Noise-aware Repeater Insertion and Wire Sizing for On-chip Interconnect Using Hierarchical Moment-Matching

Chung-Ping Chen and Noel Menezes
Strategic CAD Labs, Design Technology
Intel Corporation
Hillsboro, OR 97124

Abstract

Recently, several algorithms for interconnect optimization via repeater insertion and wire sizing have appeared based on the Elmore delay model. Using the Devgan noise metric [6] a noise-aware repeater insertion technique has also been proposed recently. Recognizing the conservatism of these delay and noise models, we propose a moment-matching based technique to interconnect optimization that allows for much higher accuracy while preserving the hierarchical nature of Elmore-delay-based techniques. We also present a novel approach to noise computation that accurately captures the effect of several attackers in linear time with respect to the number of attackers and wire segments. Our practical experiments with industrial nets indicate that the corresponding reduction in error afforded by these more accurate models justifies this increase in runtime for aggressive designs which is our targeted domain. Our algorithm yields delay and noise estimates within 5% of circuit simulation results.

1 Introduction

Repeater insertion is widely recognized as the most effective technique for delay and transition time improvement in RC nets. Essentially, the delay of an unbuffered line scales quadratically with its length. Repeater insertion makes this relationship a linear one which greatly alleviates the interconnect problem of current technologies. Another equally advantageous but frequently ignored effect of repeater insertion in large RC nets is the signal restoration that CMOS repeaters provide due to their high-gain amplifier nature. For a quiet net if the resistance of the net is much higher than the driver resistance, the receiver is effectively decoupled from the driver which makes the receiver highly susceptible to any attacking signal on the net. Inserting a repeater in such a net makes the net more immune to noise since the repeater because of its high gain works so as to preserve the correct value of the signal on the net in the presence of any attacking signal. In several cases during design even when the delay of a net is acceptable, the noise at the receivers may necessitate repeater insertion. Finally, repeaters vastly improve the transition time of the signals at the receiver as well. Consequently, it is imperative that any interconnect optimization solution must take the delay, transition times, and noise at the receivers into account.

Recently in [5] a repeater insertion technique for noise avoidance based on the Devgan noise metric [6] was presented. This technique based on the Ginneken/Lillis dynamic-programming algorithm [3, 4] exploits the hierarchical nature of the Elmore delay model [1, 2] and the Devgan noise metric [6] to construct a set of feasible solutions in a bottom-up manner. A main drawback of this technique is its use of conservative delay and noise models. The pessimism of the noise metric of [6] was recently described in [10] which illustrated some cases where Devgan's noise metric overestimates the noise by as much as 100%. A careful look in the results section of [5] shows that the usage of the Devgan noise metric caused nets without any noise problems to be flagged as noise violations which is a direct result of the conservatism of this metric. Another disadvantage of the Elmore delay model is its inability to accurately estimate transition times at the sink. The authors of the repeater insertion technique described in [7] recognize this by handling transition time constraints through the use of an empirical table-lookup method for correlating the Elmore delay to the transition time. Furthermore, the simplistic driver models employed by all these techniques do not account for the resistive shielding nature of RC nets [13]: the driver delay for an RC net is significantly smaller than that predicted using the total capacitance of the net due to the resistance in the load. The authors of [8] apply a rule-based approach to noise-aware repeater insertion. These rules are derived from empirical circuit studies performed using SPICE simulations.

The Elmore delay model and the noise metric of [6] are computed in linear time using a hierarchical bottom-up computation algorithm which make these metrics amenable to interconnect optimization algorithms. In this paper we present efficient techniques for removing this conservatism in delay, transition-time, and noise estimation in interconnect optimization through the usage of more accurate moment-matching techniques like Asymptotic Waveform Evaluation (AWE) [14] and Pade-via-Lanczos (PVL) [11] for a reasonable increase in computation time. Our basic algorithm follows the Ginneken/Lillis dynamic programming framework. The main contributions of this paper are:

