BOUNDS ON INVERSE OF THE NODE-CONDUCTANCE MATRIX

Pak K. Chan

June 1987
CSD-870021

Bounds on Inverse of the Node-Conductance Matrix

Pak K. Chan

Computer Science Department
University of California, Los Angeles
Los Angeles, California 90024
(213)825-2266

Abstract - In [1,2], we have shown that the delay estimation problem can be solved by finding the inverse of the node-conductance matrix $\mathbf{G}$ of a resistive network. The problem of finding $\mathbf{R} \equiv \mathbf{G}^{-1}$ is approached in this report in several ways. First, the use of an exact expression is examined. This method involves examining the geometry of the network and then applying a topological formula to find the desired driving-point resistance. Second, bounds for $\mathbf{R}$ are presented; we attempt to find two matrices $\overline{\mathbf{R}}=(\overline{R}_{i,j})$ and $\underline{\mathbf{R}}=(\underline{R}_{i,j})$ such that $\overline{R}_{i,j} \geq R_{i,j} \geq \underline{R}_{i,j}$, for all i and j. Evaluating these bounds largely involves finding a least-resistance path between the ground node and the rest of the nodes in a network. Third, these bounds can serve as initial guesses in an iterative procedure that would tighten up the delay bounds from both sides.

1. Tree and Mesh Networks

Topologically, we can classify *RC* circuits as either *tree* or *mesh*. By a *tree* network, we mean that the resistor network formed by replacing all voltage sources by short-circuits and removing all capacitors is a tree; otherwise it is a *mesh* network. Recall that the node-conductance matrix of a given network is denoted by $\mathbf{G}$, and that $\mathbf{R} \equiv \mathbf{G}^{-1}$ is therefore the resistance matrix.

For a given *RC tree* network, $R_{i,k}$ is the resistance of the portion of the (unique) path between the excitation and k that is common with the (unique) path between the excitation and node i. In particular, $R_{i,i}$ is the resistance between the excitation and node i [10]. Hence, the resistance matrix **R** can be readily found by inspection. Also, evaluating delays for all n nodes of a tree network requires only n multiplications. Figure 1.a shows an *RC* tree network. In this example, entries of the **R** matrix are $R_{1,1} = R_1$, $R_{2,2} = R_1 + R_2$, $R_{1,2} = R_1$, etc. And the signal delay at node i is

$$t_{d,i} = \sum_{k=1}^{n} R_{i,k}\, C_k\, [v_k(\infty) - v_k(0)] + \int_0^{\infty} [e(\infty) - e(t)]\, dt \qquad ; i = 1, \ldots, n\,. \tag{1}$$

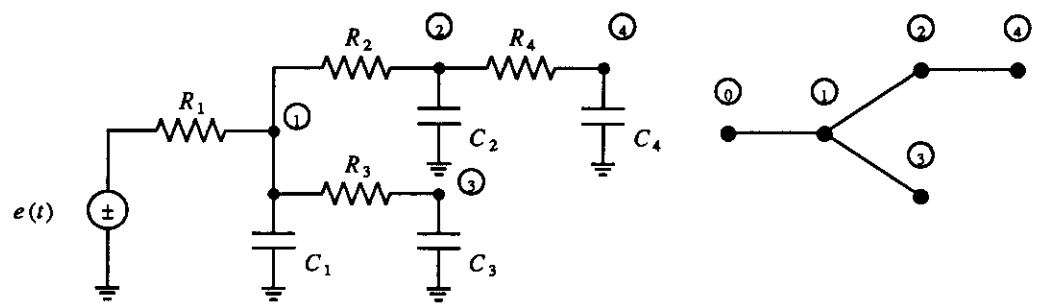

Figure 1.a. *RC* Tree and its Graph.

Tree networks are easy to evaluate; unfortunately, many practical MOS circuits are *mesh RC* networks. As pointed out by Wyatt [15], the *RC* mesh networks arise in MOS circuits on a number of occasions. One class of examples comes from lumped network models for the gates of large transistors used in the final stages of clock or pad driver. To minimize resistance and capacitance, the gate poly is sometimes laid out in a rectangular grid or a closed loop [7], and

therefore cannot be modeled as an *RC* tree. Another class of examples include interconnect loops created by automatic routers and pass transistor networks in which reconvergent paths are enabled simultaneously.

