TIMING ANALYSIS OF TREE-LIKE RLC CIRCUITS

B. Rajendran, V. Kheterpal, A. Das, J. Majumder,C. Mandal and P. P. Chakrabarti

Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur, India 721302

ABSTRACT

A new moment based technique for timing analysis of tree-like interconnect RLC circuits is presented. It is shown to be capable of handling circuits with resistors and inductors connected to ground and capacitors connected between non-ground nodes, in addition to the common Transmission-tree like structure. The technique uses a moment-based representation scheme for circuit elements, an algebra to combine moments and a node absorption strategy. A linear-time algorithm is presented for computing the moments of the transfer function. Experimental results indicate that the scheme is both accurate and fast enough to handle large circuits.

1. INTRODUCTION

With the clock frequency of today's microprocessors and systems approaching the Gigahertz range, the issue of determining the signal delay of RLC interconnect circuits accurately and efficiently has become very critical in the design flow process. Various techniques have been proposed to determine the timing behavior of interconnect circuits. Circuit simulations based on SPICE and other similar methods do not prove to be computationally efficient to handle large circuits. Moment based approaches based on the Asymptotic Waveform Evaluation(AWE) [1] have been found to be very fast and accurate for determining the time response of RLC interconnect circuits and has been efficiently extended for special topologies of tree-like circuit configurations in the Rapid Interconnect Circuit Evaluator (RICE) [2].

However, the performance of these techniques degrade when there are deviations in the circuit topology from the "pure" RLC Transmission tree-like structure. A pure tree-like circuit will not have any closed loops, resistors and inductors will only be connected between non-grounded nodes and capacitors will be always connected to ground. The deviations from pure tree-like structure are mainly due to the presence of resistor loops (resistors connected to ground), floating nodes (due to capacitors connected between two non-grounded nodes) or inductor links (inductors connected between a node and ground) in the circuit. Various other techniques for determining time response of interconnect circuits (like [3]) based on moment methods are also applicable only to restricted circuit configurations.

In this paper, we propose a new linear-time algorithm to determine the moments for a tree-like RLC circuit, and show how to predict the approximate nature of timing behavior of the circuit using them. The algorithm can evaluate the transfer function moments of a tree-like circuit, even when there are resistor loops, floating nodes or inductor links, without any extra computational cost. A scheme to represent circuit elements using moments and rules for combining moments of circuit elements using a moment algebra are presented. We then present a traversal scheme using a node absorption technique by which the moments of the transfer function at every output node can be obtained. This is implemented by two depth-first traversals of the circuit graph.

We then show how these moments can be used to approxmiate the timing behavior of the circuit being analyzed. Experimental results indicate that our methods are accurate when compared to SPICE and efficient in execution time when run for large circuits. For example, circuits containing 30,000 elements are completed in 2 seconds of CPU time on a Pentium III processor. The speed of the algorithm is tested for various types of elements including the special cases that we wish to handle and the efficiency is found to be independent of the types of elements to a great extent.

2. CONCEPT OF MOMENTS

Let $\mathbf{h(t)}$ be a circuit impulse response in the time domain and let $\mathbf{H(s)}$ be the corresponding transfer function. Then, by definition, $\mathbf{H(s)}$, the Laplace Transform of $\mathbf{h(t)}$is

$$\mathbf{H(s)} = \int_0^\infty \mathbf{h(t)} e^{-st} dt \tag{1}$$

The i^{th} moment of the function $\mathbf{h(t)}$ is defined from the above expression as

$$\mathbf{m_i} = \frac{(-1)^i}{i!} \int_0^\infty t^i \mathbf{h(t)} dt \tag{2}$$

The behavior of a circuit can be approximated by a few dominant poles of the system, which can be determined from the first few moments of the transfer function of the circuit. It has been established ([1], [4]) that this approximation technique helps in ascertaining the circuit behavior to a very high degree of accuracy.

Moment based techniques for circuit behavior approximation proceed in two steps: (i) Determining the moments of the transfer function from the circuit and (ii) Matching the moments to the approximate poles and residues to obtain the time response. Of the two, determination of moments is the most critical part, in terms of computational requirements and complexity. RICE determines the moments by a repeated graph traversal of the circuit [2]. However, when it encounters resistor loops or floating nodes in the circuit, the moment calculation is done by a modified nodal analysis of the circuit. The complexity of this approach increases with the number of such ("impure") elements in the circuit. The computation time therefore approaches the complexity of a complete circuit analysis

B. Rajendran is currently with the Solid State and Photonics Laboratory, Stanford Univeristy, CA, USA

A. Das is currently with the Computer Systems Laboratory, Stanford Univeristy, CA, USA

0-7803-7448-7/02/$17.00 ©2002 IEEE

when the number of such elements approach the size of the circuit and is quite high as it involves large matrix operations.

