Abstract
We present a new realizable reduced order modeling technique for RLC interconnect trees. Both lumped and distributed wire models can be used with this technique. Provable stability is achieved by using Hurwitz polynomials. Moment computation process is avoided but moments can still be matched implicitly. In experiments, the proposed Hurwitz three-pole model can accurately and efficiently capture inductive effect for both near end and far end nodes.

1. Introduction
As technology progresses, interconnect delay plays the crucial role on the performance of high-speed circuits. Besides RC effects, inductive effects can not be ignored anymore for high performance design. Due to the huge size of interconnect circuits, reduced order modeling techniques\cite{1,2,10} are proposed to speedup timing analysis. Recently, many interconnect delay models were proposed based on moment matching techniques\cite{1-9}. It was well known that this kind of moment matching technique is equivalent to a Padé approximation, which may generate positive poles for a stable circuit, especially when inductance is considered\cite{3}. Congruence transformation techniques, such as PRIMA\cite{10}, can guarantee stability and even passivity. However, they are too general and complicated for interconnect delay calculation, particularly, when iterations of analysis are called for performance optimization.

Besides accuracy and stability, incremental and hierarchical capabilities are another two useful features for interconnect delay calculation. For example, incremental delay calculator can quickly update delay information in timing optimization (buffer/repeater insertion\cite{11}, transistor sizing and wire sizing), which involves many local changes. The hierarchical analysis capability can reduce redundant computation in timing analysis. For example, in a datapath structure, hundreds of bi-direction drivers exist in a same interconnect network. With a hierarchical analysis, the delay computation can be reduced greatly\cite{16}.

Recently, realizable reduction\cite{14,15} was addressed for RC circuits: using the first order realizable reduction, a large circuit can be reduced into a much smaller RC network which preserves the Elmore delay time constant. Realizable reduction allows hierarchical reduction and analysis. Furthermore, realizability is a sufficient condition for a system to be stable and passive.

In this paper, we propose a new RLC interconnect delay calculation method which is accurate, efficient and capable of hierarchical and incremental analysis. The main contributions are:
- Third order realizable reduction methods are proposed for RLC interconnect trees.
- Propose a new approach to preserve stability. We use lower order Hurwitz polynomials\cite{12} to approximate the denominators of original rational transfer functions. Stability can be guaranteed while moments are matched implicitly.
- We show that the Hurwitz stability can also be guaranteed during a hierarchical analysis. Besides, for long wire modeling, the time-of-flight can be captured without destroying the Hurwitz stability.

Both lumped and distributed wire models can be used with the above reduction method. In the following discussion, we only use lumped model for discussion.

This paper is organized as follows: In section 2, we first discuss the number of moments needed to be preserved to reflect inductive effect. In section 3, we discuss a hierarchical and incremental reduction procedure using an RLC Π model. In section 4, realizable RLC Π model reduction methods are presented. In section 5, we discuss how to preserve stability of transfer functions using Hurwitz polynomials. In section 6, we present some experimental results followed by conclusions.

2. Number of Moments to Reflect Inductive Effect
For efficiency, we hope delay models as simple as possible. However, a question is still remaining: to reflect inductive effect, how many driving-point admittance coefficients and circuit response moments should be preserved at least? Some delay models\cite{5-8} are based on matching the first two moments. In the following, we show that in general the first two moments are inadequate to capture inductive effect.

2.1 Driving-Point Admittance Coefficients
\[
Y(s)=y_1 s+y_2 s^2+y_3 s^3+... \quad \text{and} \quad Y'(s)=y_1' s^0+y_2' s^0+y_3' s^0+...
\]

This work was supported in part by grants from the NSF Project MIP-9529077 and the California MICRO program.