Speaking in terms of the components of the **R** matrix, $R_{i,i}$ is the input resistance or driving-point resistance between node i and the ground, and $R_{i,j}$ is the *transresistance* between $R_{i,i}$ and $R_{j,j}$. Algebraically, $R_{i,j}$ is the common term or sum of common terms between $R_{i,i}$ and $R_{j,j}$. For instance, Figure 1.b shows a mesh *RC* circuit with $R_{1,1} = R_1$, $R_{2,2} = R_1 + (R_2 \,||\, (R_3 + R_4 + R_5))$, $R_{1,2} = R_1$, etc.

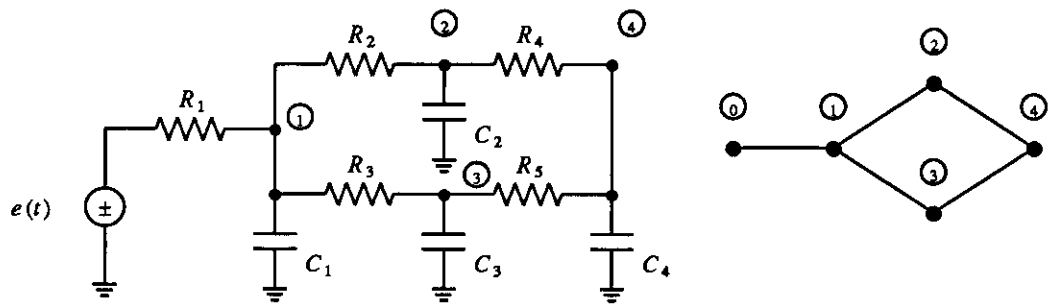

Figure 1.b. *RC* Mesh and its Graph.

There are substantial differences between finding delays for a tree and a mesh network. Given an arbitrary network, it seems helpful to determine first whether it is a tree or mesh. When the ground node is taken into account, given a *connected* graph of $n+1$ nodes and b branches (edges), the necessary and sufficient condition for the network to be a tree is that $b=n$.

Subsequently, our focus for the rest of this report will be on treatments of mesh networks.

2. A Topological Method for Finding Driving-point Resistances

First, we examine a classical approach for finding the driving-point resistance $R_{i,j}$ between any pair of nodes i and j of a given network [14]. Despite the fact that this method delivers the exact expression for $R_{i,j}$, it is not attractive since it involves listing all the *spanning trees* and *two-trees* of the network. Alternatively, we shall consider bounds; namely, to find two quantities $\overline{R}_{i,j}$ and $\underline{R}_{i,j}$ such that

$$\overline{R}_{i,j} \geq R_{i,j} \geq \underline{R}_{i,j} \ . \tag{2}$$

We shall demonstrate that this is a viable approach since it involves no more than finding a least-resistance path of a network.

To facilitate further discussion, we cannot avoid at least a few definitions. These definitions are fairly standard in the literature. Consider a graph $G=(N,E)$, consisting of a set of nodes $N=\{n_0,n_1,\cdots,n_n\}$ and a set of edges $E=\{e_1,\cdots,e_b\}$. There is a positive real number $R(e_i)\equiv R_i$ associated with each edge e_i. Without loss of generality, the node n_0 designates the ground node.

Definitions:

A *path* $P:v\rightarrow w$ in G is a sequence of nodes and edges leading from v to w. A path is *simple* if all its nodes are distinct.

A path $P: v \rightarrow v$ is called a *closed path.* A closed path $P: v \rightarrow v$ is a cycle if all its nodes are distinct and the only node to occur twice in P is v.

The *resistance* of a path is the *sum* of the values of edges on the path.

A graph is *connected* if there is a path between every pair of nodes.

A *tree* T is a connected graph without cycles.

A tree T is a *spanning tree* of G if T is a subgraph of G and T contains all nodes in G.

A *two-tree* of G is a pair of disjoint trees T_1 and T_2 such that $N = N_1 \cup N_2$.

The *degree*, d_i, of a node i is the number of branches connected to it.

An (s,t) cutset is a set of edges such that any path from s to t uses at least one edge of the set.

A classical result [3, 14] states that

$$R_{i,i} = \frac{T_i}{D}, \tag{3}$$

where

$$D \equiv \sum_{k=1}^{S} G_1^{(k)} G_2^{(k)} \cdots G_n^{(k)} \qquad T_i \equiv \sum_{q=1}^{Q} G_1^{(q)} G_2^{(q)} \cdots G_{n-1}^{(q)} \qquad G \equiv \frac{1}{R} \tag{4}$$

Let us define the *value* of a tree as the product of its branch conductances. A term in the summation for D is equal to the value of a spanning tree; the summation is carried out over all the S spanning trees. Thus, D is a homogeneous polynomial of degree n in R. If from any spanning tree one branch is first removed, then we have a pair of disjoint trees (two-trees). If the ground node and the ith node are not on the same tree, then the product of the values of these two disjoint trees constitute a term of T_i. Thus, T_i is a homogeneous polynomial of degree $n-1$ in R.

