TACO: Temperature Aware Clock-tree Optimization

Minsik Cho*, Suhail Ahmed*† and David Z. Pan*

*Dept. of ECE, The University of Texas at Austin, Austin, TX 78712

†Design Technology Org. Freescale Semiconductor, Austin, TX 78729

thyeros@cerc.utexas.edu, suhail.ahmed@freescale.com, dpan@ece.utexas.edu

Abstract— In this paper, an efficient linear time algorithm TACO is proposed for the first time to minimize the worst case clock skew in the presence of on-chip thermal variation. TACO, while tries to minimize the worst case clock skew, also attempts to minimize the clock tree wirelength by building up merging diamonds in a bottom-up manner. As an output, TACO provides balanced merging points and the modified clock routing paths to minimize the worst case clock skew under thermal variation. Experimental results on a set of standard benchmarks show 50 - 70% skew reduction with less than 0.6% wirelength overhead.

I. INTRODUCTION

Due to aggressive technology scaling, VLSI integration density as well as power density increases drastically. For example, the power density of high performance microprocessors has already reached $50W/cm^2$ at $100nm$ technology and it will reach $100W/cm^2$ at $50nm$ technology [1]. Such higher power density will cause higher chip temperature overall. Meanwhile, to mitigate the overall power consumption, many low power techniques such as dynamic power management [2], clock gating [3], voltage islands [4], dual Vdd/Vth [5] and power gating [6], [7] are proposed recently. These techniques, though helpful to reduce the overall power consumption, may cause significant on-chip thermal gradients and local hot spots due to different clock/power gating activities and varying voltage scaling. It has been reported in [8] that temperature gradients of $30°C$ can occur in a high performance microprocessor design. The magnitude of thermal gradients is expected to increase further as VLSI designs move into nanometer processes and multi-GHz frequencies.

Thermal gradients exist not only in the substrate layer, but also in higher metal layers [9]. Recent studies [10] [11] show that Joule heating of global wires can cause localized heat confinement in advanced VLSI design since the low-k inter-layer dielectrics (ILD) in nanometer designs are poor thermal conductors [12]. As a result, thermal gradient issues must be considered during *temperature aware* performance optimization. Since clock nets are among the most sensitive signals to delay variations caused by thermal variation [13] [14], it is important to study the temperature aware clock tree optimization. Note that since clock network consumes significant amount of power, the thermal impact from clock network itself should be considered as well. Existing clock tree algorithms [15]–[17],

however, all build the zero/bounded skew clock trees assuming a uniform thermal condition.

In this paper, we propose a new temperature aware clock-tree optimization (TACO) algorithm to address the drawbacks of previous clock tree optimization algorithms. To our best knowledge, this is the first time that thermally induced clock skew, (including skew induced by the clock tree itself) is taken into account to minimize both the clock skew and the total wirelength. Our major contributions are as follows:

- We show that classic deferred merge embedding (DME) based algorithms [15]–[17] are no longer sufficient for zero/bounded skew clock trees under thermal variation since the merging points and segments are *path-dependent*.
- We introduce the concept of *merging diamond* which considers both uniform and non-uniform thermal profiles during the bottom-up clock tree construction and optimization, and use it to guide both clock skew and wirelength minimization under thermal variation.
- We propose effective algorithms to prune out redundant solutions with merging diamond.
- We incorporate an accurate ADI-based thermal simulation [18] to feedback the thermal impact of the resulting tree from TACO so that *thermal closure* can be achieved.

The remainder of the paper is organized as follows. In Section II, preliminaries are described. In Section III, we formulate the problem. The TACO algorithm overview and optimization procedure are explained in Section IV. Experimental results are shown in Section V. Section VI concludes this paper.

II. PRELIMINARIES

A. Delay Model

Elmore delay model is used for interconnect delay computation considering the temperature at a given region. Clock wire resistance per unit length under thermal variation can be calculated as:

$$r = r_o\{1 + \beta \cdot T(x,y,t)\} \qquad (1)$$

where r_o is the unit length resistance at $0°C$, β is the temperature coefficient of resistance and T is the temperature expressed as a function of location (x,y) and time t [14]. The capacitance is assumed to be invariant with respect to the temperature as in [14]. And for wire model, the π network is used for the simulation.

B. Notations & Definitions

Table I lists the notations used throughout this paper.

