

# Reliable Non-Zero Skew Clock Trees Using Wire Width Optimization

Satyamurthy Pullela, Noel Menezes, and Lawrence T. Pillage  
Department of Electrical and Computer Engineering  
The University of Texas at Austin  
Austin, TX 78712

## Abstract

Recognizing that routing constraints and process variations make non-zero skew inevitable, this paper describes a novel methodology for constructing reliable low-skew clock trees. The algorithm efficiently calculates clock-tree delay sensitivities to achieve a target delay and a target skew. Moreover, the sensitivities also show that wires should be widened as opposed to lengthened to reduce skew since the former improves reliability while the latter reduces it. This paper introduces the concept of designing *reliable* clock nets with process-insensitive skew.

## 1 Introduction

Clock skew is defined as the maximum difference in the delays from the output of the clock buffer to the inputs of the clocked elements on a chip. With the increasing density of VLSI and the use of several pipeline stages in hardware design, clock skew becomes a dominating factor in determining the clock period of synchronous digital systems.

While reducing clock skew should be the main focus of clock net algorithms, clock-signal delay should be considered as well since it affects system-level skew[5]. Moreover, for reliable results, it is also important to consider that the actual width of a line on a fabricated chip may differ from the expected width due to intra-chip process variations such as over- and under-etching, mask misalignment, spot defects, etc. This paper shows that these wire width process variations can significantly impact the delay and skew.

Recent algorithms [4, 6] proposed for clock-skew reduction construct binary tree-like structures with the clock pins at the leaf nodes. "Zero-skew" trees are created in a recursive bottom-up manner from the clock pins upwards by tapping the connection between two zero-skew subtrees at such a location so as to create a parent zero-skew tree rooted at the tapping point. Earlier algorithms [5, 8] assume that the wire length is a valid measure of the delay and try to equalize the lengths from the root of the clock tree to the leaf nodes. These algorithms, in effect, minimize skew while routing the clock net. Our algorithm, on the

other hand, routes the tree first and then minimizes skew by varying the wire widths of the tree branches. This gives the router additional flexibility in routing around possible blockages. Since the algorithms described in [5, 8] can easily be modified to route around blockages, they serve as excellent starting points for the algorithm described here.

Our approach uses *sensitivities* to address the most important issues in clock tree synthesis. Firstly, skew and delay are optimized using sensitivity information. In addition, sensitivities are used to increase the reliability of synthesized clock nets. The sensitivity approach allows us to specify which wire lengths and/or widths to vary for clock tree synthesis. This paper, however, uses the latter approach since widening is shown to increase reliability (decrease delay sensitivity to process variation) while lengthening is shown decrease it.

Our paper is organized as follows: In Section 2, we detail a simple RC delay model [2] and the inaccuracies associated with using it. The futility of "zero-skew" clock nets is also introduced along with the concept of *reliable* clock nets which are insensitive to process variation. In section 3, we propose an efficient algorithm to calculate the sensitivities of the leaf-node signal delays to variations in the width of the tree branches. Techniques for delay reduction, reliability, and skew reduction based on these sensitivities are described in section 4. We then present results, conclusions, and possible extensions of our approach.

## 2 Practical Considerations

Each branch of a clock tree can be represented by a distributed resistance-capacitance segment. Each distributed RC line is modeled as an equivalent lumped L- or  $\pi$ -circuit. The lumped resistance of the L-circuit model of an RC line of length  $l_i$  is usually approximated by  $rl_i$  and the capacitance by  $cl_i$  where  $r$  and  $c$  are the resistance and capacitance per unit length. Figure 1 shows a route for eight clock pins. The equivalent RC tree with every segment replaced by an L-model is shown in Figure 2.