- A new hierarchical and accurate noise estimation algorithm is developed for handling receiver noise constraints. Unlike the Devgan noise model we can handle arbirtrarily shifted attacking noise waveforms.
- In addition to delay, transition time constraints which are imperative for interconnect optimization are handled.
- The effect of the input transition time on the repeater delay and transition time is taken into account.
- Moment-matching techniques are used for accurate RC delay estimation.
- An accurate repeater/driver model [12, 13] that takes resistance-shielding of the RC net is into account in addition to pro-

viding accurate waveform estimates at the sinks is used.
The techniques presented here have been partially presented elsewhere [9] but we describe them again for completeness. To the best of our knowledge this is the first work in interconnect optimization that handles delay, transition-time, and noise constraints in a single framework by the use of advanced delay models for both the linear RC as well as the driver/repeater components of on-chip nets.

2 The Ginneken/Lillis algorithm

We use the framework described in [3, 4] for our repeater insertion strategy. (We omit an algorithm description for brevity. For this, we direct the interested reader to [3, 4, 9].) For an RC network as shown in Figure 1, the algorithms of [3, 4] starts at the
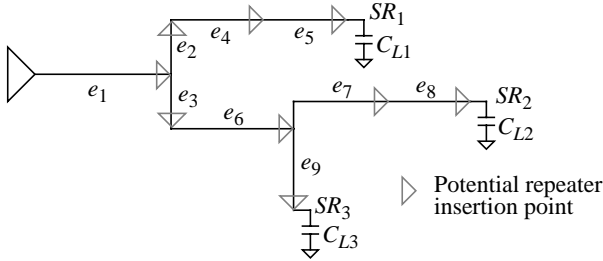
Figure 1: A routing tree with potential repeater location at the non-root-oriented end of each branch

receivers, SR_i, and inserts repeaters from the set of allowable repeaters B in turn at the end point of each routing segment e_i of the tree. The Ginneken algorithm uses dynamic programming in that at each insertion point a minimal set of solutions is constructed in terms of the set of solutions at the insertion points immediately downstream of the insertion point. In the Elmore delay formulation employed by [3, 4], each member of the solution set is characterized by a *pair* (c, t) which corresponds to a partial solution for the repeater insertion problem up to this point. The quantity c corresponds to the input capacitance for this partial solution seen from the insertion point while the quantity t represents the required time at this insertion point. The solution set of pairs at an insertion point contains all the information required to propagate a solution backwards up to the driver.

In order to handle noise, delay, and transition time more accurately during backward propagation of the solution sets we would like to capture the following information at an insertion node i: a) the required time at this node, b) the allowed noise at this node, c) the effect of transition time of the signal at this node on the required time. Figure 2 shows an example that depicts one pair[1] of a solution set at node j. For the example in Figure 2 we want to capture the required time at node j with respect to the worst-case receiver SR_i as accurately as possible, the effect of the transition time of a signal at node j on the required time, and the noise that would be generated at SR_i. Furthermore, we also want to capture information in the pair data structure that would allow us to generate the pairs for the solution set at node i immediately upstream at node j during the two fundamental backward propagation steps: (i) backward propagation through wire segment e_i, and (ii) backward propagation through a repeater inserted at node j.

---

[1]We continue to use the word "pair" to describe the data structure which represents a valid solution to the repeater insertion problem for the subtree rooted at an insertion node.
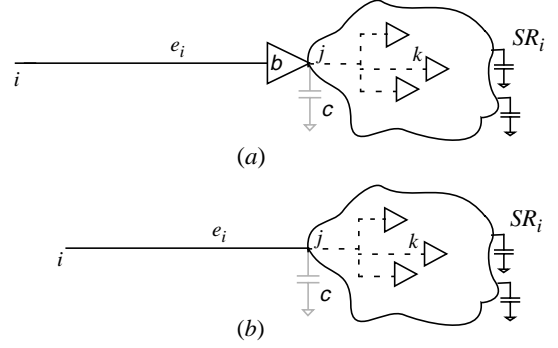
Figure 2: Backward propagation of pairs: a) across repeater b inserted at node j, b) across wire e_i (no repeater inserted).

