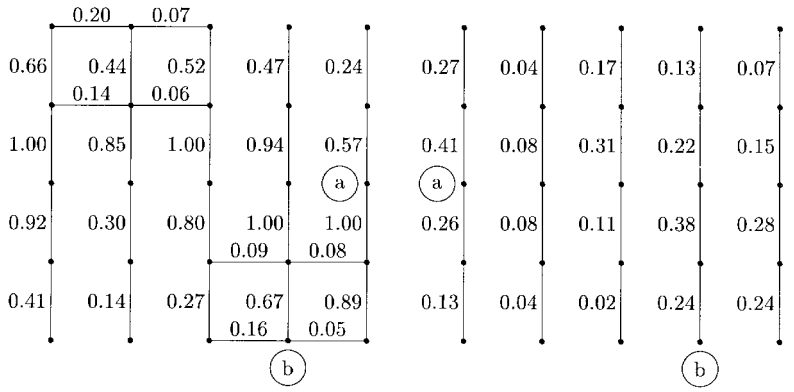

Fig. 9. Optimal solution for two points on the tradeoff curve: $T^{\mathrm{dom}} = 50$ (left) and $T^{\mathrm{dom}} = 100$ (right). The numbers indicate the optimal widths x_i of the segments; the segments with width $x_i = 0$ are not shown.

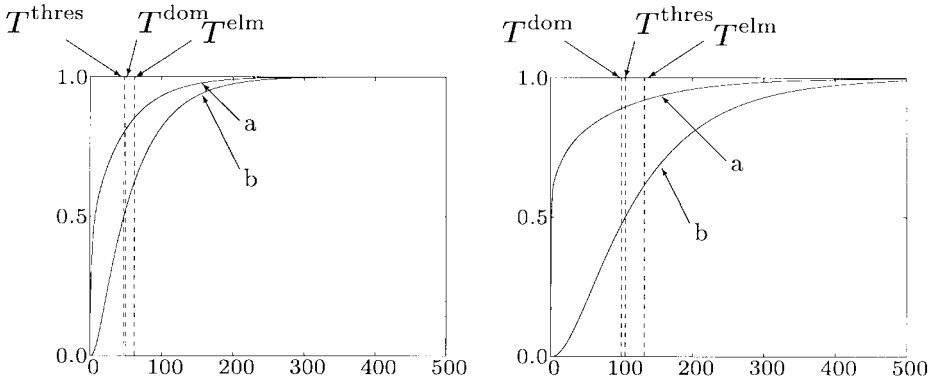

Fig. 10. Step responses for the two solutions in Fig. 9. The plots show the responses at the fastest (a) and the slowest (b) node in Fig. 9. We also show the critical 50%-threshold delay, the critical Elmore delay, and the dominant time constant.

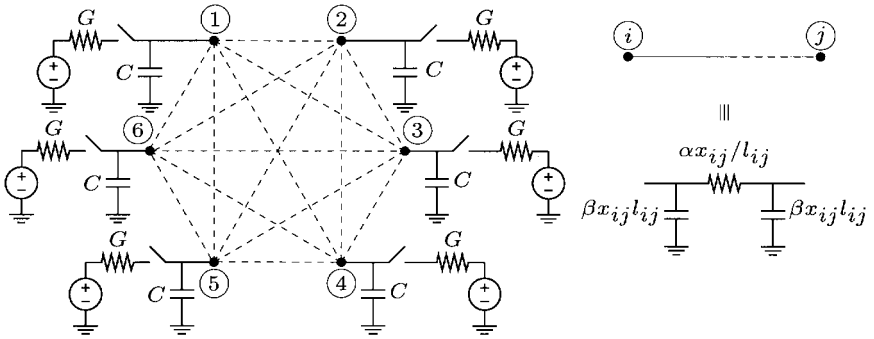

Fig. 11. Tristate bus sizing and topology design. The circuit on the left represents a tristate bus connecting six nodes. Each pair of nodes is connected through a wire, shown as a dashed line, modeled as a π segment shown at right. The bus can be driven from any node. When node i drives the bus, the ith switch is closed and the others are all open. Note that we have 15 wires connecting the nodes, whereas only *five* are needed to connect them. In this example, as in the previous example, we will use dominant time constant optimization to determine the topology of the bus as well as the optimal wire widths x_{ij}: optimal x_{ij}'s which are zero correspond to unused wires.

VI. Tri-State Bus Sizing and Topology Design

In this example we optimize a tristate bus connecting six nodes. The example will again illustrate that dominant time constant minimization can be used to (indirectly) design the optimal topology of a circuit.

The model for the bus is shown in Fig. 11. Each pair of nodes is connected by a wire (shown as a dashed line) which is modeled as a π-segment, as shown on the right in the figure. (Since in the optimal designs many of the wire segments will have width zero, it is perhaps better to think of the fifteen segments as *possible* wire segments.) The capacitance and the conductance of the wire segment between node i and node j depend on its physical dimensions, i.e., on its length l_{ij} and width x_{ij}: the conductance is proportional to x_{ij}/l_{ij}; the capacitance is proportional to $x_{ij}l_{ij}$. The lengths of the wires are given; the widths will be our design variables. The total wire area is $\Sigma_{i>j}\ l_{ij}x_{ij}$.

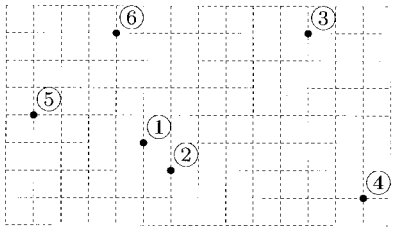

Fig. 12. Position of the six nodes. The length l_{ij} of the wire between each two nodes i and j in Fig. 11 is the ℓ_1-distance (Manhattan-distance) between the points i and j in this figure. The squares in the grid have unit size.

The bus can be driven from any node. When node i drives the bus, the ith switch is closed and the others are all open. Thus we really have six different circuits, each corresponding to a given node driving the bus. To constrain the dominant time constant, we require that the dominant time constant of each of the six drive configuration circuits has dominant time constant less than $T^{\max}$. In other words, if $T_i^{\rm dom}$ is the dominant time constant of the RC circuit obtained by closing the switch at node i and opening the other switches, then $T^{\rm dom} = \max_i T_i$ is the measure of dominant time constant for the tristate bus.

The numerical values used in the calculation are

$$G = 1, \quad C = 10, \quad \beta = 0.5, \quad \alpha = 1.$$

The wire widths are limited to a maximum value of 1.0. We assume that the geometry of the bus is as in Fig. 12 and that the length l_{ij} of the wire between nodes i and j is given by the ℓ_1-distance (Manhattan distance) between points i and j in Fig. 12.

Fig. 13 shows the tradeoff curve between maximum dominant time constant $T^{\rm dom}$ and the bus area. This tradeoff curve was computed by solving the following SDP for a sequence of values of $T_{\max}$

$$\begin{aligned} \text{minimize} \quad & \sum_{i>j} l_{ij} x_{ij} \\ \text{subject to} \quad & 0 \le x_{ij} \le 1 \\ & T_{\max}(\tilde{G}(x) + GE_{kk}) - C(x) \ge 0 \\ & \qquad k = 1, \cdots, 6. \end{aligned}$$

Here x denotes the vector with components x_{ij} (in any indexing order), $\tilde{G}(x)$ denotes the conductance matrix of the circuit when all switches are open, and $C(x)$ is the diagonal matrix with its ith element the total capacitance at node i. The matrix E_{kk} is zero except for the kth diagonal element, which is equal to one. The six different LMI constraints in the above SDP correspond to the six different RC circuits we have to consider. The conductance matrix for the circuit with switch k closed is $\tilde{G}$ with G added to its kth diagonal element, so the kth LMI constraint states that the dominant time constant of the circuit with switch k closed is less than $T_{\max}$.