We present our notation first: The branch connecting node  $i$  to its parent node in the RC tree is labeled  $i$ , and the width of this branch is denoted by  $w_i$ . The resistance of branch  $i$  is denoted by  $R_i$ . The sum of the resistances of the branches on the path from node  $i$  to the root is called the upstream resistance,  $R_{u,i}$ .  $R_{c,i}$ , is used to refer to the sum of the resistances along the

\*This work was supported in part by IBM Corp., the National Science Foundation, and the Semiconductor Research Corporation under contract # 92-DP-142.

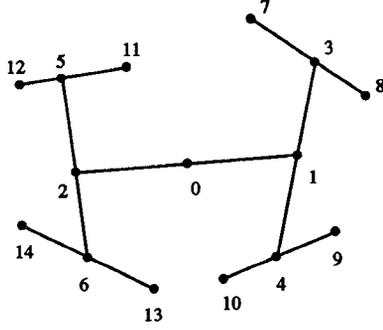


Figure 1: Example clock route on 8 pins.

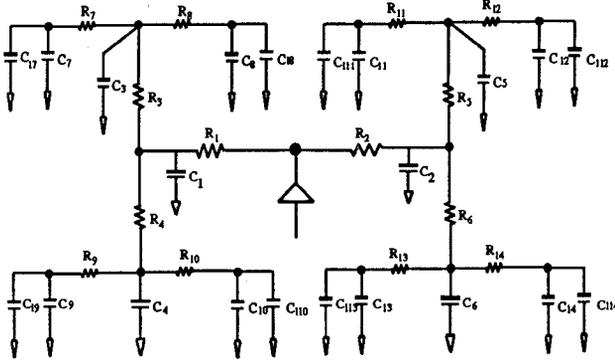


Figure 2: Equivalent RC tree for route in Figure 1.

common path from node  $i$  to the root and from node  $j$  to the root. The capacitance at node  $i$  is denoted by  $C_i$ . There are additional capacitances  $C_{ii}$  at the leaf nodes. The downstream capacitance,  $C_{d,i}$ , at a node  $i$  is defined as the sum of the capacitances at node  $i$  and all its descendant nodes.  $U(j)$  is the set of all branches that lie on the path from node  $j$  to the root of the tree.  $R_{u,j}$  (the upstream resistance) denotes the sum of resistances of the branches in set  $U(j)$ . If node  $i$  lies on the path from node  $j$  to the root, branch  $j$  ( $i$ ) as well as node  $j$  ( $i$ ) are said to lie *downstream* (*upstream*) of branch  $i$  ( $j$ ) and node  $i$  ( $j$ ).

## 2.1 Elmore Delay Approximations

The Elmore delay model [2, 7] is commonly used to approximate the signal delay in RC tree networks. This model approximates the delay from the root to any node  $n$  in a lumped RC tree by the sum of the products of the branch resistance,  $R_i$ , and the downstream capacitance,  $C_{d,i}$ , for every branch on the path from the root to the node, i.e.

$$T_{D,n} = \sum_{i \in P(n)} R_i C_{d,i} \quad (1)$$

where  $P(n)$  is the set of nodes that lie on the path from the root to node  $n$  excluding the root.

However, it should be noted that the Elmore delay is a first-order approximation which is generally applied as a dominant time constant, or step-response delay approximation for the circuit. In reality, for slow

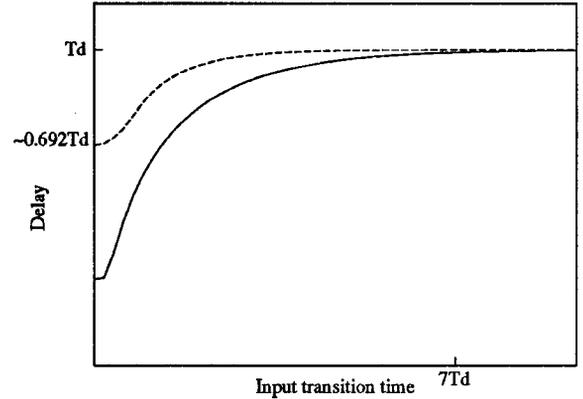


Figure 3: Plots of delay versus input transition time for two nodes in an RC tree with equal  $T_D$ s.

input transition times (inputs to the RC net, which are outputs of the clock driver), the signal is a ramp follower with an actual delay  $T_D$  [11]. For faster transition times, the Elmore delay is an approximation.