3 Required time computation

We first explain how the required time is computed for backward propagation through a wire:

3.1 Hierarchical moment computation for accurate RC delay computation

In calculating the required time at node i for the case when no repeater is inserted at node j (Figure 2b), the Ginneken algorithm makes the assumption that the required time can be computed by subtracting the RC delay of segment e_i from the required time at node j. In reality, this RC delay depends on the signal waveshape at node i and the load presented by the partial RC subtree represented by the dotted lines of Figure 2. (The partial RC subtree for a pair is the part of the subtree from the node under consideration to the first downstream repeaters/receivers.) We instead compute the delay from node i to the input node k of the first repeater along the path to the worst-case receiver. In our approach, instead of hierarchically computing the delays, we hierarchically compute the transfer functions between nodes i and j, and nodes j and k. The delay, t_{ik}, is then computed by convolution of the input signal waveshape with the composite transfer function up to node k. Effectively, the required time at node i is not computed with respect to the required time at node j but with respect to the required time at node k. This also allows us to take into account the loading effect of the partial RC subtree of the pair accurately.

This is done by using moment-matching techniques like Asymptotic Waveform Evaluation (AWE) [14] which work on the principle of matching the first $2q$ moments of the transfer function from the root to any node in an RC tree:

$$H(s) = m_0 + m_1 s + m_2 s^2 + \ldots + m_{2q-1} s^{2q-1} \tag{1}$$

to that of a reduced-order pole-and-residue representation:

$$\tilde{H}(s) = \sum_{i=1}^{q} \frac{\tilde{k}_i}{s - \tilde{p}_i}. \tag{2}$$

A 3rd-order ($q = 3$) approximation is usually sufficient for capturing a reasonably accurate response for RC nets with thousands of capacitors!

In our approach during backward propagation of a pair at node j along wire e_i, in order to calculate t_{ik}, we compute the moments of the transfer function, $H_{ik}(s)$, from the electrical parameters of the branch e_i and from the moments of $H_{jk}(s)$ and admittance $Y_j(s)$ of the partial RC subtree which are stored in the pair under consideration at node j. The electrical model for the

computation of $H_{jk}(s)$ and $Y_i(s)$ are shown in Figure 3b for which the formulae are:

$$H_{ij}(s) = \frac{1}{R(Cs + Y_j(s)) + 1},$$

$$Y_i(s) = Cs + \frac{Cs + Y_j(s)}{R(Cs + Y_j(s)) + 1}, \quad (3)$$

$$H_{ik}(s) = H_{ij}(s)H_{jk}(s),$$

where $R = R_i$ and $C = C_i/2$. The RC delay t_{ik} is then computed by convolution of the waveform at i and $H_{ik}(s)$ (Figure 3). The
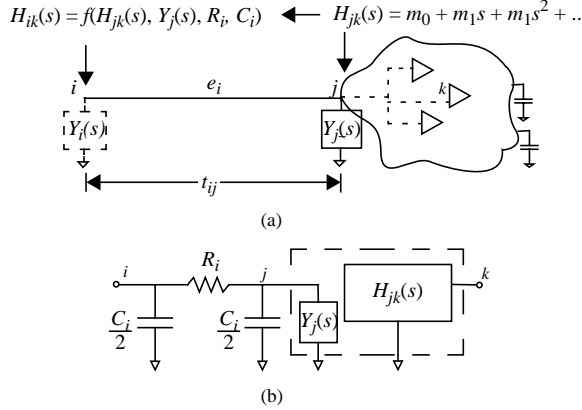

(a)

(b)

Figure 3: Hierarchical moment computation a) transfer function and admittance propagation, b) equivalent electrical model.

moments of $H_{ik}(s)$ and $Y_i(s)$ are then stored in the pair at node i. The moments of $Y_i(s)$ are also used in computing the repeater delay for the case where a repeater is inserted at node i.

The hierarchical moment generation for the transfer function and input admittance always starts from either a receiver or a repeater. For this base case, if c represents the receiver/repeater input capacitance, the moment representation of the transfer function and admittance is given by:

$$\begin{aligned} H(s) &= 1 \\ Y(s) &= cs \end{aligned}. \quad (4)$$

The transfer functions and admittance are always represented in the moment form of (1) in the pair data structure.

Wire sizing can be handled during this step by backward propagating the pairs from node j to node i for different wire widths of segment e_i. In this case, R_i and C_i are functions of the wire width.