Hurwitz Stable Reduced Order Modeling for RLC Interconnect Trees
Xiaodong Yang\textsuperscript{1}, Chung-Kuan Cheng\textsuperscript{2}, Walter H. Ku\textsuperscript{1}, Robert J. Carragher\textsuperscript{3}
\textsuperscript{1}Dept. of E. C. E., U.C. San Diego, \textsuperscript{2}Dept. of C. S. E., U.C. San Diego, \textsuperscript{3}Fujitsu Laboratories of America
We use Fig. 1 to derive the coefficients of driving-point admittance \( Y(s) \). A lumped \([R, L, C]\) model between node \( i \) and node \( j \) represents a segment of wire. The load is an admittance \( Y_d(s) \). Initially, \( Y_d(s) \) equals to \( y_1^0 \) at a sink node, where \( y_1^0 \) equals to a load capacitance. The first three driving-point admittance coefficients of \( Y(s) \) are derived as below:

\[
\begin{align*}
    y_1 &= C + y_1^0; \\
    y_2 &= -Ry_1^0 + y_2^0; \\
    y_3 &= y_3 - 2y_2Ry_1^0 + R^2(y_1^0)^2 - L(y_1^0)^2; \\
\end{align*}
\]

(2.1)

The above three coefficients have the following properties:

**Property 1:**
- \( y_1 > 0; \ y_5 < 0; \)
- For an RC tree, \( y_3 > 0 \).
- For an RLC tree, the sign of \( y_3 \) depends on the value of inductance.
- Inductance only appears in \( y_3 \) with a negative sign.

**Property 2:**
- For an RC tree, the following inequality holds:
  \[
  y_1 \cdot y_3 > (y_2)^2
  \]
  (2.2)
- For an RLC tree, the condition of inequality Eq. (2.2) depends on the value of inductance.

According to properties 1 and 2, to reflect inductive effect, at least the first three driving-point admittance coefficients should be preserved.

### 2.2 Circuit Response Moments

The Elmore delay \(-m_j\) can not reflect the downstream resistance shielding effect. Similarly, \( m_2 \) can not reflect the effect of downstream inductance.

Moments can be computed by solving a series of DC circuits[11]; all capacitance are replaced by current sources and inductance are replaced by voltage sources. When we calculate \( m_2 \) of a node, only the inductance in the path from a driver node to that node will be considered.

![An interconnect tree](image)

**Fig. 2** An interconnect tree

For example, in Fig. 2, node \( D \) is a driver node and nodes \( 1, \ 2 \) and \( 3 \) are sink nodes. According to the above analysis, \( m_2 \) at node \( 1 \) considers only the inductance in path \( c \) while ignores all the inductance in path \( a \) and \( b \). As a special case, \( m_2 \) at node \( D \) ignores all downstream inductance. Thus, no matter how inductance in the other paths varies, the output response computed by matching only \( m_1 \) and \( m_2 \) will remain the same.

For the third order moment, inductance in all the paths will be considered. Therefore, to avoid accuracy loss due to inductive effect, at least the first three moments should be preserved.

### 3. Hierarchical Reduction with a Realizable RLC \( \Pi \) model

To perform a hierarchical and incremental analysis, an interconnect tree is reduced in a bottom-up fashion starting from sink nodes. At each node, its downstream RLC load is replaced with a simple equivalent realizable circuit (i.e. all RLC elements should be positive). According to the analysis in section 2, this circuit should be capable of preserving at least \( y_1, y_2, y_3 \) and \( m_1, m_2, m_3 \). For this purpose, we choose an RLC \( \Pi \) model as an equivalent circuit since it can be used for recursive reduction.

During the reduction, two cases need to be handled: series reduction and branch mergence.

**i) Series Reduction**

![Series Reduction](image)

**Fig. 3** Series Reduction

In Fig. 3, the lumped \([R, L, C]\) model between node \( i \) and node \( j \) represents a segment of wire. The load of node \( j \) is an RLC \( \Pi \) model \([k_\Pi R^{\Pi}, L^{\Pi}, C^{\Pi}, C^{\Pi}\Pi}\). The load seen at node \( i \) is replaced by a new realizable RLC \( \Pi \) model \([R^{\Pi}, L^{\Pi}, C^{\Pi}, C^{\Pi}\Pi}\) such that \( y_1 \), \( y_2 \) and \( y_3 \) are preserved.