Figure 3 shows the variation in the actual delay for two nodes in an RC tree as a function of the input rise time. A highly-accurate third-order model was used to generate these curves [1]. For a step input, the 50% delay is approximated by  $0.692T_D$ . However, from Figure 3, we observe that this approximation may, at times, be inaccurate. Roughly speaking, when the transition time is less than  $7T_D$ , the response is no longer a ramp follower and the first order model may have significant error [11]. Therefore, a clock net designed for zero skew using the Elmore delay model may not yield zero skew for small input rise times. We expect, however, that the non-zero skew due to the inaccuracy of the Elmore delay model for small rise times is only a second-order effect when compared to the skew due to process-related wire width variations. Our approach does not preclude the use of more accurate delay models [3]; however, the first-order model does provide excellent efficiency.

In order for an algorithm to attain zero skew for any input rise time it would have to match the delay curves at all the leaf nodes in the circuit – a seemingly impossible task. However, reducing the delay will reduce the transition-time induced skew.

## 2.2 Reliable Clock Routing

During fabrication, the width of a line on a chip may differ from the expected width because of process variations. Ideally, the clock net should guarantee a certain skew and delay taking these process variations into account. It is essential for a robust clock net to exhibit a certain degree of insensitivity to these process variations.

Consider a change in the width of branch 1 of Figure 1 from a specified value of  $w_1$  to  $w_1 + \Delta w$  due to process variation.  $\Delta w$  is a random variable which does not depend on the branch width. Its statistical characteristics are determined by the process. If the new resistance and capacitance of branch 1 due to this variation are  $R_1 + \Delta R_1$  and  $C_1 + \Delta C_1$  respectively, the

change in the Elmore delay to any node  $n$  downstream of branch 1 is given by

$$\Delta T_{D_n} = \Delta R_1(C_{d_1} - C_1) \quad (2)$$

From (2), we see that the skew of binary tree-like clock nets is extremely sensitive to the changes in the widths of the branches closest to the root of the tree. Small changes in the widths of such branches can, therefore, have a large effect on skew. As a result, a carefully designed low-skew clock net may yield high skew values during actual fabrication due to process-related wire width variations, if these variations are not taken into account during design.

### 3 Sensitivity of Elmore Delays to Circuit Parameters

For an RC tree, the first moment of the impulse response at any node is the Elmore delay. It has been shown that this first moment for all the nodes in the circuit can be obtained from the solution of a dc-equivalent circuit in which all the capacitors are replaced by current sources of value equal to the capacitance[3]. Since dc-circuit sensitivities can be readily determined by using the *adjoint* sensitivity technique[9], the sensitivity of the Elmore delays to circuit parameters can be obtained with similar ease.

The adjoint of an RC circuit [9] is topologically equivalent to the original circuit except that the independent sources are set to zero and the output node of interest is driven by a unit current source. An adjoint analysis of node  $i$  in an RC tree which has a nominal delay  $T_{D_i}$  can be shown to yield the following sensitivity equations for circuit parameters  $R_j$  and  $C_j$ :

$$\begin{aligned} \frac{\partial T_{D_i}}{\partial R_j} &= C_{d_j} & R_j \in U(i) \\ &= 0 & \text{otherwise} \end{aligned} \quad (3)$$

$$\frac{\partial T_{D_i}}{\partial C_j} = R_{e_{ij}} \quad (4)$$

To compute the sensitivities for a set of nodes, a different adjoint network would have to be set up and analyzed for each node of interest. Instead, the delay sensitivities for all nodes with respect to all the resistances and capacitances in the circuit can be calculated using (3) and (4) in  $O(n^2 \log(n))$  time by path-tracing the tree. These sensitivities can be used to obtain the sensitivities with respect to the wire widths.