The main difficulty lies in using formula (4) involves listing all the spanning trees as well as the two-trees. For instance, applying the topological formula to node 4 of the mesh network as shown in Figure 1.b yields,

$$D = \frac{1}{R_1R_2R_4R_5} + \frac{1}{R_1R_2R_3R_4} + \frac{1}{R_1R_2R_3R_5} + \frac{1}{R_1R_3R_4R_5},$$

and

$$T_i = \frac{1}{R_1}\left[\frac{1}{R_4R_5}+\frac{1}{R_2R_3}+\frac{1}{R_2R_5}+\frac{1}{R_3R_4}\right] + \frac{1}{R_3R_4R_5}+\frac{1}{R_2R_4R_5}+\frac{1}{R_2R_3R_5}+\frac{1}{R_2R_3R_4}$$

or

$$R_{4,4} = R_1+\frac{(R_2+R_4)(R_3+R_5)}{R_2+R_3+R_4+R_5}.$$

3. Bounds on Driving-point Resistance

As illustrated, the problem of finding the driving-point resistance $R_{i,i}$ for node i is often cumbersome. We attack the related but easier problem of finding bounds for $R_{i,i}$. That is, we aim to find two quantities $\overline{R}_{i,i}$ and $\underline{R}_{i,i}$ such that

$$\overline{R}_{i,i} \geq R_{i,i} \geq \underline{R}_{i,i} \qquad ; i=1,\ldots,n \ .$$

First, we establish bounds for $R_{i,i}$, which is the driving-point resistance between a given node i and the ground. Then we consider bounds on the transresistance $R_{i,j}$, for $i \neq j$.

3.a. Some Provable Bounds

Lemma 1: The value, $\overline{R}_{i,i}$, of a least-resistance path (shortest path) between node i and the ground is an upper bound on $R_{i,i}$. That is, $\overline{R}_{i,i} \geq R_{i,i}$. □

To prove the stated result, we need the following theorem.

Theorem 1. Consider a one-port network consisting of two-terminal, positive resistors and a switch, as in Figure 2. Let R_o and R_c be the input resistances of the network with the switch open and closed respectively. Then $R_o \geq R_c$. □

This theorem was proved quite elegantly from Tellegen's theorem [9].

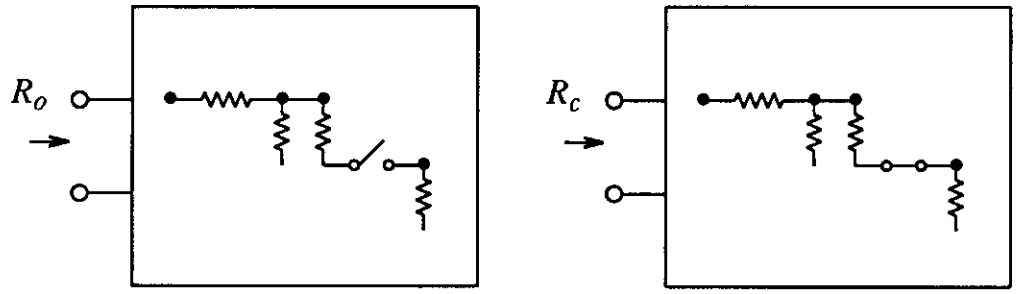

Figure 2. Open and Close Resistances.

Proof of Lemma 1: With Theorem 1 in mind, we are ready to establish the upper bound. Let n_1 and n_2 be two nodes of a resistive network G at which we wish to find the driving-point resistance R. Let P be a shortest path from n_1 and n_2. Now, let G_c be $G-P$. We can envision the network G comprising P and G_c connected to each other by some switches, as in Figure 3. When all the switches are closed, the input resistance of the network is R; whereas the input resistance is equal to the resistance of P when all the switches are opened. Thus, it follows immediately from Theorem 1 that $\bar{R} \geq R$.