Fig. 14 shows the optimal widths for two solutions on the tradeoff curve. The connections in the figure are drawn with a thickness proportional to x_{ij}. (Note however that the scales are not the same in the left and right figure.) We note that the topology of both designs are different. The left solution, which is faster, uses more connections than the solution on the right. Also note that in both cases the optimal topologies have loops.

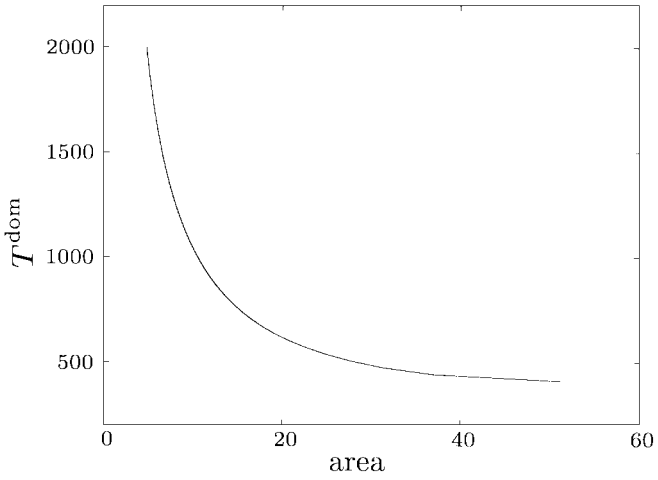

Fig. 13. Area-delay tradeoff.

Fig. 15 shows the step responses for the first solution ($T^{\rm dom} = 410$). The results confirm what we expect. The smallest delay arises when the input node is one or two (the first two plots in the left column) since they lie in the middle. The delay is larger when the input node is one of the four other nodes. Note that $T^{\rm dom}$ is equal in four of the six cases.

Again, the Elmore delay is slightly higher than dominant time constant and the dominant time constant is slightly higher than the 50%-threshold delay.

VII. Combined Wire Sizing and Spacing

The third application demonstrates another important advantage of using dominant time constant instead of Elmore delay: the ability to take into account nongrounded capacitors.

The problem is to determine the optimal wire widths *and* spacings for a bus taking into account the coupling capacitances between the wires. We consider an example with three wires, each consisting of five segments, as shown in Fig. 16. The optimization variables are the widths w_{ij} and the distances s_1 and s_2 between the wires.

The RC model of the three wires is shown in Fig. 17. The wires are connected to a voltage source with output conductance G at one end and to capacitive loads at the other end. As in the previous example, each segment is modeled as a π-segment with conductance and capacitance proportional to the segment width w_{ij}. We include a parasitic capacitance between the wires. We assume that there is a capacitance between the jth segments of wires 1 and 2, and between the jth segments of wires 2 and 3, with total values inversely proportional to the distances s_{1j} and s_{2j}, respectively. To obtain a lumped model, we split this distributed capacitance over two capacitors: the capacitance between segments j of wires 1 and 2 is lumped in two capacitors with value γ/s_{1j} and the total capacitance between segments j of wires 2 and 3 is lumped in two capacitors with value γ/s_{2j}. This leads to the RC circuit in Fig. 17.

We also impose the constraints that the distances s_{ij} between the wires must exceed a value $s_{\min}$ and that wire widths

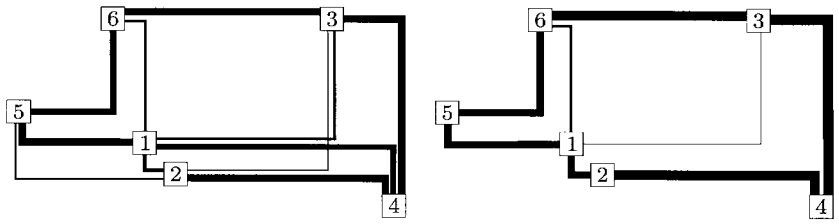

Fig. 14. Two solutions on the tradeoff curve. The left figure shows the line widths for the solution with $T^{\mathrm{dom}} = 410$. The width of the wires is proportional to x_{ij}. The width of the wires between (1, 5), (2, 4), (3, 4), and (3, 6) is equal to the maximum allowed value of one. There is no connection between node pairs (2, 6), (3, 5), (4, 5), and (4, 6). The right figure is the solution on the tradeoff curve for $T^{\mathrm{dom}} = 2000$. The widest connection is between nodes (3, 4) and has width 0.14. Again all connections are drawn with a width proportional to x_{ij}. In this solution the connections between (1, 4), (2, 3), (2, 5), (2, 5), (2, 6), (3, 5), (4, 5), and (4, 6) are absent. (Note when comparing both figures, that a different scale was used for the widths in both figures. The sizes in the right figure are roughly seven times smaller than in the left figure.)

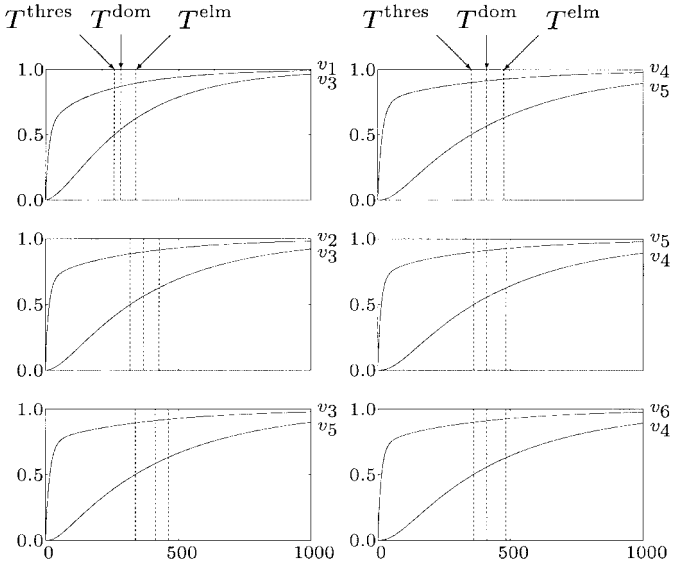

Fig. 15. Step responses for the solution on the tradeoff curve with $T^{\mathrm{dom}} = 410$ (i.e., the solution shown on the left in Fig. 14). The top left figure is the step response when switch 1 is closed, the second figure in the left column is the step response when switch 2 is closed, etc. Each shows the fastest and the slowest of the six responses. We also indicate the values of the dominant time constant, the critical Elmore delay, and the critical 50%-threshold delay.