#### 3.1 Delay Sensitivity to Wire Width

The change in delay to a pin with respect to a wire can be obtained from the sensitivity of delay with respect to the wire capacitance and resistance by applying the chain rule. The sensitivity coefficient for pin  $i$  with respect to wire  $j$  is expressed as follows:

$$\frac{\partial T_{D_i}}{\partial w_j} = \frac{\partial T_{D_i}}{\partial R_j} \frac{\partial R_j}{\partial w_j} + \frac{\partial T_{D_i}}{\partial C_j} \frac{\partial C_j}{\partial w_j} \quad (5)$$

The terms  $\partial T_{D_i}/\partial R_j$  and  $\partial T_{D_i}/\partial C_j$  can be evaluated as described previously.  $\partial R_j/\partial w_j$  and  $\partial C_j/\partial w_j$  can

be computed from the following equations for the resistance and capacitance per unit length:

$$R = \frac{K_R}{w} \quad (6)$$

$$C = K_C w + C_{fr} \quad (7)$$

where  $K_R$  and  $K_C$  are independent of the width,  $K_C w$  is the area capacitance and  $C_{fr}$  refers to the fringe capacitance. Differentiating (6) and (7) with respect to  $w$  yields

$$\frac{\partial R}{\partial w} = -\frac{R}{w} \quad (8)$$

$$\frac{\partial C}{\partial w} = \frac{C - C_{fr}}{w} \quad (9)$$

whence,

$$\frac{\partial T_{D_i}}{\partial w_j} = \frac{C_j - C_{fr_j}}{w_j} \frac{\partial T_{D_i}}{\partial C_j} - \frac{R_j}{w_j} \frac{\partial T_{D_i}}{\partial R_j} \quad (10)$$

A positive value of sensitivity obtained from (10) indicates a case where widening wire  $j$  increases the delay to leaf node  $i$  while a negative value indicates that the delay decreases.

It is clear from (10) that the sensitivities are a function of the wire widths (similar expressions can be derived in terms of wire lengths). This implies that the sensitivities need to be recomputed whenever the width of a wire is modified. The next section demonstrates the simplicity of this task.

#### 3.2 Updating Sensitivities

The computation of the sensitivities is a one-time process. Whenever the width of a wire changes, the sensitivities can be updated in  $O(n^2)$  time as opposed to the  $O(n^2 \log(n))$  time needed for the initial calculation of sensitivities.

Changing the width of branch  $j$  by  $\Delta W_j$  causes the resistance and capacitance to change by  $\Delta R_j$  and  $\Delta C_j$  respectively. The effect of this change on the sensitivities of the delay of the leaf nodes with respect to the widths of all branches in the tree can be summarized as follows:

$$\Delta \frac{\partial T_{D_i}}{\partial w_k} = \begin{cases} \frac{C_k}{w_k} \Delta R_j & k \in D(j) \text{ and } j \in U(i) \\ -\frac{R_k}{w_k} \Delta C_j & k \in U(i) \cap U(j) \\ \frac{2C_{d_j}}{w_j} \Delta R_j & j = k \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

### 4 Clock Tree Synthesis

With efficient algorithms to calculate the delay sensitivities of nodes in an RC tree, we can propose the following steps for clock-tree synthesis:

#### 4.1 Delay Reduction

Figure 4(a) shows the distribution of the delays to the leaf nodes of a typical clock tree. The extremal points of this distribution determine the skew while the average determines the delay. During delay reduction, we bring the average of this distribution to or slightly below a specified target delay,  $T_{target}$ .

Sensitivities provide a means of selecting the right wires to widen in order to bring the average delay of an RC tree as close as possible to a specified target delay. The delay to a single node can be reduced trivially by widening the wire that has the highest *negative* sensitivity to this node. However, widening this wire may have a detrimental effect on delays to nodes which have *positive* sensitivities with respect to this wire. The ideal candidate for wire widening should, therefore, be selected using a more global metric.