Our approach produces a linear-time algorithm that is independent of the number of resistor loops, inductor links and floating nodes in the circuit. Our method treats the basic circuit elements, namely, resistor, capacitor and inductor alike by representing their admittance functions as moments (expressing them as posynomial coefficents) and works out a uniform scheme to compose them.

The next section proposes the scheme to represent circuit elements as moment terms and the method to combine them for obtaining the moments of admittance functions corresponding to circuit branches.

3. REPRESENTATION OF CIRCUIT ELEMENTS AS MOMENTS

For a general two-port network, the voltage transfer function can be obtained by performing a Thevenin Anlaysis at the node where the transfer function is required. Let the *Thevenin Equivalent Voltage* and the *Thevenin Admittance* at the desired node be represented as $\mathbf{V_{Th}(s)}$ and $\mathbf{Y_{Th}(s)}$, respectively. Let the load admittance at the desired node be represented by $\mathbf{Y_L(s)}$. Then, from basic circuit theory, the transfer function $\mathbf{H(s)}$ is given by the expression

$$\mathbf{H(s) = V_{th}(s)\frac{Y_{th}(s)}{Y_{th}(s) + Y_L(s)}} \tag{3}$$

A core idea behind our approach is to symbolically represent the moment terms corresponding to the circuit elements as posynomial coefficients and to calculate the moments for the transfer function by defining an algebra for the basic mathematical operations on posynomial expressions.

To calculate the $\mathbf{k}$ most significant moment terms, it is sufficient to store the first $\mathbf{k}$ terms of the posynomial terms, with the coefficients arranged in increasing order.

The following convention is used to represent the moment terms of any function. Let a function $\mathbf{M(s)}$ be expressed a series expansion in powers of $\mathbf{s}$. Let the power of the lowest non-zero coefficient in that series be $\mathbf{n}$.

$$\mathbf{M(s) = m_0 s^n + m_1 s^{n+1} + m_2 s^{n+2} + \ldots} \tag{4}$$

Then, we represent the function as

$$[\mathbf{M(s)}] = [(n)(m_0, m_1, m_2, m_3 \ldots)] \tag{5}$$

Note that $\mathbf{n}$ can be a positive or negative integer, and $\mathbf{m_0}$ is a non-zero real number.

According to this representation, the admittance of the basic circuit elements will be written as follows.

Resistor

$$[\mathbf{Y_R(s)}] = [\mathbf{(0)(1/R, 0, 0, 0)}] \tag{6}$$

Capacitor

$$[\mathbf{Y_C(s)}] = [\mathbf{(1)(C, 0, 0, 0)}] \tag{7}$$

Inductor

$$[\mathbf{Y_L(s)}] = [\mathbf{(-1)(1/L, 0, 0, 0)}] \tag{8}$$

We next define the elementary algebraic operations for combining moments of circuit elements to obtain the moments of the admittance functions of circuits. This will give the method for combining circuit elements when connected in series and/or parallel to each other.

Addition and Subtraction of Moments

$$[\mathbf{(n_h)(h_0, h_1 \ldots)}] = [\mathbf{(n_a)(a_0, a_1 \ldots)}] \pm [\mathbf{(n_b)(b_0, b_1 \ldots)}]$$

$$n_h = min(n_a, n_b) \tag{9}$$

The moment terms of the sum (difference) of $\mathbf{A}$ and $\mathbf{B}$ the are obtained by adding (subtracting) the moments corresponding to the like terms (terms with the same exponent) of $\mathbf{A}$ and $\mathbf{B}$.

Multiplication of Moments

$$[\mathbf{(n_h)(h_0, h_1 \ldots)}] = [\mathbf{(n_a)(a_0, a_1 \ldots)}]\mathbf{x}[\mathbf{(n_b)(b_0, b_1 \ldots)}]$$

$$n_h = n_a + n_b \tag{10}$$

$$h_i = \sum_{k=0}^{i} a_k b_{i-k} \tag{11}$$

Division of Moments

$$[\mathbf{(n_h)(h_0, h_1 \ldots)}] = [\mathbf{(n_a)(a_0, a_1 \ldots)}]/[\mathbf{(n_b)(b_0, b_1 \ldots)}]$$

$$n_h = n_a - n_b \tag{12}$$

$$h_0 = \frac{a_0}{b_0} \tag{13}$$

$$h_i = \frac{a_i - \sum_{k=0}^{i-1} h_k b_{i-k}}{b_0} \tag{14}$$

3.1. Rules for Combining Moments of circuit elements

In this section, we present the rules for combining the moments of various elements in the branch of a circuit to determine its net admittance.