**ii) Branch Mergence**

![Using one RLC \( \Pi \) model to replace two branches](image)

**Fig. 4** Using one RLC \( \Pi \) model to replace two branches

With this operation, at junction node \( j \), a new realizable RLC \( \Pi \) model \([R^{\Pi}, L^{\Pi}, C^{\Pi}, C^{\Pi}\Pi}\) is used to replace branches \( a \) and \( b \), which are represented as:

\[
\Pi_a = \{R_0^{\Pi,a}, L_0^{\Pi,a}, C_0^{\Pi,a}\}, \Pi_b = \{R_0^{\Pi,b}, L_0^{\Pi,b}, C_0^{\Pi,b}\}.
\]

Since downstream nodes are disappeared when a new circuit is used to replace downstream load, the transfer functions to sink nodes should be stored. During a bottom-up reduction, the transfer functions are propagated upstream as shown in Fig. 5.

![Transfer Function Propagation](image)

**Fig. 5** Transfer Function Propagation

---

1. It can be verified[16] that these properties hold for both lumped and distributed wire models.
Let \( H_{1j}(s) \), \( H_{2j}(s) \) be the transfer functions from node \( j \) to the sink nodes 1 and 2 respectively. In branch merge, the transfer functions stored at the junction node \( j \) remains the same. In series reduction, the transfer functions stored at node \( j \) are propagated to node \( i \) by multiplying \( H_{ij}(s) \) — the transfer function from node \( i \) to node \( j \), i.e.:

\[
\{ H_{1j}(s) = H_{ij}(s) \cdot H_{1i}(s) \}, \ H_{2j}(s) = H_{ij}(s) \cdot H_{2i}(s) \}.
\]

All these transfer functions are required to be stable and capable of reflecting inductive effect.

By recursively applying the above reductions and transfer function propagation in a bottom-up tree traversal, the interconnection tree is finally reduced into the circuit shown in Figure 6. The output response at node \( D \) can be computed using gate delay engines\(^{[16][17]}\). Meanwhile, since transfer functions to all sink nodes are available at node \( D \), the output response can be computed.

\[
\text{Fig 6. Final Reduced circuit}
\]

To implement the above procedures, the following problems should be solved for an RLC circuit:

i) How to replace downstream interconnect load with a guaranteed realizable RLC \( \Pi \) model which can be used to preserve \( y_1, y_2 \) and \( y_3 \)?

ii) How to construct a provably stable reduced order transfer function from a stable circuit while preserving \( m_1, m_2, m_3 \)?

iii) During the propagation of transfer functions, how to replace a product of several stable transfer functions with a provably stable reduced order transfer function while keeping the same order of accuracy?

4. Realizable RLC \( \Pi \) Model for Load Approximation

4.1 O’Brien and Savarino’s RC \( \Pi \) model

An RC tree \( Y(s) \)

\[
Y(s) = \frac{a_1 s + a_2 s^2 + a_3 s^3 + O(s^4)}{1 + b_1 s + b_2 s^2 + O(s^3)} = y_1 + y_2 s + y_3 s^2 + \ldots
\]  

(4.2)

For an RLC \( \Pi \) model, we can also write its driving-point admittance \( Y_\pi(s) \) using these two representations:

\[
Y_\pi(s) = \frac{s(C_{\pi1} + C_{\pi2}) + 2 s R_\pi C_{\pi1} C_{\pi2} + s^3 L_\pi C_{\pi1} C_{\pi2}}{1 + s R_\pi C_{\pi2} + s^2 L_\pi C_{\pi2}}
\]

(4.3)

Therefore, we may have two ways to derive \( \{ C_{\pi1}, R_\pi, L_\pi, C_{\pi2} \} \) by matching two different set of coefficients. However, we have four variables to be solved with either only three equations (series representation) or five equations (rational representation).

Method 1: Using series representation.