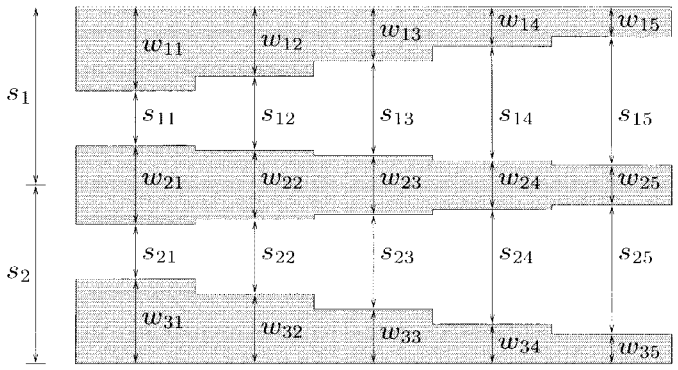

Fig. 16. Wire sizing and spacing. Three parallel wires consisting of five segments each. The conductance and capacitance of the jth segment of wire i is proportional to w_{ij}. There is a capacitive coupling between the ith segments of wires 1 and 2, and between the ith segments of wires 2 and 3, and the value of this parasitic capacitance is inversely proportional to s_{1i}, and s_{2i}, respectively. The optimization variables are the 15 segment widths w_{ij} and the distances s_1 and s_2.

are less than $w_{\max}$. We can minimize the total width $s_1 + s_2$ of the three wires subject to a bound on the dominant time

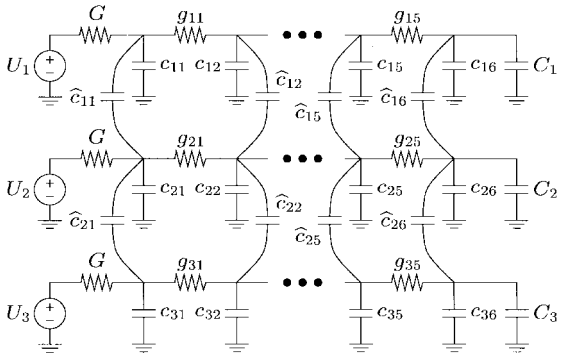

Fig. 17. RC model of the three wires shown in Fig. 16. The wires are connected to voltage sources with output conductance G at one end and to load capacitors C_i at the other end. The conductances g_{ij} and capacitances c_{ij} are part of the π-models of the wire segments. The capacitances $\hat{c}_{ij}$ model the capacitive coupling. The conductances and capacitances depend on the geometry of Fig. 16 in the following way: $g_{ij} = \alpha w_{ij}, c_{i1} = \beta w_{i1}, c_{ij} = \beta(w_{ij} + w_{i(j-1)})\ (1<j<6), c_{i6} = \beta w_{i5}, \hat{c}_{i1} = \gamma/s_{i1}, \hat{c}_{ij} = \gamma/s_{ij} + \gamma/s_{i(j-1)}\ (1<j<6), \hat{c}_{i6} = \gamma/s_{i5}$.

constant of the circuit by solving the optimization problem

$$\begin{aligned} \text{minimize} \quad & s_1 + s_2 \\ \text{subject to} \quad & T_{\max} G(w) - C(w,s) \geq 0 \\ & s_{1j} = s_1 - w_{1j} - 0.5 w_{2j}, \qquad j = 1, \cdots, 5 \\ & s_{2j} = s_2 - w_{3j} - 0.5 w_{2j}, \qquad j = 1, \cdots, 5 \\ & s_{ij} \geq s_{\min}, \qquad i = 1, 2, \quad j = 1, \cdots, 5 \\ & w_{ij} \leq w_{\max}, \qquad i = 1, 2, 3, \quad j = 1, \cdots, 5 \end{aligned} \tag{17}$$

in the variables s_1, s_2, w_{ij}, s_{ij}. Note that the capacitance matrix contains terms that are inversely proportional to the variables s_{ij} and therefore problem (17) is *not* an SDP. However, by a change of variables $t_{ij} = 1/s_{ij}$, problem (17) can be reformulated as a convex optimization problem

$$\begin{aligned} \text{mimimize} \quad & s_1 + s_2 \\ \text{subject to} \quad & T_{\max} G(w) - C(w,t) \geq 0 \\ & 1/t_{1j} \leq s_1 - w_{1j} - 0.5 w_{2j}, \qquad j = 1, \cdots, 5 \\ & 1/t_{2j} \leq s_2 - w_{3j} - 0.5 w_{2j}, \qquad j = 1, \cdots, 5 \\ & 0 \leq t_{ij} \leq 1/s_{\min}, \qquad i = 1, 2, \quad j = 1, \cdots, 5 \\ & w_{ij} \leq w_{\max}, \qquad i = 1, 2, 3, \quad j = 1, \cdots, 5 \end{aligned} \tag{18}$$

with variables s_1, s_2, t_{ij} and w_{ij}. Note that we replace the equalities in the second and third constraints by inequalities.

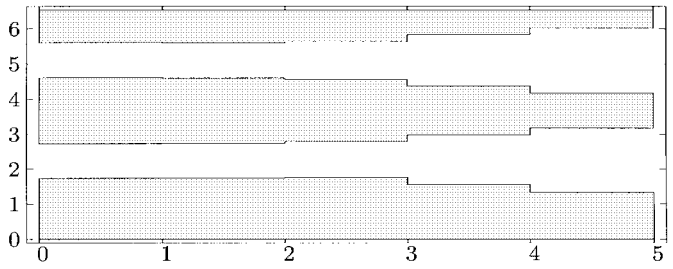

Fig. 18. Solution of (17) for $T_{\max} = 130$. Note that the distance between the wires is equal to its minimal allowed value of 1.0.

We first argue that this can be done without loss of generality. Suppose (s_i, w_{ij}, t_{ij}) are feasible in (18) with a certain objective value and that one of the nonlinear inequalities in t_{ij}, e.g., the inequality

$$1/t_{1j} \leq s_1 - w_{1j} - 0.5w_{2j}$$

is not tight. Decreasing t_{1j} increases the smallest eigenvalue of the matrix $T_{\max}G(w) - C(w,t)$. (This is readily shown from the Courant–Fischer minimax theorem. It is also quite intuitive: reducing coupling decreases the dominant time constant.) Therefore we can replace t_{1j} by the value

$$\tilde{t}_{1j} = 1/(s_1 - w_{1j} - 0.5w_{2j})$$

while still retaining feasibility in (18) and without changing the objective value. Without loss of generality, we can therefore assume that at the optimum the second and third constraints in (18) are tight. Hence problem (18) is equivalent to (17).

Problem (18) can be readily cast as an SDP by expressing the second and third constraints as the LMI's

$$\begin{bmatrix} t_{1j} & 1 \\ 1 & s_1 - w_{1j} - 0.5w_{2j} \end{bmatrix} \geq 0$$

and

$$\begin{bmatrix} t_{2j} & 1 \\ 1 & s_2 - w_{3j} - 0.5w_{2j} \end{bmatrix} \geq 0.$$

Figs. 18–21 illustrate the solution of (17) for two values of $T_{\max}$ assuming the parameter values

$$G = 100, \quad C_1 = 10, \quad C_2 = 20, \quad C_3 = 30$$
$$\alpha = 1, \quad \beta = 0.5, \quad \gamma = 2, \quad s_{\min} = 1, \quad w_{\max} = 2.$$

Figs. 18 and 19 illustrate a solution for $T_{\max} = 130$. The widest wire is number three since it drives the largest load, and the narrowest wire is number one which drives the smallest load. We also see that the smallest distance between the wires is equal to its minimum allowed value of 1.0 which means that the cross coupling did not affect the optimal spacing between the wires. Fig. 19 shows the output voltages for steps applied to one of the wires, while the two other input voltages remains zero.

Figs. 20 and 21 illustrate a solution for $T_{\max} = 90$. Here the distance between the second and third wires is larger than the minimum allowed value of 1.0. Fig. 21 shows the output voltages for the same situations as above.

As an interesting variation, we consider the same circuit, but with the signal direction of the second wire reversed, i.e., we place the driver at the right end of the second wire and the capacitance C_2 at the left. The rest of the circuit and the paramaters are left unchanged (Figs. 22–25).