We define the terms *Downstream* and *Upstream* nodes based on the directional traversal of the circuit branch to elucidate these rules. A node a is said to be *upstream* to another node b, if node a is adjacent to node b and the direction of traversal is from node b to node a. As per the above convention, node b is said to be *downstream* to the node a.

- *Initialisation*

  Consider the circuit shown in Figure 1. Consider node n; there are three branches $b1$, $b2$ and $b3$ for this node as indicated. Suppose, we want to determine the admittance moments of a branch (say $b1$) in the circuit as seen from node n. Then, consider a traversal of the nodes of that particular branch starting from a leaf node in the branch towards node n.

  Based on the conventions mentioned earlier, let $\mathbf{Y_d(s)}$, the *Downstream Admittance Function* of a node, be defined as the net admittance of all elements looking downstream from that node.

  Hence, at the start of the algorithm, the *Downstream Admittance Function* at the first node to be traversed is initialized to the admittance moments of the first circuit element based on the expressions 6, 7 and 8.

  Let $\mathbf{Y_u(s)}$, the *Upstream Admittance Function* of a node refer to the net admittance of all the elements as seen from the node and including the element upstream of that node.

  This element (upstream) can be a a Resistor, Capacitor or an Inductor, appearing in series or parallel with its downstream node. Based on our moment representation scheme,

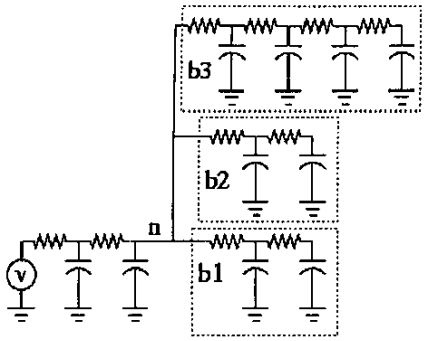

Fig. 1. Branches in a Circuit

expressions can be written to represent how the moments of the *Upstream Admittance Function* will change depending on the traversed element and the *Downstream Admittance Function*. The expressions corresponding to these six cases are given below.

- *Resistor in Series*

$$[\mathbf{Y_u(s)}] = \frac{[\mathbf{Y_d(s)}][(\mathbf{0})(\mathbf{1/R},\mathbf{0},\mathbf{0},\mathbf{0})]}{[\mathbf{Y_d(s)}] + [(\mathbf{0})(\mathbf{1/R},\mathbf{0},\mathbf{0},\mathbf{0})]} \quad (15)$$

- *Inductor in Series*

$$[\mathbf{Y_u(s)}] = \frac{[\mathbf{Y_d(s)}][(\mathbf{-1})(\mathbf{1/L},\mathbf{0},\mathbf{0},\mathbf{0})]}{[\mathbf{Y_d(s)}] + [(\mathbf{-1})(\mathbf{1/L},\mathbf{0},\mathbf{0},\mathbf{0})]} \quad (16)$$

- *Capacitor in Series*

$$[\mathbf{Y_u(s)}] = \frac{[\mathbf{Y_d(s)}][(\mathbf{1})(\mathbf{C},\mathbf{0},\mathbf{0},\mathbf{0})]}{[\mathbf{Y_d(s)}] + [(\mathbf{1})(\mathbf{C},\mathbf{0},\mathbf{0},\mathbf{0})]} \quad (17)$$

- *Resistor in Parallel*

$$[\mathbf{Y_u(s)}] = [\mathbf{Y_d(s)}] + [(\mathbf{0})(\mathbf{1/R},\mathbf{0},\mathbf{0},\mathbf{0})] \quad (18)$$

- *Capacitor in Parallel*

$$[\mathbf{Y_u(s)}] = [\mathbf{Y_d(s)}] + [(\mathbf{1})(\mathbf{C},\mathbf{0},\mathbf{0},\mathbf{0})] \quad (19)$$

- *Inductor in Parallel*

$$[\mathbf{Y_u(s)}] = [\mathbf{Y_d(s)}] + [(\mathbf{-1})(\mathbf{1/L},\mathbf{0},\mathbf{0},\mathbf{0})] \quad (20)$$

4. ALGORITHM FOR MOMENT CALCULATION OF CIRCUIT TRANSFER FUNCTIONS

Based on the above formulation of representing circuit elements, an algorithm to calculate the transfer function moments at all the nodes of an RLC tree is presented in this section. This algorithm requires two complete depth-first traversals (DFS) of the circuit.