We introduce an extra equation by introducing a new coefficient \( y_3^* \), which is the third order admittance coefficient by ignoring all downstream inductance. For example, for a lumped RLC model (Eqs. (2.1)),

\[
y_3^* = y_3 + L(y_1)^2 - (y_3 - y_3^*)
\]

(4.4)

From properties 1 and 2, we can verify that the following inequalities are correct:

\[
y_3^* > y_3; \ \ y_3^* > \ 0; \ \ y_1 \cdot y_3^* > (y_2)^2
\]

(4.5)

By matching \( y_1, y_2, y_3, y_3^* \), we can derive \( \{ C_{\pi1}, R_\pi, L_\pi, C_{\pi2} \} \) as:

\[
C_{\pi1} = y_1 - (y_2)^2 / y_3^*; \ \ C_{\pi2} = (y_2)^2 / y_3^*;
\]

\[
R_\pi = -(y_3^*)^2 / (y_2)^3; \ \ L_\pi = (y_3^* - y_3) / C_{\pi2}^2;
\]

According to properties 1, 2 and Eq. (4.5), the RLC elements of this \( \Pi \) model are all positive and therefore it is realizable.

Method 2: Using rational function representation.

Without changing \( y_1, y_2, y_3 \), we modify Eq.(4.2) into:

\[
Y(s) = \frac{a_1 s + a_2 s^2}{1 + b_1 s + b_2 a_3 / a_1 s^2}
\]

(4.6)

Meanwhile, without changing the \( y_1, y_2 \) and \( y_3 \) of \( Y_\pi(s) \), Eq. (4.3) can be rewritten as follows:

\[
Y_\pi(s) = \frac{s(C_{\pi1} + C_{\pi2}) + s^2 R_\pi C_{\pi1} C_{\pi2}}{1 + s R_\pi C_{\pi2} + s^2 L_\pi C_{\pi2} / (C_{\pi1} + C_{\pi2})}.
\]

(4.7)

By matching the four denominator and numerator coefficients of Eq. (4.6) and Eq. (4.7), we can derive \( \{ C_{\pi1}, R_\pi, L_\pi, C_{\pi2} \} \) as:

\[
C_{\pi1} = a_2 / b_1; \ \ C_{\pi2} = a_1 - C_{\pi1} = a_1 - a_2 / b_1;
\]

\[
R_\pi = b_1 / C_{\pi2}; \ \ L_\pi = (a_1 b_2 - a_3 / C_{\pi2}^2;
\]

The realizability of this RLC \( \Pi \) model is guaranteed with the following properties.

Property 3. For both distributed and lumped models,
\[ a_i > 0, \quad b_i > 0, \quad i = 1, 2, 3, \ldots; \]
\[ b_1 - a_2/a_1 > 0; \quad b_2 - a_3/a_1 > 0; \]

5. Stable Reduced Order Transfer Function

For a linear system, its transfer function can be expressed as a rational function as below:
\[ H(s) = \frac{1 + a_1 s + a_2 s^2 + \ldots + a_m s^m}{1 + b_1 s + b_2 s^2 + b_3 s^3 + \ldots + b_n s^n} \]

Lemma 1: For a linear stable system, the denominator of its transfer functions must be a positive real polynomial\(^{12}\).

To reduce the order of a rational transfer function \(H(s)\), basically, the moment matching techniques such as AWE\(^1\) include the following two steps:

Step 1: Compute moments.
Step 2: Construct a rational function by moment matching.

In step 2, traditional moment matching methods can not guarantee the positive real polynomial property. Therefore, a reduced order transfer function may be unstable.

5.1 Preserve Stability Using Hurwitz Polynomial

Definition of a Hurwitz Polynomial\(^{12}\).

Given a polynomial function:
\[ f(x) = c_n x^n + c_{n-1} x^{n-1} + \ldots + c_1 x + c_0, \]
its odd part - even part ratio (or the even part - odd part ratio, if \(n\) is even), can be expressed as:
\[ \frac{q_1 x + \ldots + q_{m-1} x}{q_2 x + \ldots + q_n x} \]

If \(q_i > 0, \forall i\), we call \(f(x)\) a Hurwitz Polynomial.

Lemma 2 All the roots of a Hurwitz polynomial have negative real parts.