Note in the application of this section that we cannot guarantee that the peak due to crosstalk stays under a certain level. This would be a specification in practice, but it is difficult to incorporate into the optimization problem. However we influence the level indirectly: minimizing the dominant time constant makes the crosstalk peak shorter in time (since the dominant time constant determines how fast all voltages settle around their steady-state value). Indirectly, this also tends to make the magnitude of the peak smaller (as can be seen by comparing the crosstalk levels for the two solutions in the examples).

A practical heuristic based on the dominant time constant minimization that would guarantee a given peak level is as follows. We first solve problem (17) for a given value of $T_{\max}$. Then we simulate to see if the crosstalk level is acceptable. If not, we increase the spacing of the wires until it is. Then we determine the optimal wire sizes again, keeping the wires at least at this minimum distance. This iteration is continued until it converges. The dominant time constant of the final result will be at least as good as the first solution and the crosstalk level will not exceed the maximum level.

VIII. Conclusions

We have presented a new method for wire and transistor sizing based on using the dominant time constant as a measure of signal delay in RC circuits. The main advantage of using this measure is that RC circuits with general nontree topologies can be optimally sized using convex optimization. This is in contrast to Elmore delay sizing methods, which only work for RC trees. We demonstrated the power of this method by applying it to several important examples of significant practical importance: sizing of clock meshes, sizing of a tristate bus, and sizing and spacing of a bus, taking crosstalk into account.

A. Computational Complexity of Dominant Time Constant Minimization

The method we described uses the recently developed interior-point methods for semidefinite programming (see, e.g., [27], [28]). Since real world sizing problems are likely to be very large, we briefly discuss the complexity of the SDP methods. Two factors determine the overall complexity of these methods: the total number of iterations and the complexity of an iteration. It can be shown that the number of iterations to solve an SDP to a given accuracy ϵ grows at most as $O(\sqrt{n}\log(1/\epsilon))$ where n is the size of the matrix $A(x)$ in (13) [27]. In practice, the performance is even better than suggested by this worst-case bound. The number of iterations usually lies between 5 and 50 and is almost independent of problem size. For practical purposes it is therefore fair to consider the total number of iterations as constant and to regard the complexity of an iteration as the key factor in the overall complexity.

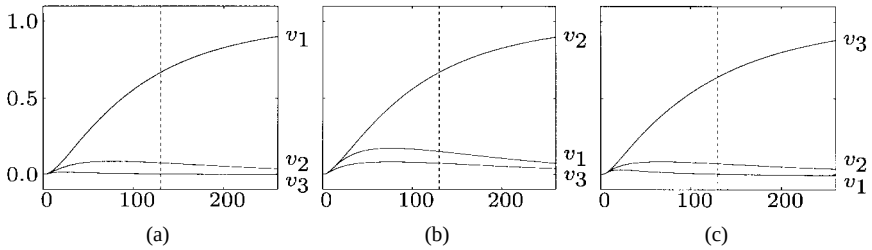

Fig. 19. Responses for the solution of Fig. 18. (a) The voltages at the output nodes due to a step applied to the first wire, (b) second wire, or (c) third wire. The dashed line marks the dominant time constant.

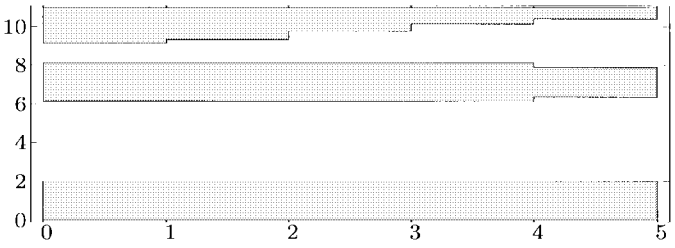

Fig. 20. Solution of (17) for $T_{\max} = 90$.

Each iteration involves solving a large system of linear equations to compute search directions. Little can be said about the complexity of this computation since it largely depends on the amount of problem structure that can be exploited. If the problem has no structure, i.e., if the matrices A_i in (13) are completely dense, then the cost of one iteration is $O(mn^3 + m^2n^2)$. This is the case for the general-purpose SDP software SP and SDPSOL [35], [36], which were used for the numerical examples in this paper. These codes solve problems up to several hundred variables without difficulty, but become impractical for larger problems since they do not exploit the problem structure. In all practical applications, however, there is a great deal of structure that can be exploited, and specialized codes are orders of magnitude more efficient than the general-purpose software (for a few examples see [37] and [38]).

SDP problems arising in dominant time constant minimization possess two forms of sparsity that should be exploited in a specialized code. First, the capacitance and conductance matrices C and G are usually sparse matrices (indeed C is often diagonal). Secondly, each variable x_i affects only a very small number of elements of C and G (i.e., the different matrices C_i and G_i in (2) are extremely sparse).

We can also comment on the current status of SDP software. Several general-purpose software packages for SDP's are now available, e.g., [35], [36], [39]–[45]. These packages exploit little or no sparse structure and are therefore only useful for circuits of small size (several hundred nodes). The examples in this paper were produced using the SP-package of [35]. To give an idea of the run time, solving one instance of the SDP (16) of Section V takes about 15 s on a 120 Mhz Pentium machine.

Semidefinite programming is a very active area of research in optimization, and much of this activity is directed at developing general-purpose software for large sparse SDP's (see [46], [42], 47] for a few recent reports in this topic). One can expect that this research will lead to software capable of handling much larger circuits (several thousand nodes).

For even larger problems (more than 10 000 nodes), it may be necessary to develop special-purpose techniques. These techniques can take advantage of the fact that the function $T^{\mathrm{dom}}(x)$ and its (sub)-gradients can be evaluated very efficiently using the Lanczos algorithm for computing the largest eigenvalue of a sparse symmetric matrix. In other words, the problems we described in this paper are not only convex, and hence fundamentally tractable, there also exist extremely efficient algorithms for evaluating the objective and constraint functions and their derivatives. Reference [48] describes recent work in this direction.

B. Comparison with Elmore Delay

We conclude with an overview of the differences between Elmore delay and the dominant time constant. The most important difference is that the dominant time constant *always* leads to tractable convex or quasi-convex optimization problems with no restrictions on circuit topology. This follows from (12) which holds regardless of the circuit topology. Specifically, we note the following advantages.

- Elmore delay optimization applies only to circuits with one input source. The dominant time constant can be applied to circuits with multiple sources, a problem that has only recently received attention [17], [49].
- The circuits may contain loops of resistors, e.g., clock meshes. Although for grounded capacitor RC circuits with loops of resistors, the Elmore delay is still a meaningful approximation of signal delay [13], [11], [14], it does not have the simple posynomial form as it does for RC trees, and convex optimization cannot be used to minimize it.
- The possibility of handling nontree topologies allows us to design the topology of the interconnection itself. For example, in the optimization of a clock mesh we start with a full grid of possible wire segments. After optimal wire sizes are computed, some (and often, many) of the wires have zero widths, which means they are not needed in the circuit (see also [17] for problems of designing interconnection topology).
- Dominant time constant minimization handles circuits with capacitive coupling between the nodes (see Section VII).