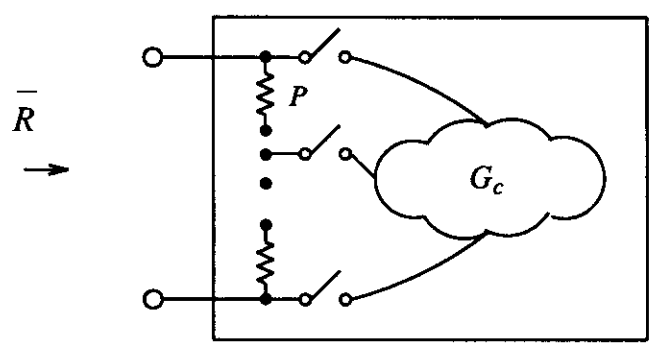

Figure 3. Proof of the Upper Bound.

Lemma 2: Since, by Theorem 1, introducing short-circuits in a resistor network cannot increase the input resistance, for a given resistive network G, a lower-bound network can be derived by short-circuiting *some* nodes in G such that the resulting network is a *series-parallel* resistor network. According to Duffin [5], a network is a series-parallel connection if and only if there is no embedded network having the Wheatstone bridge configuration.

The absence of Wheatstone bridge configuration enables the input resistance, $\underline{R}_{i,i}$, of such a series-parallel network to be easily evaluated by Ohm's Law. Namely, a) resistances are additive for resistors in series, and b) conductances are additive for resistors in parallel. Alternatively, a worst-case estimate for the resistance of a parallel connection of resistors is the least resistance among the resistors divided by the number of resistors.

□

There can be many ways of short-circuiting some nodes of a given network to derive a series-parallel network. Here is one systematic way. First, we arrange $G=(N,E)$ as a *multistage graph* G' with respect to a given node i. A multistage graph $G'=(N,E)$ is a graph in which the nodes are partitioned into $s\geq 2$ sets, N'_j (or stages), $1 \leq j \leq s$; such that for each edge $(u,v)\in E$, then $u\in N'_j$ and $v\in N'_k$ for some $1<j\leq k<s$. This implies that there can be edges within a stage. The sets N'_1 and N'_s are chosen to be $N'_1=\{i\}$ and $N'_s=\{0\}$. There may be edges that *skip* stages. For instance, in Figure 4.a the edge that has resistance R connects stages 2 and 4 and bypassing stage 3. By introducing new nodes and edges to redistribute the resistances, we come up with a multistage graph which does not have edges that skip stage. The white nodes in Figure 4.b are there for the purpose of bridging. Specifically, the resistor R has been split into two resistors of resistances $R/2$ joined together by a white node. In general, if a resistor skips h number of stages, we split the

resistor into $h+1$ pieces and introduce h white nodes, one for each of the stages that is skipped. Finally, a lower-bound network is derived by short-circuiting all the nodes within each stage. This results in a series-parallel resistor network as in Figure 4.c.

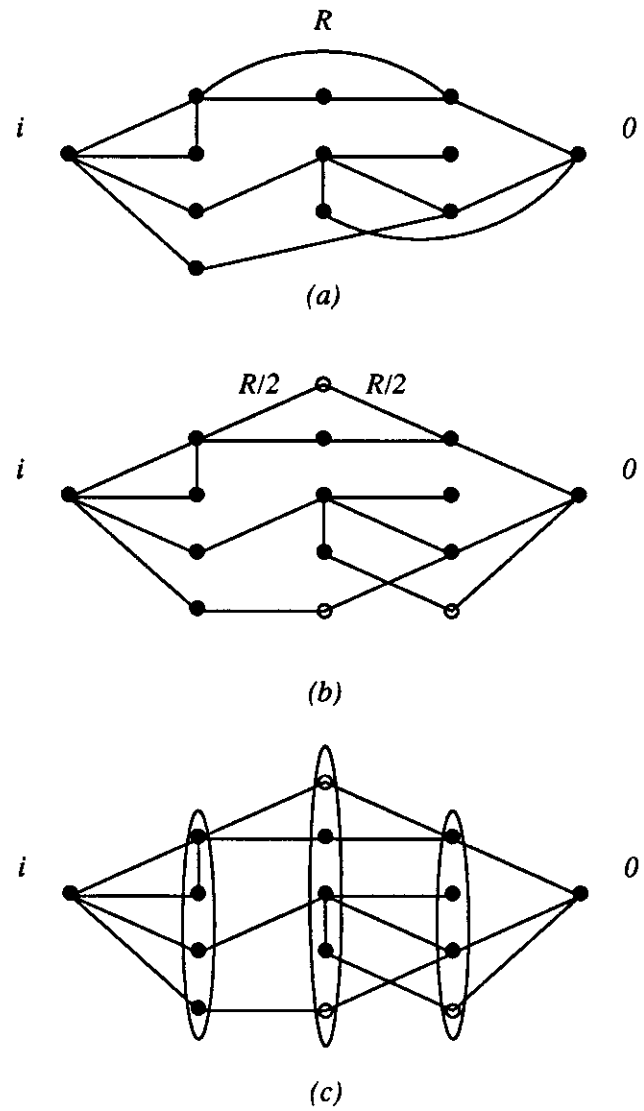

Figure 4. Steps in Deriving Series-Parallel Network.