$TR(i)$	A binary clock tree rooted at point i
$TR_l(i)$	The left child of point i
$TR_r(i)$	The right child of point i
P_u	The uniform thermal profile of the chip
P_w	The worst case thermal profile of the chip
$D(i,P)$	Delay from point i to any leaf in $TR(i)$ under the thermal profile P
$SKEW(i,P)$	$Max(D(i,P)) - Min(D(i,P))$ of point i under the thermal profile P
$MS(i,j,P)$	A merging segment of two points i, j under the thermal profile P
$DIST(i,j)$	Manhattan distance between i and j
r_o	The wire resistance per unit length
c_o	The wire capacitance per unit length

DEFINITION 1. **Worst Case Thermal Profile:** We define a thermal profile P_w as the worst case thermal profile for a given clock tree $TR(r)$, if $SKEW(r,P_w) \geq SKEW(r,P)$ for any sampled thermal profile P.

DEFINITION 2. **Equal Delay Point:** A point i is defined as an equal delay point with a thermal profile P, if $D(TR_l(i),P) = D(TR_r(i),P)$. For instance, point p in Fig. 1(a) is an equal delay point under the uniform thermal profile P_u and point y is an equal delay point under the worst case thermal profile P_w.

DEFINITION 3. **Merging Segment:** A merging segment is defined as the loci of equal delay points, notated as $MS(u,v,P)$ with respect to two child points u,v under a thermal profile P like $MS(u,v,P_u)$ and $MS(u,v,P_w)$ in Fig. 1(a).

DEFINITION 4. **Balanced Skew Point:** A point i is defined as balanced skew point, if $SKEW(i,P_x) = SKEW(i,P_y)$ with respect to two thermal profiles P_x and P_y. In Fig. 1(a), a point b is the balanced skew point which has the same skew for two thermal profiles P_u and P_w.

DEFINITION 5. **X-Cut Regions:** X-Cut regions are the set of four regions obtained by performing $\pm 45°$ cuts with lines intersecting at point i, denoted as $XR(i)$. An example is shown in Fig. 6(b).

DEFINITION 6. **Merging Diamond:** Merging diamond is defined as the smallest polygon centered at a point i and formed by balanced skew points/equal delay points from each region in $XR(i)$ and is notated as $bMD(i)/eMD(i)$. Fig. 1(b) shows two kinds of merging diamonds, $bMD(p)$ and $eMD(p)$. Note that merging diamond can be degenerated.

C. DME Based Algorithm & Limitation

DME based algorithms [15]–[17] can embed any given clock tree topology in the Manhattan plane with exact zero skew and minimum wirelength. DME based algorithm consists of two phases, bottom-up and top-down. During the bottom-up phase, a merging segment such as $MS(u,v,P_u)$ in Fig. 1(a)
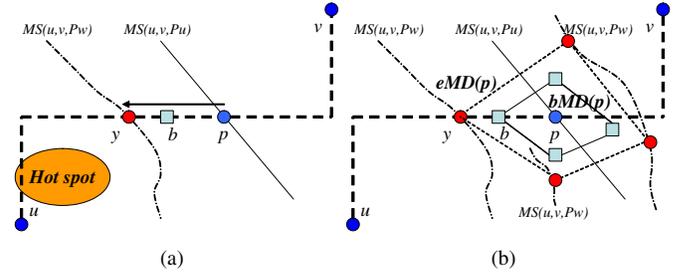
Fig. 1. (a) $MS(u,v,P_u)$ denotes a merging segment under uniform thermal profile and $MS(u,v,P_w)$ denotes a merging segment under worst case thermal profile. Equal delay point location moves from point p to point y as hot spot appears; (b) $eMD(p)$ is due to equal delay points on $MS(u,v,P_w)$ and $bMD(p)$ is due to balanced skew points (shown as square points).

is created at each node in a clock tree. Under the uniform thermal profile P_u, $MS(u,v,P_u)$ is a $\pm 45°$ slope line as any point p on $MS(u,v,P_u)$ has equal distance and hence equal delay from u and v. During the top-down phase, a point like point p is chosen from each merging segment for minimum wirelength.

However, DME based algorithm does not deal with a non-uniform thermal profile as a merging segment may no longer be a $\pm 45°$ slope line because different paths with the same wirelength may have different delays (*path-dependence*) due to non-uniform wire resistance. Instead, it may consist of a set of discontinuous curves and/or points. Determination of such merging segments is expensive as the time complexity of such computations can be shown to be $O(N(\frac{D}{G}!))$, where N denotes the number of edges of the clock tree, D denotes the average distance between two sub-trees and G is DRC safe grid size.

III. PROBLEM FORMULATION

Since thermal profiles may be time variant, it is difficult to consider all different thermal profiles while constructing the clock tree. Instead, we simplify the problem by using the worst case thermal profile that captures the worst case systematic thermal-induced skew for the given initial clock tree. Such worst case thermal profile can be captured from a set of sampled thermal simulations performed over a period of time. Given a worst case thermal profile, it is still not computationally trivial to construct a good clock tree with DME based algorithms as mentioned in Section II-C.