The Elmore time constant, in addition to being quite useful as a measure of delay when sizing RC trees, is sometimes

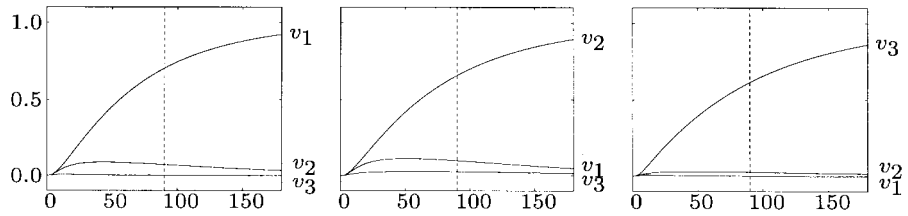

Fig. 21. Responses for the solution shown in Fig. 20. (a) The voltages at the output nodes, due to a step applied to the first wire, (b) second wire, or (c) third wire. The dashed line marks the dominant time constant.

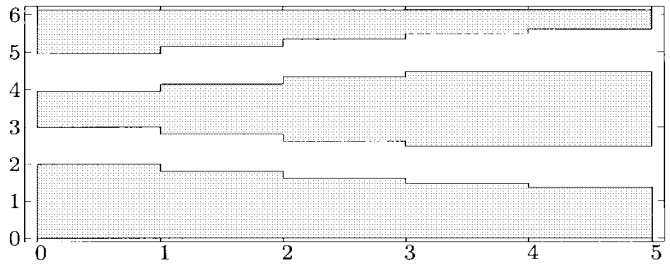

Fig. 22. Solution of (17) with the signal direction of the second wire reversed, and for $T_{\max} = 130$.

more appropriate to use than the dominant time constant. We give here two examples where this is the case.

- Consider a path consisting of several stages of buffered wire segments. The total Elmore delay of the path is the sum of the Elmore delays of the segments and is still a posynomial function that can be efficiently minimized by geometric programming. In contrast, it is not possible to efficiently minimize the sum of the dominant time constants since in general the sum of quasi-convex functions is not quasiconvex.
- The dominant time constant is useful as an alternative to the critical Elmore delay, i.e., the Elmore delay to the node with the slowest response. It is not a good measure for the delay to the other nodes.

Fishburn and Dunlop make an interesting remark in the conclusion of their paper on the TILOS program [3, §10]. They address the question whether it is justified to assume perfect step inputs, or whether the program should take into account a more realistic input waveform: "although there exist several static timing analyzers and a transistor sizer that take into account input waveform shape, we hesitate to do so without a convexity proof in hand. If a more accurate model turns out to be nonconvex, there is always the danger that the optimizer might become trapped in a local minimum that is not a global minimum, resulting in a more pessimal solution than the less acurate model."

A similar argument can be made in favor of the approach in this paper. Accurate expressions for the delay in transistor circuits are important for simulation and timing verification, and approximations based on the first few moments seem to be very well suited for this purpose (see, for example, [50]–[53]). For delay optimization, however, these expressions lead to complicated nonconvex optimization problems, with possibly many local minima. This is already the case for the Elmore delay (the first moment of the transfer function) of a grounded capacitor RC circuit with loops of resistors. Optimizing the dominant time constant on the other hand leads to tractable convex optimization problems even in general RC circuits.

Appendix I
Singular C or G

We now come back to the assumption in Section II that the capacitance matrix C and conductance matrix G are both strictly positive definite. This assumption simplified the definition and interpretation of the dominant time constant since it ensures that the number of generalized eigenvalues, i.e., the number of roots of the polynomial $\det(\lambda C + G)$ is exactly equal to n.

When both C and G were singular, the dominant time constant minimization still leads to meaningful results provided we do not define the dominant time constant in terms of the largest generalized eigenvalue, but use the LMI definition

$$T^{\mathrm{dom}} = \inf\{T | TG - C \geq 0\} \tag{19}$$

(we should add that $T^{\mathrm{dom}} = +\infty$ if there is no T with $TG - C \geq 0$). In this appendix we show that this definition is indeed meaningful and valid when C and G are only positive semidefinite, i.e., possibly singular.

Given arbitrary positive semidefinite C and G, one can always change coordinates to bring the circuit equations $C\dot{v} = -Gv$ into the form

$$\begin{bmatrix} \hat{C} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d\hat{v}_1/dt \\ d\hat{v}_2/dt \\ d\hat{v}_3/dt \end{bmatrix} = - \begin{bmatrix} \hat{G}_{11} & \hat{G}_{12} & 0 \\ \hat{G}_{12}^T & \hat{G}_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{v}_1(t) \\ \hat{v}_2(t) \\ \hat{v}_3(t) \end{bmatrix} \tag{20}$$

with $\hat{C}$ and $\hat{G}_{22}$ strictly positive definite. Note that these equations are a combination of differential and algebraic equations.

Assume $\hat{C} \in \boldsymbol{R}^{p \times p}$. The circuit equation (20) is equivalent to

$$\begin{aligned} d\hat{v}_1/dt &= -\hat{C}^{-1}\big(\hat{G}_{11} - \hat{G}_{12}\hat{G}_{22}^{-1}\hat{G}_{12}^T\big)\hat{v}_1(t) \\ \hat{v}_2(t) &= -\hat{G}_{22}^{-1}\hat{G}_{12}^T\hat{v}_1(t) \end{aligned}$$

and $\hat{v}_3(t)$ completely arbitrary. Let $\lambda_i, i = 1, \cdots, p$ be the eigenvalues of the matrix

$$-\hat{C}^{-1}\big(\hat{G}_{11} - \hat{G}_{12}\hat{G}_{22}^{-1}\hat{G}_{12}^T\big)$$

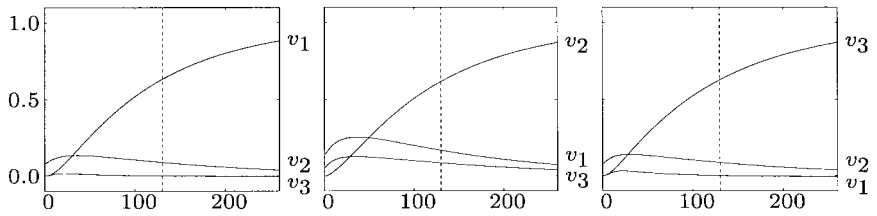

Fig. 23. Responses for solution of Fig. 22. (a) The voltages at the output nodes due to a step applied to the first wire, (b) second wire, or (c) third wire (right). The dashed line marks the dominant time constant. The voltage v_2 is the voltage across the capacitor C_2 which is placed at the left end of the wire.

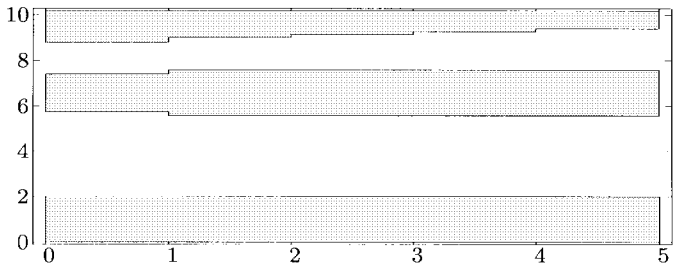

Fig. 24. Solution of (17) with the signal direction of the second wire reversed, and for $T_{\max} = 90$.

sorted in decreasing order. Then all solutions of (20) have the form

$$\widehat{v_1}(t) = \sum_i \alpha_i e^{\lambda_i t}, \quad \widehat{v_2}(t) = \sum_i \beta_i e^{\lambda_i t}, \quad \widehat{v_3}(t) \text{ arbitrary.}$$

The components of v_3 correspond to nodes that are not connected by capacitors or resistors to the rest of the circuit. It is therefore natural to ignore v_3 when defining the dominant time constant (or, equivalently, to impose the extra assumption that $v_3(t) \equiv 0$) and to say that $T^{\rm dom} = -1/\lambda_1$ (and $T^{\rm dom} = \infty$ if $\lambda_1 = 0$).