In order to obtain a tight lower bound by the forementioned procedure, it seems appropriate to short-circuiting those branches with smaller resistances. Therefore, in rearranging G as a multistage graph one should place the smaller resistors within the same stage whenever it is possible.

Example: Figure 5.a illustrates a Wheatstone bridge resistor network. R is the resistance between the two terminals when the switch is open. A lower-bound network, as in Figure 5.b is derived by short-circuiting R_1. Let $\underline{R}$ be the resistance of such a network. We have,

$$R \geq \underline{R} = \frac{R_2 R_3}{R_2 + R_3} + \frac{R_4 R_5}{R_4 + R_5} \geq \frac{1}{2}\left[\min(R_2, R_3) + \min(R_4, R_5)\right].$$

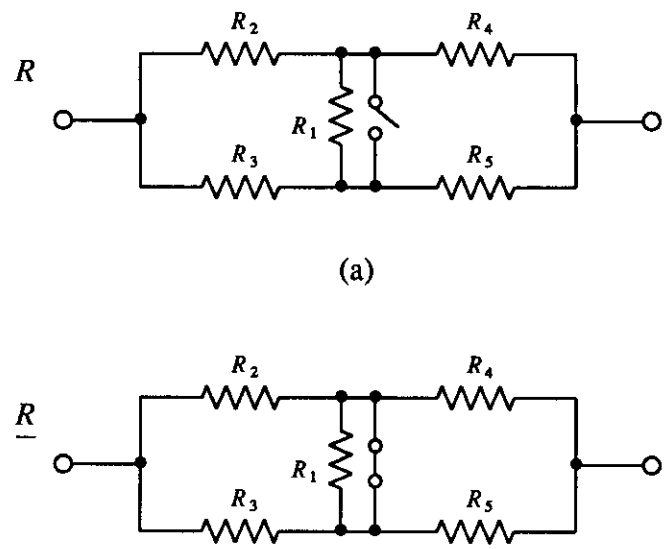

Figure 5. Wheatstone Bridge and its Derived Network.

After furnishing $\overline{R}_{i,i}$, $\overline{R}_{j,j}$, $\underline{R}_{i,i}$, and $\underline{R}_{j,j}$, the next problem is finding $\overline{R}_{i,j}$ and $\underline{R}_{i,j}$ for $i \neq j$.

Lemma 3: Since $R_{i,j}$ is the common terms between $R_{i,i}$ and $R_{j,j}$, an upper bound for $R_{i,j}$ is

$$\overline{R}_{i,j} = \min(\overline{R}_{i,i}, \overline{R}_{j,j}) . \tag{5}$$

□

When all the $\overline{R}_{i,i}; i=1,\ldots,n$ are desired, the upper bound suggests by Lemma 1 involves finding a shortest path between the ground node and the rest of the nodes in the network. The shortest path problem is a well solved problem. Dijkstra's algorithm is probably the best algorithm for our purpose since it requires only $O((n+1)^2)$ additions [4]. Admittedly, the lower bound of Lemma 2 is not as convenient to evaluate as the upper bound, since it requires multiplications and divisions for the step in calculating parallel connection of resistors. More seriously, it does not seem easy to establish a reasonable lower bound $\underline{R}_{i,j}$ based on $\underline{R}_{i,i}$ and $\underline{R}_{j,j}$. To remedy this, we suggest another lower bound which is efficient to evaluate. Unfortunately, we are unable to prove its validity rigorously at this point, so we shall only conjecture.

3.b. A Dubious Lower Bound

A lower bound, $\underline{R}_{i,i}$ on $R_{i,i}$ is

$$\frac{\bar{R}_{i,i}}{w} \tag{6}$$

where w is the *max cut* of the graph. The *max cut* is the number of nodes in an *maximum* cutset. This cutset partitions the graph G into two disjoint graphs $G_1=(N_1,E_1)$ and $G_2=(N_2,E_2)$, where N_1 and N_2 are disjoint, and G_1 and G_2 are connected graphs, such that the number of edges in the cutset is maximized. Figure 6 shows the max cuts for a tree and a mesh network.