In the first traversal, the moments of the admittance of all the downstream branches as seen from every node is computed and stored at the corresponding node. This is done by performing a pre-order traversal on the graph corresponding to the circuit, starting at the input node and doing the combination of moments along the return path of the traversal. The admittances are combined using the rules specified in Section 3.1.

In the second traversal, the moments for the transfer function at every node of the circuit is calculated. This is done by a DFS on the circuit, starting from the input node by a *Node Absorption* strategy as outlined below.

4.1. Node Absorption

Any excitation source can be considered as a *Thevenin Voltage source*, $\mathbf{V_{Th}}$, connected in series with an equivalent *Thevenin Admittance*, $\mathbf{Y_{Th}}$, or a *Norton Current source*, $\mathbf{I_N}$, connected in parallel with an equivalent *Norton Admittance*, $\mathbf{Y_N}$.

These quantities are related by the following equations

$$[\mathbf{V_{Th}}][\mathbf{Y_{Th}}] = [\mathbf{I_N}] \quad (21)$$

$$[\mathbf{Y_{Th}}] = [\mathbf{Y_N}] \quad (22)$$

These equivalence relations can be used to advance the excitation source from a node to its adjacent node, by absorbing the first node.

In general, by converting the excitation source from a Thevenin Equivalent to a Norton Equivalent, or vice versa, a node can be absorbed into its adjacent node.

4.2. Evaluation of Transfer Function Moments at the Leaf Nodes

To calculate the transfer function moments at a node, the algorithm tries to evaluate the moments of the *Equivalent Voltage* and the *Equivalent Admittance* at that nodes.

Starting from the driver node, the excitation source is shifted downstream, one node at a time, by the *Node Absorption* scheme. At every node where there are side branches, the admittances of the side branches are added up while performing the path traversal. The load admittance corresponding to a node is determined by summing up the admittances of all the branches that are downstream to that node.

When the transfer function is being evaluated at a node, the excitation source can either be a Norton Equivalent Current Source or a Thevenin Equivalent Voltage source. If a Norton Equivalent (denoted by $\mathbf{I_{Nf}}$ and $\mathbf{Y_{Nf}}$) is present, the following expression gives the moments of the Transfer Function $\mathbf{H(s)}$:

$$[\mathbf{H(s)}] = \frac{[\mathbf{I_{Nf}(s)}]}{[\mathbf{Y_L(s)}] + [\mathbf{Y_{Nf}(s)}]} \quad (23)$$

Similarly, if a Thevenin Equivalent (denoted by $\mathbf{V_{Nf}}$ and $\mathbf{Y_{Nf}}$) is present, the moments of $\mathbf{H(s)}$ can be obtained from the expression :

$$[\mathbf{H(s)}] = [\mathbf{V_{Nf}(s)}]\left[\frac{[\mathbf{Y_{Nf}(s)}]}{[\mathbf{Y_L(s)}] + [\mathbf{Y_{Nf}(s)}]}\right] \quad (24)$$

Also, at all nodes, the moments of the Equivalent Voltage and the Equivalent Admittance are stored for continuing with the path traversal during the DFS to the other nodes. This way, the transfer function at all the nodes are calculated. It is clear that each DFS takes linear time (for calculating a constant number of moment

terms) and therefore we have a linear-time algorithm to obtain the moments of the transfer function of the circuit under consideration.

5. TIME RESPONSE FROM MOMENTS

The basic idea behind all moment based approaches is to match the first $2q$ moments of the circuit to that of a reduced order response model, yielding accurate frequency and time-domain approximations in terms of the circuit's q most-dominant poles.

The reduced q-th order model is of the form :

$$\bar{h}_q(t) = k_1 e^{p_1 t} + k_2 e^{p_2 t} + \ldots + k_q e^{p_q t} \quad (25)$$

$$\bar{H}_q(s) = \frac{k_1}{s - p_1} + \frac{k_2}{s - p_2} + \ldots + \frac{k_q}{s - p_q} \quad (26)$$

As proposed in [1], the q dominant poles in the equation 25 are approximated by matching the first $2q$ moments of the circuit response. The steps to obtain the poles are residues of the reduced order model from the moments are reproduced here for the sake of completeness. The moment matching equations are obtained by equating like terms in the Taylor series expansion of the expression in equation 26.