Finally, to see that this definition coincides with (19), note that $T \geq -1/\lambda_1$ if and only if the LMI

$$\begin{bmatrix} T\hat{G}_{11} - \hat{C} & T\hat{G}_{12} \\ T\hat{G}_{12}^T & T\hat{G}_{22} \end{bmatrix} \geq 0$$

holds. This can be easily shown by using a Schur complement (see e.g., [28]).

Appendix II
Some Relations Between the Delay Measures

In this section we derive several bounds between the three delay measures. The results allow us to translate upper bounds on $T^{\rm dom}$ into upper bounds on Elmore delay and threshold delays. Some of the bounds will turn out to be quite conservative. As the examples in the paper show, the 50% threshold delay, the Elmore delay, and the dominant time constant are much closer in practice than the bounds derived here would suggest.

A. Bounds on Node Voltages

We start by rewriting (7) as

$$v(t) = e^{-C^{-1}Gt}v(0) = C^{-1/2}e^{-C^{-1/2}GC^{-1/2}t}C^{1/2}v(0)$$

and use the second form to derive an upper bound on $\|v(t)\|_\infty$

$$\begin{aligned} \|v(t)\|_\infty &= \|C^{1/2}e^{-C^{-1/2}GC^{-1/2}t}C^{-1/2}v(0)\|_\infty \\ &\leq \|C^{1/2}\|_\infty \|e^{-C^{-1/2}GC^{-1/2}t}\|_\infty \\ &\quad \|C^{-1/2}\|_\infty \|v(0)\|_\infty \\ &\leq \sqrt{n}\kappa_\infty(C^{1/2})\|e^{-C^{-1/2}GC^{-1/2}t}\| \, \|v(0)\|_\infty \\ &= \sqrt{n}\kappa_\infty(C^{1/2})\|v(0)\|_\infty e^{-t/T^{\rm dom}} \end{aligned} \tag{21}$$

where the condition number κ_∞ is defined as $\kappa_\infty(A) = \|A\|_\infty\|A^{-1}\|_\infty$ and $\|A\|$ denotes the spectral norm of A, i.e., its largest singular value. The first inequality follows from the submultiplicative property of the matrix norm ($\|AB\|_\infty \leq \|A\|_\infty\|B\|_\infty$) and the definition of the infinity-induced matrix norm ($\|Ax\|_\infty \leq \|A\|_\infty\|x\|_\infty$). The second inequality follows from the relation between the infinity-induced and the spectral norm of a matrix ($\|A\|_\infty \leq \sqrt{n}\|A\|$ for $A \in \boldsymbol{R}^{m\times n}$). In the last line we used the fact that the largest eigenvalue of the symmetric matrix $e^{-C^{-1/2}GC^{-1/2}t}$ is $e^{-t/T^{\rm dom}}$ and that the largest eigenvalue and the spectral norm of a positive definite symmetric matrix coincide.

Note that for diagonal $C = \mathrm{diag}(C_1, \cdots, C_n)$ we have $\kappa_\infty(C^{1/2}) = (\max_i C_i)^{1/2}/(\min_j C_j)^{1/2}$.

For grounded capacitor RC circuits with $v(0) > 0$ we can also derive a lower bound on $\|v(t)\|_\infty$. Recall that for a grounded capacitor RC circuit the matrix $e^{-C^{-1}Gt}$ is elementwise nonnegative. We therefore have

$$\begin{aligned} \max_k v_k(t) &= \max_k \left(e_k^T e^{-C^{-1}Gt}v(0)\right) \\ &\geq v_{\min}(0) \max_k (e_k^T e^{-C^{-1}Gt}\mathbf{1}) \\ &\geq v_{\min}(0)e^{-t/T^{\rm dom}} \end{aligned} \tag{22}$$

where $v_{\min}(0)$ is the smallest component of $v(0)$. The last inequality follows from the Gershgorin disk theorem [59, p. 341], which together with the elementwise nonnegativity, implies that the eigenvalues of the matrix $e^{-C^{-1}Gt}$ are bounded above by largest row sum $\max_k e_k^T e^{-C^{-1}Gt}\mathbf{1}$. In a grounded-capacitor RC tree we can assume $v(0) = \mathbf{1}$ and therefore

$$\max_k v_k(t) \geq e^{-t/T^{\rm dom}}. \tag{23}$$

B. Threshold Delay and Dominant Time Constant

From (21), we see that for

$$t \geq T^{\rm dom} \log\left(\sqrt{n}\kappa_\infty(C^{1/2})\frac{\|v(0)\|_\infty}{\alpha}\right)$$

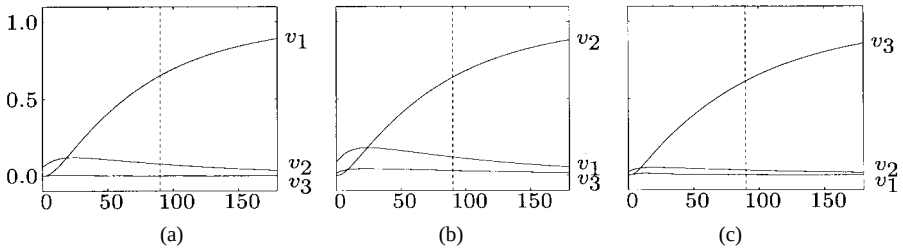

Fig. 25. Responses for the solution shown in Fig. 24. (a) The voltages at the output nodes, due to a step applied to the first wire, (b) second wire, (c) or third wire. The dashed line marks the dominant time constant.

TABLE I
BOUNDS FOR GROUNDED CAPACITOR RC TREES κ STANDS FOR $C_{\max}^{1/2}/C_{\min}^{1/2}$

	T^{thres}	T^{elm}	T^{dom}
T^{thres}		$\leq T^{\text{elm}}/\alpha$	$\leq T^{\text{dom}} \log\left(\sqrt{n}\kappa/\alpha\right)$
T^{elm}	$\leq T^{\text{thres}}\sqrt{n}\kappa/\log(1/\alpha)$		$\leq T^{\text{dom}}\sqrt{n}\kappa$
T^{dom}	$\leq T^{\text{thres}}/\log(1/\alpha)$	$\leq T^{\text{elm}}$	

we have $\|v(t)\|_\infty \leq \alpha$, so we conclude

$$T^{\text{thres}} \leq T^{\text{dom}} \log\left(\sqrt{n}\kappa_\infty(C^{1/2})\frac{\|v(0)\|_\infty}{\alpha}\right).$$

In a similar way, we can derive from (22) the lower bound

$$T^{\text{thres}} \geq T^{\text{dom}} \log\left(\frac{v_{\min}(0)}{\alpha}\right) \tag{24}$$

on the critical threshold delay of a grounded capacitor RC circuit. If $v(0) = \mathbf{1}$, we obtain

$$T^{\text{thres}} \geq T^{\text{dom}} \log(1/\alpha).$$

C. Dominant Time Constant and Elmore Delay

From (21), we have for each k

$$\begin{aligned} T_k^{\text{elm}} &= \int_0^\infty v_k(t)\, dt \\ &\leq \sqrt{n}\kappa_\infty(C^{1/2})\|v(0)\|_\infty \int_0^\infty e^{-t/T^{\text{dom}}}\, dt \\ &= T^{\text{dom}}\sqrt{n}\kappa_\infty(C^{1/2})\|v(0)\|_\infty. \end{aligned} \tag{25}$$

Thus we have a bound between critical Elmore delay and dominant time constant. For grounded capacitor circuits we obtain the lower bound from (23)

$$T^{\text{elm}} \geq T^{\text{dom}} v_{\min}(0)$$

and for a grounded capacitor RC tree

$$T^{\text{elm}} \geq T^{\text{dom}}.$$

D. Threshold Delay and Elmore Delay

We have already seen that $T^{\text{thres}} \leq T^{\text{elm}}/\alpha$ when the voltage decays monotonically.