A cost  $D_j$  is assigned to every branch  $j$  in the tree. This cost takes into account the effect of widening wire  $j$  on all the leaf nodes. If  $S_{ij}$  denotes the sensitivity of the delay to pin  $i$  with respect to wire  $j$  then the cost  $D_j$  is given by:

$$D_j = \sum_{i=1}^N S_{ij}(T_{D_i} - T_{target}) \quad (12)$$

where  $N$  denotes the number of leaf nodes (clock pins) in the tree.

We observe the following from the above equation:

- If  $T_{D_i} > T_{target}$ , and  $S_{ij} < 0$  (widening decreases delay),  $D_j$  decreases.
- If  $T_{D_i} < T_{target}$ , and  $S_{ij} > 0$ ,  $D_j$  decreases.
- If  $T_{D_i} > T_{target}$ , and  $S_{ij} > 0$ ,  $D_j$  increases.
- If  $T_{D_i} < T_{target}$ , and  $S_{ij} < 0$ ,  $D_j$  increases.

The first two cases clearly aid delay reduction by bringing the distribution of the delays closer to the target delay. The last two cases tend to disturb this distribution by increasing the delay of the pins with already high delays and lowering those with already low delays. Hence, at each iteration the wire with the *least* cost is widened by a constant amount which is based upon the minimum grid size. The sensitivities are recomputed and this procedure continues until the specified target delay is achieved.

#### 4.2 Skew Reduction

Assuming that the distribution of delays in Figure 4(a) is normal, we might define the clock-tree skew as  $6\sigma_{nom}$ , where  $\sigma_{nom}$  is the standard deviation of this normal distribution. Referring to Figure 4(b), the *nominal skew* may be approximated by the difference between the nominal delay at pin  $i$ ,  $T_{D_i}^{nom}$ , and the nominal delay at pin  $j$ ,  $T_{D_j}^{nom}$ , since these delays lie near the boundaries of the nominal distribution.

As shown in Figure 4(b), however, delays  $T_{D_j}$  and  $T_{D_i}$  can vary significantly due to process disturbances. In fact, when all the wires in the clock tree are thin, statistical variations in the widths of wires closest to

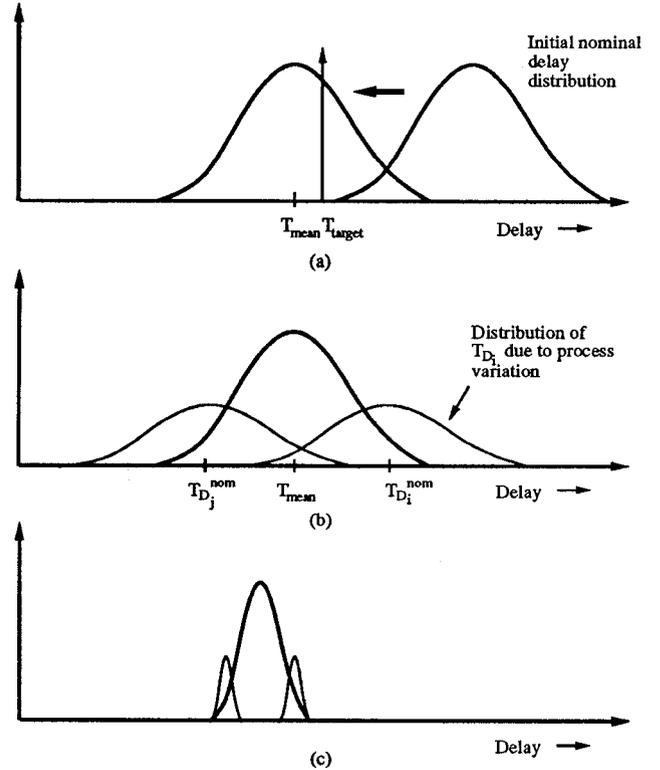


Figure 4: Distribution of delays before and after de-sensitization and skew minimization