In Figure 3, if the load at node \(j\) is a realizable RLC \(n\) model, the transfer function from node \(i\) to \(j\) can be expressed as a rational function:
\[ H_{ij}(s) = \frac{1 + a_1 s + a_2 s^2}{1 + b_1 s + b_2 s^2 + b_3 s^3 + O(s^4)} \tag{5.1} \]

Note that the denominator of \(H_{ij}(s)\) are the same as shown in Eqs. (4.2), since \(Y(s)\) and \(H(s)\) share the same denominator.

Property 4.

With a realizable load, the denominator and numerator coefficients of a rational transfer function are all positive.

Since the rational transfer function of Eq. (5.1) is derived exactly from a stable system, all the poles must have negative real parts. Therefore, its denominator is a Hurwitz polynomial. However, its order is still too high to be used. For example, with a distributed line model, it has an infinite order. We are more interested in finding an approximation with a much lower order. The idea is to replace the original denominator with some lower order Hurwitz polynomials.

Property 5.
The coefficients in Eq. (5.1) fulfill the following inequalities:
\[ b_1 > a_1 > 0; \quad b_2 > a_2 > 0; \quad b_1 \cdot b_2 > b_3 > 0 \]

Theorem 1.

i) \(f(s) = 1 + (b_1 - a_1) s\) is a first order Hurwitz polynomial.

ii) \(f(s) = 1 + b_1 s + (b_2 - a_2) s^2\) is a 2nd order Hurwitz polynomial.

iii) \(f(s) = 1 + b_1 s + b_2 s^2 + b_3 s^3\) is a 3rd order Hurwitz polynomial.

Theorem 1 can be proven from property 5 and the definition of a Hurwitz polynomial. From Theorem 1, we can construct the following stable reduced order models.

i) H1P: stable one-pole model (preserve \(m_1\))
\[ H_{ij}(s) = \frac{1}{1 + (b_1 - a_1) s} \tag{5.2} \]

ii) H2P: stable two-pole model (preserve \(m_1, m_2\))
\[ H_{ij}(s) = \frac{1 + a_1 s}{1 + b_1 s + (b_2 - a_2) s^2} \tag{5.3} \]

iii) H3P: stable three-pole model (preserve \(m_1, m_2, m_3\))
\[ H_{ij}(s) = \frac{1 + a_1 s + a_2 s^2}{1 + b_1 s + b_2 s^2 + b_3 s^3} \tag{5.4} \]

In constructing above models, we never compute the moments. However, the moments are actually matched implicitly.

According to the discussion in section 2, during a hierarchical reduction, the two-pole model can not fully reflect inductive effect since only \(m_1, m_2\) are preserved.\(^2\) To reflect the inductive effect for all the nodes, the H3P model of Eq. (5.4) is preferred.

As we discussed in section 2, during a hierarchical reduction, we need to propagate transfer functions upstream. This operation causes the multiplication of several transfer functions.

Theorem 2 Given two \(k\)-th order Hurwitz stable transfer functions \(H_i(s), H_j(s)\), there exists a \(k\)-th order Hurwitz stable transfer function \(H(s)\) which preserves the first \(k\) moments of \(H_i(s) \cdot H_j(s)\).

In the following we use a third case as an example:

\[ H_i(s) = \frac{1 + c_1 s + c_2 s^2}{1 + d_1 s + d_2 s^2 + d_3 s^3 + O(s^4)}; \quad H_j(s) = \frac{1 + e_1 s + e_2 s^2}{1 + f_1 s + f_2 s^2 + f_3 s^3} \]

are both Hurwitz stable, \(H_i(s) \cdot H_j(s)\) can be approximated by Eq. (5.5), which is also Hurwitz stable and preserves the first three moments:
\[ H(s) = \frac{1 + a_1 s + a_2 s^2}{1 + b_1 s + b_2 s^2 + b_3 s^3} \tag{5.5} \]

Where
\[ a_1 = a_1 + a_1; \quad a_2 = a_2 + a_2 + a_1 a_1; \]
\[ b_1 = b_1 + b_1; \quad b_2 = b_2 + b_2 + b_1 b_1; \quad b_3 = b_3 + b_3 + b_1 b_1; \]

2. In Eq. (5.3), the impact of downstream inductance is actually cancelled by the term \((b_2-a_2)\).
where \( L_0, C_0 \) are the unit length inductance and capacitance respectively. \( \Gamma \) can be calculated during a tree traversal.