3.2 Accurate driver/repeater modeling

When we need to consider the backward propagation of required time in the case where a repeater is inserted at node j (Figure 2a), we use the delay model of [12] which provides accurate driver delay estimates in the presence of RC loads. The inputs to this driver model are a repeater precharacterized in terms of the input transition time, t_{in}, and output load, C_L, and, the admittance $Y(s)$ of the RC net load. (This is the same $Y_j(s)$ of Figure 3.) The model of [12] computes the delay from the repeater input to the worst-case repeater directly using $H_{jk}(s)$ and $Y_j(s)$. As in the previous subsection, the required time for the case where a repeater is inserted at node j is obtained by subtract-

ing the driver-RC delay computed by [12] from the required time at node k.

4 Backward noise propagation

In the previous section, we described how the required time was backward propagated through wire segments and repeaters. Rather than computing delays on a segment or repeater basis, all delays are computed from the node of interest to the first downstream receiver/repeater along the worst-case path. We apply this same principle for propagating noise constraints backwards. We assume that a noise constraint expressed as a dc noise level, N_{SRi}, is specified for every receiver, SR_i. During backward propagation of a pair along a wire or a repeater, the noise is directly computed at the worst-case receiver/repeater of the partial RC subtree represented by the pair. If the computed noise exceeds the noise constraint for this receiver/repeater, then this pair represents an invalid solution to the repeater insertion problem from a noise point-of-view. If not, this pair is added to the solution set.

4.1 Receiver noise propagation through a wire segment

We apply a novel hierarchical approach to accurate noise waveform computation at the receiver/repeater input. The driver model we use for noise is similar to that of [15] where the driver is modeled as a grounded resistor. The attackers are modeled as saturated voltage ramps as shown in Figure 4. We first illustrate the principles of our approach.

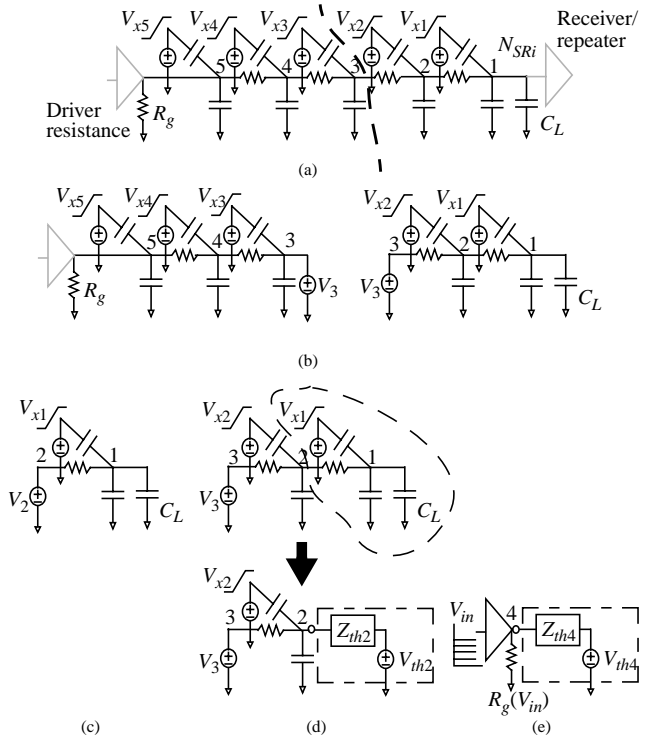The noise waveform at the receiver of Figure 4a can be

Figure 4: Backward propagation for noise computation, a) circuit for noise modeling, b) circuit decomposition, c) subcircuit for first segment, d) subcircuit for second segment which includes Thevenin equivalent of first subcircuit, e) computing repeater output voltage for node 4.

computed using the principles of linear superposition and decomposition: In the circuit of Figure 4a if the voltage at any node along the line is known then the part of the circuit from the driver up to that node can be replaced with a voltage source of that value (decomposition principle). This is illustrated in Figure 4b where the circuit is decomposed at node 3 by placing a voltage source of value V_3 corresponding to the voltage at node 3 in the original circuit at the equivalent node in the subcircuit. Furthermore, due to the linear nature of the subcircuit the noise waveform at the sink is the summation of the responses due to each of the attacker sources from this subcircuit and the newly introduced voltage source (linear superposition principle). With this in mind we now outline our approach to backward propagation of the receiver noise waveform through a wire segment.