We divide the clock tree construction problem under thermal variation into two sub-problems: the first is to construct the initial clock tree under the uniform thermal condition that can be solved efficiently by DME, BST or other linear algorithms, and the second is defined as follows:

Optimization of a Given Clock Tree under Thermal Variation: *Given a zero skew clock tree $TR(r)$ under the uniform thermal profile P_u, and a worst case thermal profile P_w, optimize $TR(r)$ such that the worst case clock skew of $TR(r)$ is minimized under thermal variation with minimal change in the total wirelength.*

The advantages of our incremental approach include:

- The complexity of the problem is reduced to linear time.
- The optimized tree inherits the good properties of the given tree $TR(r)$.
- The thermal effect from clock tree itself is considered during the thermal simulation to ensure the thermal closure.

IV. TACO ALGORITHM

In this section, we present our temperature aware clock-tree optimization algorithm, TACO, which incrementally optimizes a given zero skew clock tree constructed under the uniform thermal condition. Basically, TACO migrates the tree nodes to new locations to reduce the worst case clock skew under thermal variation.

The motivation for TACO algorithm is based on the observation that the equal delay points in a clock tree under uniform temperature will get shifted toward hot spots under the worst case thermal gradients [14]. However, such shifting may result in significant skew under the uniform temperature (e.g., when the chip powers up or wakes up). To illustrate this, in Fig. 1(a), let p be a parent node of u and v at some level of a zero skew clock tree under uniform temperature. Hence an equal delay point with respect to u and v is located at p. In the presence of thermal gradient however, due to the thermally increased resistance, the equal delay point gets shifted toward the hot spot to a new location y. As this sub-tree is still rooted at p, there will be a thermally induced skew proportional to the shift in the equal delay point. This skew can be minimized by setting the root point of the sub-tree at y. But such a setting would cause significant skew under the uniform temperature.

TACO addresses this problem by finding a balanced skew point between the two equal delay points corresponding to

---

Algorithm 1 TACO Algorithm

Input: Clock tree $TR(u)$, Worst case thermal profile P_w
1: **if** $TR_r(u)$ and $TR_l(u)$ are leaves **then**
2:     Construct $eMD(u)$
3:     Shrink $eMD(u)$
4:     Construct $bMD(u)$
5: **else**
6:     TACO $(TR_r(u), P_w)$ // right child traversal
7:     TACO $(TR_l(u), P_w)$ // left child traversal
8:     **for** each pair from $bMD(TR_r(u))$, $bMD(TR_l(u))$ **do**
9:         Construct $eMD(u)$
10:        Shrink $eMD(u)$
11:        Construct $bMD(u)$
12:     **end for**
13:     Find the smallest $bMD(u)$ // solution pruning
14:     **if** u is the root **then**
15:        Return the best solution
16:     **end if**
17: **end if**
Output: $TR(u)$ with the reduced worst case clock skew

Fig. 2. (a) $bMD(r)$ is constructed with two child merging diamonds $bMD(u)$ and $bMD(v)$ in bottom-up manner. Point s of $bMD(r)$ is due to point u' and v'; (b) a point s with the smallest worst case clock skew from $bMD(r)$ is selected as a new root and its corresponding tree is traversed in top-down manner.

the two different thermal conditions, uniform and worst case thermal profiles. As shown in Fig. 1(a), TACO finds such a balanced skew point b which provides the same amount of clock skew under both uniform and worst case thermal profiles. At each node of the clock tree, four closest balanced skew points are searched and a *merging diamond* is constructed. Fig. 1(b) shows a merging diamond, $bMD(p)$, made of the balanced skew points corresponding to node p at some level of the clock tree. Note that this balanced skew merging diamond provides a middle ground between the equal delay point p under uniform temperature and the equal delay merging diamond ($eMD(p)$) under worst case thermal profile.

Fig. 2(a) shows how merging diamonds are constructed in a bottom-up manner. At first, given the initial clock tree rooted at r (shown in dotted lines), child merging diamonds $bMD(u)$ and $bMD(v)$ are constructed. Next, the parent merging diamond $bMD(r)$ is constructed using $bMD(u)$ and $bMD(v)$. Then a point in $bMD(r)$ with the smallest worst case clock skew (let it be s) is selected at the root level and its corresponding tree (shown in solid lines in Fig. 2(b)) is traced in a top-down manner. Since balanced skew points like s, u' and v' are searched in the vicinity of merging points of the given initial clock tree, the final tree built by TACO ensures minimized overall worst case clock skew while keeping the total wirelength similar to that of the initial clock tree.

The overall flow of TACO is shown in Algorithm 1, which will be explained in detail in the rest of this section. Section IV-A shows how to construct balanced skew merging diamonds. Section IV-B illustrates the bottom-up construction of merging diamonds. Section IV-C explains the top-down solution selection and the iterative process for thermal closure. Section IV-D gives the complexity analysis of the overall TACO algorithm.