$$m_0 + m_1 s + m_2 s^2 + \ldots \infty = -\sum_{j=1}^{q} \left[\frac{k_j}{p_j} \sum_{i=0}^{\infty} \left(\frac{s}{p_j} \right)^i \right] \quad (27)$$

From the above equation, we can write the $2q$ moment matching equations.

$$m_j = -\sum_{i=1}^{q} \frac{k_i}{p_i^{j+1}}, \forall j = 0, 1, 2 \ldots, 2q-1 \quad (28)$$

The above equations, which are non-linear in p_i can be transformed to a set of linear equations (in a set of variables a_i) as given below.

$$\begin{bmatrix} m_0 & m_1 & \cdots & m_{q-1} \\ m_1 & m_2 & \cdots & m_q \\ .. & . & \cdots & . \\ .. & . & \cdots & . \\ m_{q-1} & m_q & \cdots & m_{2q-2} \end{bmatrix} \begin{bmatrix} a_q \\ a_{q-1} \\ .. \\ .. \\ a_1 \end{bmatrix} = - \begin{bmatrix} m_q \\ m_{q+1} \\ .. \\ .. \\ m_{2q-1} \end{bmatrix} \quad (29)$$

where, $m_0, m_1, \ldots, m_{2q-1}$ are the $2q$ moments of the circuit transfer function.

The poles of the system are obtained by solving the characteristic equation

$$1 + a_1 s + a_2 s^2 + \ldots + a_{q-1} s^{q-1} + a_q s^q = 0 \quad (30)$$

The residues are obtained by solving the linear equations

$$\begin{bmatrix} p_1^{-1} & p_2^{-1} & \cdots & p_q^{-1} \\ p_1^{-2} & p_2^{-2} & \cdots & p_q^{-2} \\ .. & . & \cdots & . \\ .. & . & \cdots & . \\ p_1^{-q} & p_2^{-q} & \cdots & p_q^{-q} \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ .. \\ .. \\ k_q \end{bmatrix} = - \begin{bmatrix} m_0 \\ m_1 \\ .. \\ .. \\ m_{q-1} \end{bmatrix} \quad (31)$$

Note that, to get a q^{th} order approximation for the transfer function, 2q moment terms are needed.

6. EXPERIMENTAL RESULTS

To demonstrate the performance and accuracy of the algorithm, a variety of circuit models were tested. These circuits were randomly generated, so that they have random distribution of resistor loops, inductor links and floating nodes due to capacitors connected between non-grounded nodes in the tree. The experiments were performed in a Pentium 700 MHz machine with 128MB RAM under Linux operating system. The time (averaged for a group of 10 similar circuits of each category) required for computation of moments for some typical cases are represented in Table 1. As seen, the time required for computation increases linearly with the circuit size.

Circuit Size (No. of Nodes)	CPU Time (Seconds)
5000	0.31
10000	0.59
15000	1.02
20000	1.18
30000	1.85

Table 1. CPU time for moment computation of various circuits

Inductor Links	Resistor Loops	Capacitor Couplings	Run-Time (Seconds)
5000	0	0	0.56
0	5000	0	0.44
0	0	5000	0.47
2000	2000	2000	0.58
3000	3000	3000	0.59

Table 2. Comparison of Run-Times for circuits with same size but varying topologies

To examine that the efficiency of the algorithm is independent of the circuit topology, circuits with different complexities (in terms of the distribution of resistor loops, inductor links and floating nodes), but same size (10,000 elements) were generated and tested. The test results are shown in Table 2.

The accuracy of the method has been verified by determining the circuit responses from the moments and comparing the output with SPICE simulation. Circuits of 50 components were taken for comparison due to the large CPU and memory requirements for running a complete SPICE simulation.

7. REFERENCES

[1] L. T. Pillage and R. A. Rohrer, "Asymptotic Waveform Evaluation for Timing Analysis" *IEEE Trans. on CAD*, Vol 9, No 4, pp352-368, April 1990.

[2] C. L. Ratzlaff and L. T. Pillage, "RICE : Rapid Interconnect Circuit Evaluation Using AWE" *IEEE Trans on CAD of ICs & Ckts*, Vol 13, No 6, pp 763-776, June 1994.

[3] A. B. Kahng and S. Muddu, "Two pole Analysis of Interconnection Trees" *IEEE Trans on CAD of ICs & Ckts*, 1994

[4] B. Tutuianu, F. Dartu and L. T. Pileggi "An Explicit RC-Circuit Delay Approximation based on the first Three Moments of the Impulse responnse" *Proc. 33 DAC Conf* 1996