the driver may have an enormous impact on the *actual* value of skew. In other words, if the delays  $T_{D_i}$  and  $T_{D_j}$  are also modeled as normal distributions with standard deviations of  $\sigma_{T_{D_i}}$  and  $\sigma_{T_{D_j}}$  respectively, the  $\sigma_{T_{D_i}}$ 's may be significantly larger than the  $\sigma_{nom}$ . A worst-case *bound* on the skew may, therefore, be expressed as

$$skew_{bound} = 6\sigma_{nom} + 3\sigma_{T_{D_i}} + 3\sigma_{T_{D_j}} \quad (13)$$

To reduce the skew in a reliable manner, we must reduce the variance of the nominal delay distribution, as well as the variances of all the nominal delays due to process variations, as shown in Figure 4(c). Toward this end, we begin by first attempting to reduce all  $\sigma_{T_{D_i}}$ 's to much less than  $\sigma_{nom}$ . Then, once all wires are de-sensitized in this manner, we attempt to reduce  $\sigma_{nom}$  while maintaining the de-sensitization conditions on all wires.

##### 4.2.1 De-sensitization

Low values of absolute sensitivities imply negligible change in delays for small changes in wire widths. This information can be used to guide the reduction of the  $\sigma_{T_{D_i}}$ 's due to process variations. It is well understood that increasing the widths of the wires should reduce the delay sensitivities at the leaf nodes since small process variations of the widths would result

in smaller changes in the overall delays, and consequently, smaller skew. The objective then is to make the delay sensitivities at the leaf nodes *small* enough so that the upper bound on the skew is acceptable without widening wires excessively.

Consider, as an example, that the width of a single wire  $j$  in the clock tree varies from its expected value due to a single process disturbance (e.g. a spot defect). Under such circumstances, we would want to make the delay to all leaf nodes insensitive to changes in the width of wire  $j$ . The maximum process variation in the width of wire  $j$  corresponds to a maximum change in delay, at one of the leaf nodes,  $\Delta T_{D_{max}}^j$ . Therefore, to de-sensitize wire  $j$ , we might increase  $w_j$ , so that  $\Delta T_{D_{max}}^j$  is less than the maximum allowable change in skew,  $\Delta T$ . Or, increase  $w_j$  until

$$\Delta w_{max} S_{ij} < \Delta T \quad (14)$$

for all leaf nodes  $i$ , where  $\Delta w_{max}$  represents the maximum change in  $w_j$  due to process variations.

Under the assumption of a single-defect model, we would de-sensitize all of the wires in the tree as follows: Starting at the leaf nodes, widen all of the wires in a bottom-up manner so that the maximum change in delay at any pin due to process variations at wire  $j$  is less than some acceptable change in skew,  $\Delta T$ . In this paper, we have chosen  $\Delta T$  to be  $\sigma_{nom}/2$ .

It is important to point out that the wires are widened in a bottom-up manner from the leaf-nodes in order to properly consider the possible changes in the upstream sensitivities which tend to increase the  $\sigma_{T_{D_i}}$ s. Note that it can be shown that increasing the width of a wire  $j$ , is guaranteed to make all positive sensitivities smaller, and most negative sensitivities smaller in magnitude. However, those wires which lie along the unique path from  $j$  to the root node may have the magnitude of their negative sensitivities increased by the increase in their downstream capacitance. Thus, the bottom-up traversal from the leaf-nodes will use the correct value of downstream capacitance.

Of course, all of this assumes a single defect model which is far from accurate. Under- and over-etching, for example, influence a large number of wires simultaneously in a manner which is difficult to predict. Therefore, we establish a more stringent condition for de-sensitization by widening wires until the following holds:

$$\Delta w_{max} S_{ij} < \Delta T/L \quad (15)$$

where  $L$  is the depth of the tree. In other words, we do not allow any wire in the tree to cause a change in delay of more than  $\Delta T/L$  at a leaf node thereby ensuring that the maximum possible change in delay from the root to a leaf node is less than  $\Delta T$ .