For the circuit of Figure 4c, if the voltage at node 2, V_2, is known then the voltage at node 1 (the receiver) is given by

$$V_1 = V_2 H_{21} + V_{x1} H_{x1}. \tag{5}$$

(We use H_{**} to represent the appropriate transfer functions of the subcircuit under consideration.) Now, we consider the effect of the addition of the segment between node 2 and node 3 to V_1. In order to do this we replace the subcircuit of Figure 4c by its Thevenin equivalent as illustrated in Figure 4d. The voltage V_2 can be now be expressed as

$$V_2 = V_3 H_{32} + (V_{th1} H_{th1} + V_{x2} H_{x2}). \tag{6}$$

We can now substitute the value of V_2 from (6) in (5) to obtain the receiver noise waveform in terms of V_3:

$$V_1 = V_3(H_{32}H_{21}) + (V_{x1}H_{x1} + V_{x2}H_{x2}H_{21} + V_{th1}H_{th1}H_{21}) \tag{7}$$

Now in order to backward propagate to node 4 we replace this circuit by its Thevenin equivalent and compute V_3 in terms of V_4 which can be substituted in (7) to obtain the receiver noise waveform in terms of V_4.

Hence, at any given stage of this hierarchical computation we represent the receiver noise waveform V_1 in terms of a constant waveform (the term in the second parentheses of (7)) which represents the contribution of the attacker voltage sources encountered so far during backward propagation through wire segments and a transfer function multiplier (the term in the first parentheses of (7)) which represents the contribution of voltage sources yet to be encountered. Both the attacker voltages as well as the transfer functions are stored in moment form which makes for a very efficient computation.

An interesting problem here is the relative time shifts of the attacker voltage sources in order to generate the worst possible noise at the receiver. Through extensive experimentation we have determined that aligning all the attacker ramps so that they terminate at the same time point provides a reasonable estimate of the worst-case noise at the receiver for current process technologies.

4.2 Backward propagation of receiver noise constraints through repeaters

When a repeater is inserted at a node, we need to determine the maximum dc noise level at the repeater input which will not violate the dc noise constraint at the worst-case downstream repeater/receiver. This maximum input dc noise level will serve as the noise constraint for propagation from the inserted repeater backwards. To evaluate the effect of a repeater insertion at a particular node, for example at node 4 of Figure 4, we simply simulate the repeater and the Thevenin equivalent of the propagated subcircuit at node 4 to obtain the repeater output voltage V_4 (shown in

Figure 4e). Now the receiver noise waveform V_1 can be readily computed in terms of V_4 by an expression for node 4 similar to (7). This simulation is carried out for different input dc noise levels, V_{in}. The maximum input noise level for which the peak of the computed receiver voltage V_1 equals the receiver dc noise constraint is the noise constraint at the input of this repeater.

The simulation for the circuit of Figure 4e can be carried by a circuit simulator with the actual non-linear devices of the repeater included in the simulation. However, for efficiency, as stated earlier, we use a grounded resistor to model the repeater for noise purposes. The value of this grounded resistor is a (nonlinearly increasing) function of the input dc noise level V_{in} and is determined by empirical studies. This resistance modeling of the repeater converts the circuit to a linear one which is efficiently analyzed using the moment matching techniques outlined earlier.

5 Transition-time effects

The delay and output transition time of a repeater, and consequently the delay of the RC net that it drives, is a strong function of the transition time of the input signal. Hence, in Figure 2a, while inserting a repeater at node j, we take the transition time of the input signal into account. This transition time, however, is unknown because of the bottom-up nature of this algorithm. Hence, we perform this analysis for different input transition times. The required time at node j is then described in the pair as a function of the input transition times. Furthermore, in subsection 3.1 during backward propagation along a wire, the RC delay computation from node i to node k is carried out for different input transition times at node i. Again, the required time at node i is stored in the pair as a function of the input transition times.