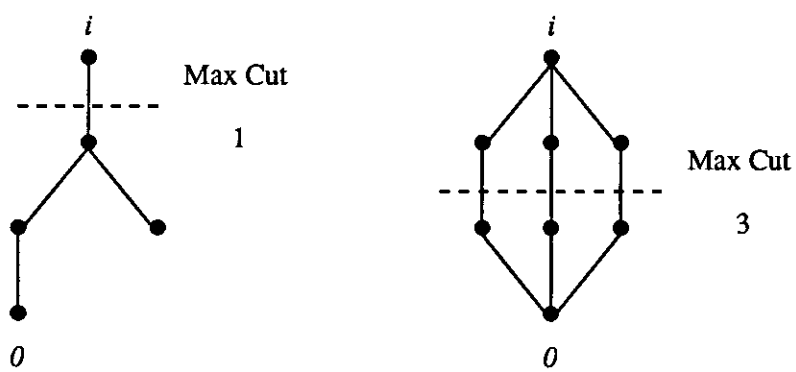

Figure 6. Max Cuts of Tree and Mesh Networks.

The max cut for a planar graph can be found by a polynomial bounded algorithm [8]. When the graph is nonplanar, the problem of finding the max cut is NP-complete [6]. Even so, we can map the nonplanar graph into a planar one by introducing new nodes and edges at the crossovers. The max cut of this planar graph would obviously be greater than or equal to the max cut of the original nonplanar graph. But since we are looking for a lower bound for $R_{i,i}$, an over estimate of the max cut would only make the lower bound cruder. Intuitively, the max cut is an upper bound on the maximum possible number of parallel branches. Notice that w is equal to one for any tree network, therefore the upper bound ($\overline{R}_{i,i}$) and lower bound ($\underline{R}_{i,i}$) are both equal to $R_{i,i}$; whereas the lower bound is exact if a network consists of b identical parallel branches.

Claim (6) can be established by a conjecture. The intuition behind it is as follows. Let n_1 and n_2 be the two nodes of a resistive network G at which we wish to find the input resistance. Let w be the max cut for the network, and P be a shortest path. Notice that if all d_1 branches that are connected to n_1 are replaced by w number of P branches running from n_1 to n_2, this cannot increase the input resistance. Because P is a shortest path, any other path that runs from n_1 to n_2 must be greater than or equal to P. After the replacement, we have a network consisting of w parallel branches of Ps. The input resistance of such a network is simply the resistance of P divided by w, hence the stated lower bound is established.

Assuming that (6) is true, then we proceed to find $\underline{R}_{i,j}$. For any given node i there can be more than one shortest path from i to the ground. Let $\{P_{1,i}, \cdots, P_{g,i}\}$ and $\{P_{1,j}, \cdots, P_{h,j}\}$ be the sets of shortest paths for nodes i and j respectively. Then, a lower bound on $R_{i,j}$ is the maximum sum of the common edges for all the shortest paths between

node i and j, that is

$$\underline{R}_{i,j} = \frac{\max\limits_{1 \le r \le g, 1 \le q \le h} \left\{ \sum\limits_{e \in P_{r,i} \cap P_{q,j}} R(e) \right\}}{w} . \tag{7}$$

□

This lower bound may be conservative in the sense that it is possible for $R_{i,i}$ and $R_{j,j}$ to have common resistances which do not lie on any one of the shortest paths. In this case, $\underline{R}_{i,j}$ will be grossly under-estimated to be zero.

Example:

Figure 7 shows a mesh resistive network with 5 branches. Given that the resistance values are as shown, we wish to find the resistance matrix bounds of this network and compare them with the exact answer.

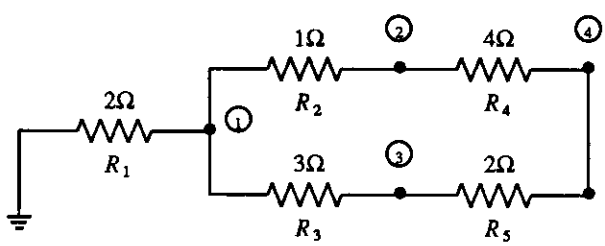

Figure 7. Resistive Network.