For a grounded capacitor circuit we can also put together bounds (25) and (24), which yields

$$T^{\text{elm}} \leq T^{\text{thres}}\sqrt{n}\kappa_\infty(C^{1/2})\frac{\|v(0)\|_\infty}{\log(v_{\min}(0)/\alpha)}$$

and for a grounded capacitor RC tree

$$T^{\text{elm}} \leq T^{\text{thres}}\sqrt{n}\kappa_\infty(C^{1/2})\frac{1}{\log(1/\alpha)}.$$

E. Summary

Table I summarizes the bounds for grounded-capacitor RC trees for which we have a complete set of upper and lower bounds.

ACKNOWLEDGMENT

The authors would like to thank F. Wu of Intel for suggesting a tristate bus example. They are also grateful to J. Fishburn, S. Sapatnekar, and the reviewers for very useful comments and suggestions.

REFERENCES

[1] W. C. Elmore, "The transient response of damped linear systems with particular regard to wideband amplifiers," *J. Appl. Phys.*, vol. 19, pp. 55–63, 1948.
[2] J. Rubinstein, P. Penfield, and M. A. Horowitz, "Signal delay in RC tree networks," *IEEE Trans. Computer-Aided Design*, vol. 2, pp. 202–211, Feb. 1983.
[3] J. P. Fishburn and A. E. Dunlop, "TILOS: A posynomial programming approach to transistor sizing," in *Proc. ICCAD'85*, 1985, pp. 326–328.
[4] D. Hill, D. Shugard, J. Fishburn, and K. Keutzer, *Algorithms and Techniques for VLSI Layout Synthesis*. Norwell, MA: Kluwer, 1989.
[5] J.-M. Shyu, A. Sangiovanni-Vincentelli, J. P. Fishburn, and A. E. Dunlop, "Optimization-based transistor sizing," *IEEE J. Solid-State Circuits*, vol. 23, no. 2, pp. 400–409, 1988.
[6] D. P. Marple and A. El Gamal, "Optimal selection of transistor sizes in digital VLSI circuits," in *Stanford Conf. on VLSI*, 1987, pp. 151–172.
[7] S. Sapatnekar, V. B. Rao, P. Vaidya, and S.-M. Kang, "An exact solution to the transistor sizing problem for CMOS circuits using convex optimization," *IEEE Trans. Computer-Aided Design*, pp. 1621–1634, Dec. 1993.
[8] S. S. Sapatnekar, "Wire sizing as a convex optimization problem: exploring the area-delay tradeoff," *IEEE Trans. Computer-Aided Design*, vol. 15, pp. 1001–1011, Aug. 1996.