In such a situation, it is hard to approximate a rising edge using only a few poles (a few exponential functions in time domain). However, the accuracy could be improved if the time-of-flight delay can be extracted as follows:

\[
H(s) = \frac{1 + a_1s + a_2s^2}{1 + b_1s + b_2s^2 + b_3s^3} = e^{-\Gamma s} H\hat{(s)}
\]

This formulation corresponds to \( h(t) = h\hat{(t)}(t-\Gamma) \) in time domain.

In Eq. (5.6), \( H\hat{(s)} = \frac{1 + a_1\hat{s} + a_2\hat{s}^2}{1 + b_1\hat{s} + b_2\hat{s}^2 + b_3\hat{s}^3} \)

where, \( a_1\hat{s} = \Gamma + a_1; a_2\hat{s} = a_2 + a_1\hat{s} - \frac{1}{2}\Gamma^2 \)

Since we only modify the residues while keeping the poles unchanged, the stability will not be destroyed.

### 6. Experimental Results

#### Experiment 1. Realizable RLC \( \Pi \) model construction

The test circuit is a 3,000 \( \mu \)m long RC/RLC transmission line with 1pF load capacitance. The RLC \( \Pi \) models are constructed using both Method 1 and Method 2. Fig. 8 compares the near end response computed using a Hurwitz 3-pole model and SPICE. From this figure, for both RLC and RC wire, method 1 and method 2 match the results of SPICE quite well. For the RLC case, the H3P model captures the ringing effect in the rising edge.

#### Experiment 2. Hurwitz Two-pole model vs. Three-pole model

An RLC tree with 512 nodes is used in this experiment. We compare the results of both Hurwitz two-pole model and three-pole model with SPICE simulation.

1) Driver load node response .

In Figure 9, the two-pole model, which only matches the first two circuit response moments, can not reflect the inductive effect. Also we can see that its curve is almost identical to the result of SPICE without considering inductance in the whole circuit. The reason is that \( m_2 \) at this node does not have any information about the downstream inductance at all. Therefore, if we perform reduction for this node with only \( m_1, m_2 \), the whole circuit is just like an RC tree. However, with the three-pole model, the inductive effect in this case — a sharp peak in the rising edge, can be captured accurately. This experiment verifies our discussions in section 3: to reflect inductive effect, at least the first three moments should be matched.

2) Sink node response .

Fig. 10 shows the output response of a far end sink node. As we can see, the difference between the two-pole model and three-pole model is small. The reason is that the second order moment at far end has some information of the upstream inductance\(^3\). Additionally, there are less high frequency components existing in far end due to the low-pass feature of VLSI interconnect. It may be one of the reasons that some other delay models[5-8] based on matching the first two moments can also achieve good approximation sometimes.

\[
b_3 = b_1b_2b_3 + b_1b_2 + b_1b_3 + b_1 + b_3 - a_1a_2 - a_1a_3;
\]

It can be verified that above coefficients still fulfill property 5. Thus, the denominator of Eq.(5.5) is still a Hurwitz polynomial.
Experiment 3. Long wire modeling

A balanced RLC H-tree is used as the test circuit. The longest path of the test circuit is 8,000µm and the length of the longest segment is 4,000µm. Transmission line model is used for each wire. Fig. 11 shows the output response for a sink node. By capturing the time-of-flight delay, the accuracy for the rising edge is improved.

![Figure 11. Long wire modeling capturing / not capturing time-of-flight](image)

Experiment 4. Delay Comparison

The test circuit is an RLC tree with 258 nodes [18].

a. Compare with SPICE (rise time $t_{\text{rise}} = 10\text{ps}$).

Fig. 12 shows the output response computed with a Hurwitz 3-pole (H3P) model and the simulation result of SPICE. Table 1 shows the 50% and 90% delay comparison.