A detailed description of handling transition-time effects is given in [9]. We assert here that a repeater insertion algorithm will suffer from significant accuracy problems if transition-time effects are not taken into account.

6 Ginneken's pruning lemma

An important aspect of Ginneken's work [4] is the effective pruning mechanism applied to solution sets which states that:
> For any two pairs, (c, t) and (c', t'), in a solution set S, if $c < c'$, and $t > t'$, then the pair (c', t') can be removed from set S.

We have extended this pruning technique to take noise and transition time into account as well. When two pairs of a solution set are compared in order to eliminate the obviously non-superior one, transition time and noise at the worst-case receiver are two additional metrics applied to this comparison. This pruning technique drastically reduces the solution set size at each insertion point.

7 Experimental results

We have run our implementation of the algorithm described above on thousands of nets taken from a previous-generation microprocessor. These nets had a demanding transition time requirement in addition to the user-specified delay requirement. We use a library of inverters of different sizes as repeaters. A comparison of our algorithm for a set of nets that were manually designed by circuit designers for the same delay and transition time constraints is shown in Figure 5. We can see that our algorithm was able to optimize the delay of these nets to a value very close to those achieved by manual design. More importantly, in all
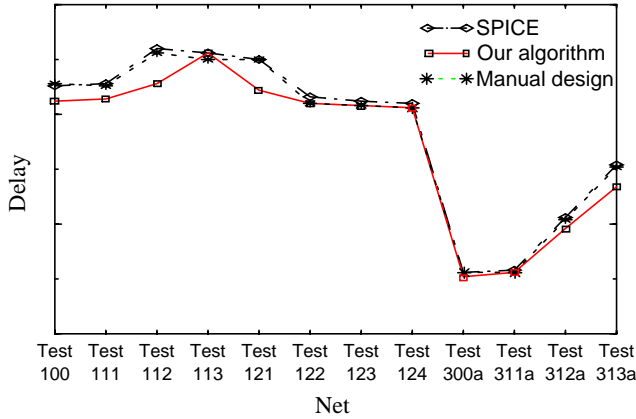
Figure 5: Comparison of nets optimized using our algorithm against the manually designed nets. (The delay axis units have been intentionally omitted.)

these case the transition time requirements was met with a <5% error. Also shown are the delays predicted by our algorithm compared to the delay predicted by circuit simulation (SPICE) for the optimized nets.

In Figure 6 we show the effect of imposing noise constraints by performing repeater insertion on a 10000 μm line that has been divided into 500 μm segments. In Figure 6a the results of repeater insertion with delay and transition time considerations only show a 3 repeater solution in which the last unbuffered segment is 3.5 mm long. Imposing a noise constraint at the receiver of a third of the supply voltage tends reduce the length of the last segment to 3mm in order to meet the noise requirement at the receiver. In Figure 6c we see that lowering the tolerable receiver noise to a fifth of the supply voltage increases the number of inserted repeaters to five which is expected. More importantly,
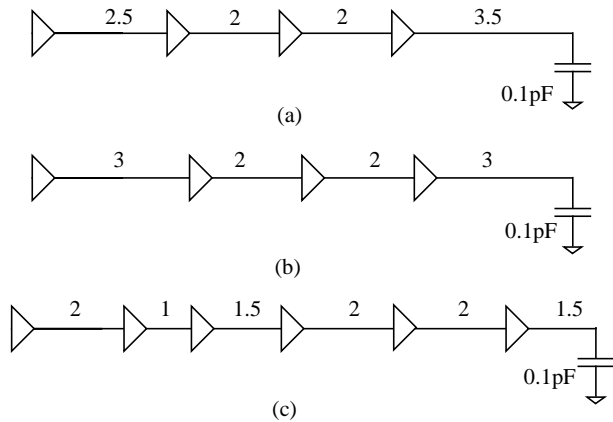

Figure 6: Repeater insertion for a 10000 μm line (a) no noise constraint, (b) noise constraint of $V_{DD}/3$, (c) noise constraint of $V_{DD}/5$. (All lengths are in mm.)

when simulated the post-repeater-insertion nets exhibited a receiver noise within 10mV of the specified noise constraint for this particular example. Thus, we remove much of the pessimism associated with other noise models. The runtime required for optimizing the net was on the order of 16 seconds on an IBM RS6000 workstation.