#### 4.2.2 Reducing Nominal Skew

After the wires are de-sensitized, we attempt to narrow the distribution of delays about its mean value, thereby reducing the nominal skew. This is accomplished by attempting to make the delays of the leaf

nodes as close to the mean of the distribution as possible. In this paper, the sensitivities are used to select wires which narrow the distribution while disturbing the mean of the distribution as little as possible. Let  $\Delta T_{ij}$  denote the change in the delay of node  $i$  when the width of wire  $j$  is changed by an amount  $\Delta w_j$ . The skew would be reduced to zero if

$$\Delta T_{ij} = T_{mean} - T_{D_i} \quad (16)$$

for all leaf nodes  $i$  in the tree. However, it seems highly improbable that skew could be eliminated by varying the width of a single wire. Therefore, we select a wire that forces the new delays of as many pins as possible, closer to the current mean delay. To choose the best wire to widen for skew reduction we assign a cost

$$D_j = \sum_{i=1}^N (|T_{D_i} + \Delta T_{ij} - T_{mean}|) \quad (17)$$

to each wire  $j$ . Skew could be eliminated if we could find a wire with zero cost. Alternatively, we select to widen the wire with the least cost. The change in delay  $\Delta T_{ij}$  is estimated using sensitivities by

$$\Delta T_{D_i} = S_{ij} \Delta w_j \quad (18)$$

It should be noted that (18) is valid only for small changes in width. The wire width increment,  $\Delta w_j$ , is constant for all iterations and is based upon the grid size.

To summarize, the complete clock tree synthesis approach is to reduce delay, de-sensitize, then reduce skew. However, we recall from Section 4.2.1 that widening a wire  $j$  may increase the magnitude of the sensitivity of wires upstream of  $j$ . In order to avoid undoing the de-sensitization process, we add an additional step (4) to the clock tree synthesis procedure as follows:

1. Meet RC delay requirement. (Delay reduction phase).
2. Starting at the leaf nodes, widen wires until their  $\Delta T_{D_{max}}$  is less than  $\sigma_{nom}/2L$ . (Desensitization phase).
3. Use (17) to select the best wire to reduce skew. Widen this wire by a predetermined increment. Update  $\sigma_{nom}$ .
4. Starting at the modified wire, trace back to root widening wires as necessary to ensure that  $\Delta T_{D_{max}}$  is still less than  $\sigma_{nom}/2L$ .
5. Go to step 3 until skew objectives are met.

## 5 Results

We applied the clock tree synthesis procedure to the 5 examples in [4]. The per micron resistance and capacitance are assumed to be  $3m\Omega$  and  $0.02fF$  respectively. These examples were routed using the methods

Example	# pins	Initial		Final		Worst case skew (ns)	$w_{max}$
		Skew (ns)	Delay (ns)	Skew (ns)	Delay (ns)		
r1	267	0.325	2.23	0.080	1.270	0.203	4.5
r2	598	1.888	5.56	0.179	4.016	0.572	7.0
r3	862	1.529	7.55	0.149	4.295	0.498	11.8
r4	1903	4.893	20.25	0.579	11.506	1.252	7.9
r5	3101	4.370	40.85	0.898	16.978	2.141	6.4

Table 1: Delay and skew results for examples in [4].

of means and medians[5]. The initial skew and average delay are shown in columns 3 and 4 of Table 1.