Since $\overline{R}_{i,i}$ is just a shortest path from node i to the ground, we have

$$\bar{R}_{1,1} = 2\Omega \qquad \bar{R}_{2,2} = 3\Omega \qquad \bar{R}_{3,3} = 5\Omega \qquad \bar{R}_{4,4} = 7\Omega\,,$$

and with $w_1=1, w_2=w_3=w_4=2$; by using the lower bounds in (6), we have

$$\underline{R}_{1,1} = 2\Omega \qquad \underline{R}_{2,2} = 1.5\Omega \qquad \underline{R}_{3,3} = 2.5\Omega \qquad \underline{R}_{4,4} = 3.5\Omega\,,$$

whereas using the exact expression gives

$$R_{1,1} = 2\Omega \qquad R_{2,2} = 2.9\Omega \qquad R_{3,3} = 4.1\Omega \qquad R_{4,4} = 4.5\Omega\,.$$

We see that these lower bounds are a bit conservative. We suggest one way to improve them. A graph that can be divided into two complementary subgraphs with *only one node in common* has been called *separable*. The common node is referred to as the *articulation point*. If the network under consideration is separable, a tighter lower bound can be obtained as the sum of the lower bounds of the separable components. A good algorithm for separating a connected graph into components has been described by Tarjan [12]. This algorithm requires O(b) time when applied to a connected graph with b edges.

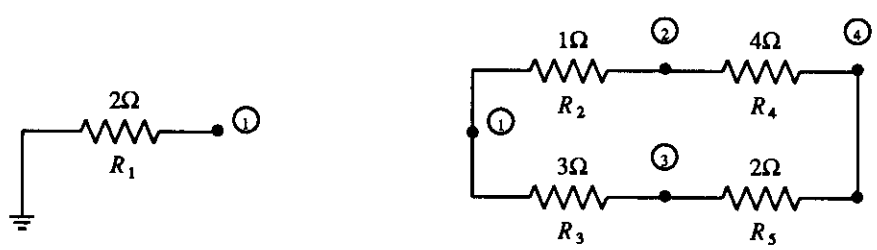

Figure 8. Decomposed Resistive Network.

Now, returning to Figure 7, node 1 is an articulation point. We decompose the resistive network at node 1 into two parts, as shown in Figure 8. The lower bound of the network is evaluated as the sum of the lower bounds of its components. Thus, the improved lower bounds are,

$$\underline{R}_{1,1} = 2\Omega \qquad \underline{R}_{2,2} = 2.5\Omega \qquad \underline{R}_{3,3} = 3.5\Omega \qquad \underline{R}_{4,4} = 4.5\Omega\,.$$

Using (5) and (7), the $\overline{\mathbf{R}}$ and $\underline{\mathbf{R}}$ resistance matrices are

$$\overline{\mathbf{R}} = \begin{bmatrix} 2 & 2 & 2 & 2 \\ 2 & 3 & 3 & 3 \\ 2 & 3 & 5 & 5 \\ 2 & 3 & 5 & 7 \end{bmatrix} \Omega \qquad \underline{\mathbf{R}} = \begin{bmatrix} 2 & 2 & 2 & 2 \\ 2.0 & 2.5 & 2.0 & 2.5 \\ 2.0 & 2.0 & 3.5 & 3.5 \\ 2.0 & 2.5 & 3.5 & 4.5 \end{bmatrix} \Omega\,,$$

whereas the exact $\mathbf{R}$ matrix is

$$\mathbf{R} = \begin{bmatrix} 2 & 2 & 2 & 2 \\ 2.0 & 2.9 & 2.3 & 2.7 \\ 2.0 & 2.3 & 4.1 & 3.5 \\ 2.0 & 2.7 & 3.5 & 4.5 \end{bmatrix} \Omega\,.$$

4. Bounds on Delays

From the bounds on the resistance matrix $\mathbf{R}$, bounds for the delays ($\mathbf{t}_d$) for the linear model can simply be,

$$\overline{\mathbf{t}}_d = \overline{\mathbf{R}}\mathbf{C}\cdot[\mathbf{v}(\infty)-\mathbf{v}(0)] + \overline{\mathbf{R}}\cdot\mathbf{D}\cdot\int_0^\infty[\mathbf{e}(\infty) - \mathbf{e}(t)]dt \ ,$$

$$\underline{\mathbf{t}}_d = \underline{\mathbf{R}}\mathbf{C}\cdot[\mathbf{v}(\infty)-\mathbf{v}(0)] + \underline{\mathbf{R}}\cdot\mathbf{D}\cdot\int_0^\infty[\mathbf{e}(\infty) - \mathbf{e}(t)]dt \ . \tag{8}$$