8 Conclusion

We have presented an approach to repeater insertion and wire sizing that is based on the optimal Lillis/Ginneken dynamic programming framework. However, using state-of-the-art delay modeling techniques and an elegant hierarchical approach to traditional moment-matching techniques like Asymptotic Waveform Evaluation, we have reasonably incorporated important noise and transition-time constraints into the repeater insertion framework. Furthermore, our comparison with SPICE shows that our algorithm produces solutions that are within 5% of the specified delay constraints.

References

[1]. W. C. Elmore, "The transient response of damped linear networks with particular regard to wideband amplifiers," Journal of Applied Physics, vol. 19, no. 1, 1948.

[2] P. Penfield and J. Rubinstein, "Signal delay in RC tree networks," *IEEE Trans. Computer-Aided Design,* vol. CAD-2, pp. 202-211, July 1983.

[3] J. Lillis. C.-K. Cheng, and T.-T. Lin, "Optimal and efficient buffer insertion and wire sizing," *Proc. Custom Integrated Circuits Conference,* pp. 259–262, May 1995.

[4] L. P. P. P. van Ginneken, "Buffer placement in distributed RC-tree networks for minimal Elmore Delay," *Proc. International Symposium on Circuits and Systems,* pp. 865-868, 1990.

[5] C. J. Alpert, A. Devgan, and S. T. Quay, "Buffer insertion for noise and delay optimization," *Proc. 35th ACM/IEEE Design Automation Conferenc*e, pp. 362-367, June 1997.

[6] A. Devgan, "Efficient noise coupled noise estimation for on-chip interconnects," *Proc. of the Intl. Conf. on Computer-Aided Design*, pp. 147-151, Nov. 1997.

[7] J. Culetu, C. Amir, and J. MacDonald, "A practical repeater insertion method in high speed VLSI circuits," *Proc. 35th ACM/ IEEE Design Automation Conferenc*e, pp. 392-395, June 1997.

[8] D. Li, A. Pua, P. Srivastava, and U. Ko, "A repeater optimization methodology for deep sub-micron, high-performance processors," *Proc. of the Intl. Conf. on Computer Design*, pp. 726-731, Oct. 1997.

[9] N. Menezes and C.-P. Chen, "Spec-based repeater insertion and wire sizing for on-chip interconnect," *Proc. of the 12th Intl. Conf. on VLSI Design,* pp. 476-483, Jan. 1999.

[10] K. Rahmat, J. Neves, and J.-F. Lee, "Methods for calculating coupling noise in early design: a comparative analysis," *Proc. of the Intl. Conf. Computer Design*, pp. 76-81, Oct. 1998.

[11] P. Feldman and R.W. Freund, "Efficient linear circuit analysis by Pade approximation via the Lanczos process," *IEEE Trans. Computer-Aided Design.*, vol. 14, no. 5, pp. 639-649, May 1995.

[12] F. Dartu, N. Menezes, J. Qian, and L.T. Pillage, "A gate-delay model for high-speed CMOS circuits," *Proc. 31st ACM/IEEE Design Automation Conferenc*e, pp. 576–580, June 1994.

[13] J. Qian, S. Pullela and L. T. Pillage, "Modeling the *effective capacitance* for the RC interconnect of CMOS gates," *IEEE Trans. Computer-Aided Design.*, vol. 13, no. 12, pp. 1526-1535, Dec. 1994.

[14] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design,* vol. 9, no. 4, pp. 352-366, April 1990.

[15] K. L. Shepard, *et al*. "Global harmony: coupled noise analysis for full-chip RC interconnect networks," *Proc. of the Intl. Conf. on Computer-Aided Design*, pp. 139-146, Nov. 1997.

[16] A. B. Kahng, and S. Muddu, "New efficient algorithms for computing effective capacitance," *Proc. of the 1998 Intl. Symposium on Physical Design,* pp. 147-151, April 1998.