Selecting a target delay of one-half the initial delay and a target skew of 5% of the initial delay, we applied the clock tree synthesis procedure described above. Using wire width increments of 0.3 microns during the skew reduction phase (steps 3 – 5) these targets were achieved in less than 100 iterations. As expected, the quality of the results increases with the number of iterations. Columns 5 and 6 show the nominal skew and delay after 100 iterations. To verify the effectiveness of the desensitization phase, a Monte Carlo simulation of 500 trials was performed on the final clock tree by varying the widths of all wires using a normal distribution with  $\sigma = 0.05$  microns (which is a reasonable assumption for today's submicron processes). The worst case skew results of the Monte Carlo simulation are shown in Column 7. Column 8 shows the maximum wire widths in the final clock tree. Note that, as expected, the widest wires appear near the root – they were widened during the desensitization phase.

## 6 Extensions to Clock Meshes

The first moment computation for an RC tree configuration has been shown to be a simple task [10]. We have also demonstrated the ease of calculation of sensitivities for RC tree structures. Most clock routes have tree topologies; however, recently there has been a great deal of interest in clock meshes. The calculation of the moments for RC meshes can be accomplished in a very efficient manner [1]. As for the sensitivities, we know that they can be calculated for RC meshes since the adjoint method applies to generalized circuits. We are, however, currently working on ways to calculate the sensitivities of RC meshes with acceptable efficiency. Such a capability would be extremely useful, since along with enabling wire-width optimization for RC clock meshes, an efficient adjoint sensitivity approach for meshes could be used to calculate the sensitivities with respect to adding loops of metal. In words, the sensitivities could aid in predicting the effect of adding a loop where there was none to begin with, to improve skew and/or reliability.

## 7 Conclusion

We have presented a technique to perform clock tree routing using wire width adjustment. Our algorithm yields low values for delay and skew for any type of tree configuration. Most importantly, we introduce

the concept of reliable clock net design. Our approach increases the reliability of the clock net by considering the effect of process variations on the skew.

## 8 Acknowledgment

We wish to thank Ashok Balivada for his participation in the initial stages of this project. We also wish to thank Dr. Ren-Song Tsay for providing us with the example circuits.

## References

- [1] C. L. Ratzlaff, N. Gopal, and L. T. Pillage. "RICE: Rapid Interconnect Circuit Evaluator," *Proc. 28th ACM/IEEE Des. Auto. Conf.*, Jun 1991.
- [2] W. C. Elmore. "The transient response of damped linear networks with particular regard to wideband amplifiers," *J. Applied Physics*, 19(1), 1948.
- [3] L. T. Pillage and R. A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Trans. Comp. Aided Design*, 9, 1990.
- [4] Ren-Song Tsay. "Exact zero skew," *Proc. IEEE Int'l. Conf. Computer-Aided Des.*, Nov. 1991.
- [5] M. A. B. Jackson, A. Srinivasan, and E. S. Kuh. "Clock routing for high performance ICs," *Proc. 27th ACM/IEEE Des. Auto. Conf.*, Jun 1990.
- [6] Ting-Hai Chao, Yu-Chin Hsu, and Jan-Ming Ho. "Zero skew clock net routing," *Proc. 29th ACM/IEEE Des. Auto. Conf.*, Jun 1992.
- [7] P. Penfield and J. Rubinstein. "Signal delay in RC tree networks," *Proc. 19th ACM/IEEE Des. Auto. Conf.*, 1981.
- [8] A. Kahng, J. Cong and G. Robins. "High-performance clock routing based on recursive geometric matching," *Proc. 28th Des. Auto. Conf.*, 1991.
- [9] S. W. Director and R. A. Rohrer. "The generalized adjoint network sensitivities," *IEEE Trans. Circuit Theory*, Vol. CT-16, no. 3, Aug. 1969.
- [10] C. J. Terman. "Simulation tools for digital LSI design," PhD thesis, Massachusetts Institute of Technology, Sept. 1983.
- [11] C. L. Ratzlaff, S. Pullela, and L. T. Pillage. "Modeling the RC-interconnect effects in a hierarchical timing analyzer," *Proc. IEEE Custom Integrated Circuits Conf.*, 1992