If these bounds are unsatisfactory, we need some way to tighten them up. The idea is to present the problem as a system of linear equations; namely,

$$\mathbf{G}\cdot\mathbf{t}_d = \mathbf{C}\cdot[\mathbf{v}(\infty)-\mathbf{v}(0)] + \mathbf{D}\cdot\int_0^\infty[\mathbf{e}(\infty) - \mathbf{e}(t)]dt \ ,$$

and use a numerical iterative procedure to solve for $\mathbf{t}_d$. If one is interested in an estimate of $\mathbf{t}_d$, then it is appropriate to use the mean of (8) as a good initial estimate for the iterative procedure. On the other hand, bounds on $\mathbf{t}_d$ can also be obtained if there is an iterative bounding procedure that would use (8) as the initial conditions and successively tighten up the bounds. Iterative bounding procedures have been presented in some excellent literature such as [11] and [13].

5. Remarks and Conclusions

We have addressed the problem of finding the inverse of the node-conductance matrix of a resistive network in several ways. Our main result is the bounding matrices, $\overline{\mathbf{R}}$ and $\underline{\mathbf{R}}$. Since the derivation of these matrices depends solely on Tellegen's theorem, our result apparently is applicable to all resistive networks that obey Kirchhoff's laws, whether the networks are linear or nonlinear, time-invariant or time-varying, reciprocal or

nonreciprocal.

We observe that the upper bound matrix $\overline{\mathbf{R}}$ tends to be closer to $\mathbf{R}$ if the resistive network is *tree-like*. Likewise, $\underline{\mathbf{R}}$ tends to be closer to $\mathbf{R}$ if the resistive network consists mainly of parallel connections of identical resistors.

From the bounding matrices, we can obtain bounds on delays.

REFERENCES

[1] P. K. Chan, "An Extension of Elmore's Delay," *IEEE Trans. on Circuits and Systems,* Vol. CSA-33, No. 11, Nov. 1986, pp. 1147-1149.

[2] P. K. Chan, "An Extension of Elmore's Delay and its Application for Timing Analysis of MOS Pass Transistor Networks," *IEEE Trans. on Circuits and Systems,* Vol. CSA-33, No. 11, Nov. 1986, pp. 1149-1152.

[3] S. P. Chan, "Introductory topological analysis of electrical networks," in *Holt, Rinehart and Winston,* 1969.

[4] E. W. Dijkstra, "A Note on Two Problems in Connexion with Graphs," *Numerische Mathematik,* No. 1, 1959, pp. 269-271.

[5] R. J. Duffin, "Topology of series-parallel networks," *J. Math. Analysis Application,* Vol. 31, No. 85, 1965, pp. 303-318.

[6] M. Garey and D. Johnson, in *Computers and Intractibility: A guide to the Theory of NP-Completeness,* San Francisco: Freeman, 1979.

[7] L. A. Glasser and D. A. Dobberpuhl, *The Design and Analysis of VLSI Circuits*: Addison-Wesley, 1985.

[8] F. Hadlock, "Finding a Maximum Cut of a Planar Graph in Polynomial Time," *SIAM Journal on Computing,* Vol. 4, No. 3, Sept. 1975.

[9] P. Penfield, Jr., R. Spence, and S. Duinker, *Tellegen's Theorem and Electrical Networks*: M.I.T. Press, 1970.

[10] J. Rubinstein, P. Penfield, Jr., and M. A. Horowitz, "Signal Delay in RC Tree Networks," *IEEE Trans. on CAD of Integrated Circuits and Systems,* Vol. CAD-2, No. 3, Jul. 1983, pp. 202-211.

[11] J. Schroder, *Operator Inequalities*: Academic Press, 1980.

[12] R. Tarjan, "Depth-First Search and Linear Graph Algorithms," *SIAM Journal of Computing,* June 1972, pp. 146-160.

[13] R. S. Varga, *Matrix Iterative Analysis*: Prentice-Hall Inc., 1962.

[14] L. Weinberg, *Network Analysis and Synthesis*: McGraw-Hill Book Company, 1962.

[15] J. L. Wyatt, Jr., "Signal Delay in RC Mesh Networks," *IEEE Transactions on Circuits and Systems,* Vol. CSA-32, No. 5, May 1985, pp. 507-510.