![Figure 12 Output response for different nodes (The input signal rise time is 10ps)](image)

<table>
<thead>
<tr>
<th>Node</th>
<th>SPICE 50% (ps)</th>
<th>SPICE 90% (ps)</th>
<th>RICE 50% (ps)</th>
<th>RICE 90% (ps)</th>
<th>H3P 50% (ps)</th>
<th>H3P 90% (ps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>106.9</td>
<td>727.7</td>
<td>727.7</td>
<td>0.03</td>
<td>107.0</td>
<td>727.5</td>
</tr>
<tr>
<td>2</td>
<td>236.3</td>
<td>857.0</td>
<td>858.7</td>
<td>0.21</td>
<td>236.8</td>
<td>858.7</td>
</tr>
<tr>
<td>3</td>
<td>260.6</td>
<td>883.7</td>
<td>885.4</td>
<td>0.19</td>
<td>261.6</td>
<td>885.4</td>
</tr>
<tr>
<td>4</td>
<td>274.2</td>
<td>895.4</td>
<td>898.1</td>
<td>0.30</td>
<td>274.8</td>
<td>898.1</td>
</tr>
<tr>
<td>5</td>
<td>280.7</td>
<td>902.0</td>
<td>904.3</td>
<td>0.25</td>
<td>282.8</td>
<td>904.3</td>
</tr>
</tbody>
</table>

![Table 1: 50%, 90% Delay Comparison (tr=10ps): SPICE, H3P](image)

Table 2: 50%, 90% Delay Comparison ($t_{\text{rise}} = 0$): SPICE, RICE, H3P

<table>
<thead>
<tr>
<th>Node</th>
<th>SPICE 50% (ps)</th>
<th>SPICE 90% (ps)</th>
<th>RICE 50% (ps)</th>
<th>RICE 90% (ps)</th>
<th>H3P 50% (ps)</th>
<th>H3P 90% (ps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>96.4</td>
<td>717.8</td>
<td>95.9</td>
<td>717.6</td>
<td>96.3</td>
<td>717.6</td>
</tr>
<tr>
<td>2</td>
<td>225.3</td>
<td>847.0</td>
<td>225.2</td>
<td>846.8</td>
<td>226.5</td>
<td>848.5</td>
</tr>
<tr>
<td>3</td>
<td>251.6</td>
<td>872.8</td>
<td>251.5</td>
<td>873.1</td>
<td>250.0</td>
<td>873.9</td>
</tr>
<tr>
<td>4</td>
<td>264.1</td>
<td>885.6</td>
<td>263.9</td>
<td>884.7</td>
<td>263.1</td>
<td>885.9</td>
</tr>
<tr>
<td>5</td>
<td>270.5</td>
<td>892.0</td>
<td>270.3</td>
<td>891.9</td>
<td>271.0</td>
<td>891.2</td>
</tr>
</tbody>
</table>

![Figure 13 Magnified curves: SPICE, RICE (6 moments), Hurwitz 3-pole, Hurwitz 2-pole](image)

According to Table 2, in general the accuracy of Hurwitz three-pole model is comparable with RICE for this test case. In Fig. 13 the output response of node 1 is magnified to show the difference. From this figure, we can identify the difference among the four different methods. In this particular case, Hurwitz three-pole model is better than RICE, which is even worse than the Hurwitz two-pole model. However, RICE quickly catches the rest of slow-varying rising edge. The 50% delay and 90% delay, as seen in Table 2, are still close to SPICE.

7. Conclusions

In this paper, we introduced a new realizable and stable reduced order modeling technique for an RLC/RC interconnect tree. Realizability is united for a general RLC $\Pi$ model. Hurwitz polynomials are used to construct stable reduced order models. Although moments are not computed during the process, they can see from the figure and table, the accuracy of H3P is good for both near end and far end node.
still be matched implicitly. The Hurwitz stability is preserved during the hierarchical reduction. It is shown that the Hurwitz stable three-pole model, which matches the first three circuit response moments, can efficiently and accurately capture the inductive effect for both near end and far end nodes. Our technique is also useful for interconnect timing analysis and optimization procedures due to its incremental and hierarchical features.

8. References

[18] Personnel Communication with Frank Liu in Synopsys