[9] J. Cong, L. He, C.-K. Koh, and P. H. Madden, "Performance optimization of VLSI interconnect layout," *Integration, VLSI J.*, vol. 21, pp. 1–94, 1996.
[10] P. K. Chan, "An extension of Elmore's delay," *IEEE Trans. Circuits Syst.*, vol. 33, pp. 1147–1152, Nov. 1986.
[11] J. L. Wyatt, "Signal delay in RC mesh networks," *IEEE Trans. Circuits Syst.*, vol. 32, pp. 507–510, 1985.
[12] P. K. Chan and M. D. F. Schlag, "Bounds on signal delay in RC mesh networks," *IEEE Trans. Computer-Aided Design*, vol. 8, pp. 581–589, June 1989.
[13] T.-M. Lin and C. A. Mead, "Signal delay in general RC networks," *IEEE Trans. Computer-Aided Design*, vol. 3, pp. 321–349, April 1984.
[14] J. L. Wyatt, "Signal propagation delay in RC models for interconnect," in *Circuit Analysis, Simulation and Design*, vol. 3, A. E. Ruehli, Ed. Amsterdam, The Netherlands: Elsevier, 1987, pp. 254–291.
[15] J. Cong and C. K. Koh, "Simultaneous driver and wire sizing for performance and power optimization," *IEEE Trans. VLSI Syst.*, pp. 408–425, Dec. 1994.
[16] T.-C. Lee and J. Cong, "The new line in IC design," *IEEE Spectrum*, pp. 52–58, Mar. 1997.
[17] J. Cong and L. He, "Optimal wiresizing for interconnects with multiple sources," in *Proc. IEEE Int. Conf. Computer-Aided Design*, 1995, pp. 568–574.
[18] J. Cong and K.-S. Leung, "Optimal wiresizing under Elmore delay model," *IEEE Trans. Computer-Aided Design*, pp. 321–336, Mar. 1995.
[19] N. Menezes, R. Baldick, and L. T. Pileggi, "A sequential quadratic programming approach to concurrent gate and wiring sizing," in *Proc. IEEE Int. Conf. Computer-Aided Design*, 1995, pp. 144–151.
[20] J. P. Fishburn, "Shaping a VLSI wire to minimize Elmore delay," in *Proc. Eur. Design Test Conf.*, 1997.
[21] J. P. Fishburn and C. A. Schevon, "Shaping a distributed-RC line to minimize Elmore delay," *IEEE Trans. Circuits Syst. I*, vol. 42, pp. 1020–1022, Dec. 1995.
[22] L. Vandenberghe, S. Boyd, and A. El Gamal, "Optimizing dominant time constant in RC circuits," Information Systems Laboratory, Stanford University, CA, Tech. Rep., Nov. 1996. Available: URL http://www-isl.stanford.edu/people/boyd/rc.html.
[23] A. Berman and R. J. Plemmons, "Nonnegative Matrices in the Mathematical Sciences," in *Classics in Applied Mathematics*. Philadelphia, PA: SIAM, 1994.
[24] R. Gupta, B. Krauter, B. Tutuianu, J. Willis, and L. T. Pileggi, "The Elmore delay as a bound for RC trees with generalized input signals," in *Proc. 32nd Design Automation Conf.*, 1995, pp. 364–369.
[25] S. Boyd and L. El Ghaoui, "Method of centers for minimizing generalized eigenvalues," *Linear Algebra Applicat., Special Issue Numerical Linear Algebra Methods in Contr., Signals Syst.*, vol. 188, pp. 63–111, July 1993.
[26] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, vol. 15 of *Studies in Applied Mathematics*. Philadelphia, PA: SIAM, June 1994.
[27] Yu. Nesterov and A. Nemirovsky, *Interior-point polynomial methods in convex programming*, vol. 13 of *Studies in Applied Mathematics*. Philadelphia, PA: SIAM, 1994.
[28] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Rev.*, vol. 38, no. 1, pp. 49–95, Mar. 1996.
[29] A. S. Lewis and M. L. Overton, "Eigenvalue optimization," *Acta Numerica*, pp. 149–190, 1996.
[30] F. Alizadeh, "Interior point methods in semidefinite programming with applications to combinatorial optimization," *SIAM J. Optim.*, vol. 5, no. 1, pp. 13–51, Feb. 1995.
[31] J.-P. A. Haeberly and M. A. Overton, "Optimizing eigenvalues of symmetric definite pencils," in *Proc. 1994 American Contr. Conf. Baltimore*, 1994.
[32] Y. E. Nesterov and A. S. Nemirovskii, "An interior-point method for generalized linear-fractional programming," *Math. Programming*, vol. 69, no. 1, pp. 177–204, July 1995.
[33] A. Nemirovsky, "Long-step method of analytic centers for fractional problems," Technion, Haifa, Israel, Tech. Rep. 3/94, 1994.
[34] M. P. Desai, R. Cvijetic, and J. Jensen, "Sizing of clock distribution networks for high performance CPU chips," in *Proc. 33rd ACM/IEEE Design Automation Conf.*, 1996, pp. 389–394.
[35] *Software for Semidefinite Programming. User's Guide, Beta Version.*, L. Vandenberghe and S. Boyd, Stanford University, CA, Oct. 1994. Available: http://www-isl.stanford.edu/people/boyd.
[36] SDPSOL: *A Parser/Solver for Semidefinite Programming and Determinant Maximization Problems with Matrix Structure. User's Guide, Version Beta.*, S.-P. Wu and S. Boyd, Stanford University, CA, June 1996.
[37] L. Vandenberghe and S. Boyd, "A primaldual potential reduction method for problems involving matrix inequalities," in *Math. Programming 69*, pp. 205–236, 1995.
[38] S. Boyd, L. Vandenberghe, and M. Grant, "Efficient convex optimization for engineering design," in *Proc. IFAC Symp. Robust Contr. Design*, Sept. 1994, pp. 14–23.
[39] *LMI Control Toolbox*, P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali, The MathWorks, Inc., Natick, MA, 1995.
[40] LMI-TOOL: *A Front-End for LMI Optimization, User's Guide*, L. El Ghaoui, R. Nikoukha, and F. Delebecque, 1995. Available via anonymous FTP: ftp.ensta.fr under /pub/elghaoui/lmitool.
[41] K. Fujisawa and M. Kojima, "SDPA (semidefinite programming algorithm) User's Manual," Department of Mathematical and Computing Sciences. Tokyo Institute of Technology, Tech. Rep. B-308, 1995.
[42] CSDP, *A C library for Semidefinite Programming*, B. Borchers, New Mexico Inst. of Mining Tech., Socorro, Mar. 1997.
[43] *SDPHA—A MATLAB Implementation of Homogeneous Interior-Point Algorithms for Semidefinite Programming*, F. A. Potra, R. Sheng, and N. Brixius, University of Iowa, Iowa City, Apr. 1997.
[44] *SDPT3:A MATLAB Software Package for Semidefinite Programming*, K. C. Toh, M. J. Todd, and R. H. Tütüncü, 1996. Available: http://www.math.nus.sg/~mattohkc/index.html.
[45] SDPPACK *User's Guide, Version 0.9 Beta.*, F. Alizadeh, J. P. Haeberly, M. V. Nayakkankuppam, M. L. Overton, and S. Schmieta, New York University, June 1997.
[46] S. J. Benson, Y. Ye, and X. Zhang, "Solving large-scale sparse semidefinite programs for combinatorial optimization," University of Iowa, Iowa City, Tech. Rep., Sept. 1997.
[47] K. Fujisawa, M. Kojima, and K. Nakata, "Exploiting sparsity in primal-dual interior-point methods for semidefinite programming," *Math. Programming*, vol. 79, no. 1-3, pp. 235–253, Oct. 1997.
[48] C. Helmberg and F. Rendl, "A spectral bundle method for semidefinite programming," Konrad-Zuse Zentrum fuer Informationstechnik Berlin, Tech. Rep. SC 97-37, 1997.
[49] J. Cong and L. He, "Optimal wiresizing for interconnects with multiple sources," *ACM Trans. Design Automation Electron. Syst.*, vol. 1, 1996.
[50] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design*, vol. 9, pp. 352–366, April 1990.
[51] P. Feldmann and R. W. Freund, "Efficient linear circuit analysis by Padé approximation via the Lanczos process," *IEEE Trans. Computer-Aided Design*, vol. 14, pp. 639–649, May 1995.
[52] K. Gallivan, E. Grimme, and P. Van Dooren, "Padé approximation of large-scale dynamic systems with Lanczos methods," in *Proc. 33rd Conf. Decision Contr.*, 1994, pp. 443–448.
[53] K. Gallivan, E. Grimme, and P. Van Dooren, "Asymptotic waveform evaluation via a Lanczos method," *Appl. Math. Lett.*, vol. 7, no. 5, pp. 75–80, 1994.
[54] B. Hoppe, G. Neuendorf, D. Schmitt-Landsiedel, and W. Specks, "Optimization of high-speed CMOS logic circuits with analytical models for signal delay, chip area, and dynamic power dissipation," *IEEE Trans. Computer-Aided Design*, vol. 9, pp. 236–247, Mar. 1990.
[55] W. Chuang, S. S. Sapatnekar, and I. N. Hajj, "Timing and area optimization for standard-cell VLSI circuit design," *IEEE Trans. Computer-Aided Design*, vol. 14, pp. 308–320, Mar. 1995.
[56] C. H. Tan and J. Allen, "Minimization of power in VLSI circuits using transistor sizing, input ordering, and statistical power estimation," in *Proc. 1st Int. Workshop Low Power Design*, Napa, CA, 1994, pp. 75–80.
[57] S. S. Sapatnekar, "A Convex Programming Approach to Problems in VLSI Design," Ph.D. dissertation, Coordinated Science Laboratory, University of Illinois at Urbana-Campaign, Aug. 1992.
[58] S. S. Sapatnekar and S.-M. Kang, *Design Automation for Timing-Driven Layout Synthesis*. Norwell, MA: Kluwer, 1993.
[59] G. Golub and C. Van Loan, *Matrix Computations*, 2nd ed. Baltimore, MD: The Johns Hopkins Press, 1989.

Lieven Vandenberghe received the electrical engineering degree and the Ph.D. degree in electrical engineering from the Katholieke Universiteit Leuven, Belgium, in 1987 and 1992, respectively.

He is an Assistant Professor in the Electrical Engineering Department at the University of California, Los Angeles. He was previously a Research Associate in the Information Systems Laboratory at Stanford University, CA. His research interests are in optimization and its applications in VLSI, control, and signal processing.

Stephen Boyd received the A.B. degree in mathematics from Harvard University, Cambridge, MA, in 1980 and the Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley, in 1985.

In 1985, he joined the Electrical Engineering Department at Stanford University, CA, where he is now a Professor and the Director of the Information Systems Laboratory. His interests include computer-aided control system design and convex programming applications in control, signal processing, and circuits.

Abbas El Gamal received the Ph.D. degree in electrical engineering from Stanford University, CA, in 1978.

He is currently an Associate Professor of Electrical Engineering at Stanford University. From 1978–1980 he was an Assistant Professor of electrical engineering at the University of Southern California, Los Angeles. He was on leave from Stanford University from 1984–1987 first as Director of LSI Logic Research Lab, where he developed compilation technology and DSP and image processing ASIC's, then as Co-founder and Chief Scientist of Actel Corporation, an FPGA supplier. From 1990–1995 he was a Co-founder and Chief Technical Officer of Silicon Architects, which is currently part of Synopsys. His research interests include: VLSI circuits, architectures, and synthesis, FPGA's and mask programmable gate arrays, image sensors, image compression, error correction, and information theory. He has authored or coauthored over 60 papers and holds 15 patents in these areas.