A. Merging Diamond Construction

In this subsection, we describe how to construct a balanced skew merging diamond using three key steps in TACO. At each node of a given clock tree, an equal delay merging diamond is initially constructed and is further shrunken before a balanced skew merging diamond is constructed (line 2 - 4 and line 9 - 11

Fig. 3. (a-c) an equal delay point $a/b/c$ is found along a given path between two children, u and v; (d) an equal delay point d is found and an equal delay merging diamond $eMD(p)$ is constructed.

in Algorithm 1). The concept of merging diamonds is helpful in pruning inferior balanced skew points during bottom-up construction.

1) Equal Delay Merging Diamond Construction: Fig. 3 illustrates how to construct an equal delay merging diamond around a parent p with two children u and v by allowing only L-shape routing. In Fig. 3(a), the equal delay point a is calculated by using Equation (24) in [14] along the shown routing path. Other equal delay points b, c and d are calculated in the same way for different L-shape paths as shown in Fig. 3 (b), (c) and (d). These four points, (a, b, c, d) form an equal delay merging diamond around point p, denoted by $eMD(p)$ as shown in Fig. 3(d).

2) Equal Delay Merging Diamond Shrinkage: Shrinking $eMD(p)$ around the original merging point p as much as possible results in (i) minimal worst case clock skew and (ii) minimal change in wirelength, due to minimal shift of the equal delay points from the original merging point. As thermal profile around p is continuous (temperature locality),
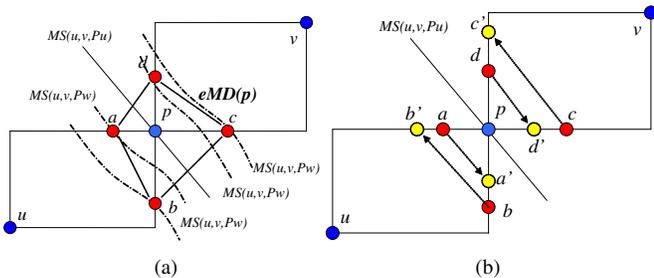
Fig. 4. (a) each equal delay point of $eMD(p)$ is on $MS(u, v, P_w)$; (b) four points a', b', c' and d' are projected from a, b, c and d respectively based on the temperature locality to make $eMD(p)$ smaller.
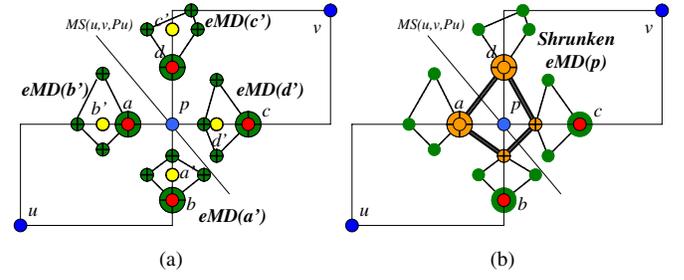
Fig. 5. (a) merging diamonds $eMD(a')$, $eMD(b')$, $eMD(c')$ and $eMD(d')$ are constructed around projected points a', b', c' and d'; (b) the shrunken $eMD(p)$ is constructed with the selected equal delay points.

each point in $eMD(p)$ has a corresponding $MS(u, v, P_w)$, which runs almost parallel to $MS(u, v, P_u)$ as in Fig. 4(a). Using this postulation, we project new potential equal delay points like a', b', c' and d' as shown in Fig. 4(b). Since these points are not exact, but projected, TACO discovers the exact equal delay points by constructing a merging diamond around each of these projected points. For example, the merging diamond, $eMD(b')$ shown in Fig. 5(a), can be constructed in the same way as described in Section IV-A.1 by treating b' as a parent p. A new shrunken diamond $eMD(p)$ can be formed using all the exact equal delay points obtained so far (16 points from $eMD(a'), eMD(b'), eMD(c'), eMD(d')$ and 4 points a, b, c, d) as in Fig. 5(b). By repeating this procedure, it may be possible to find even smaller $eMD(p)$. According to our observation, the first iteration reduces the size of a merging diamond by 14% on average and the second iteration reduces it by only 0.002% on average for all test cases. This shrinkage step yields near smallest equal delay merging diamond which in turn yields near smallest balanced skew merging diamond at each level of the clock tree.

3) Balanced Skew Merging Diamond Construction: To minimize the worst case clock skew, we need to compute a balanced skew merging diamond $bMD(p)$ from the shrunken $eMD(p)$ at each level of the clock tree. A balanced skew merging diamond is formed using balanced skew points. On each routing path passing through a point i in $eMD(p)$ and a point x_i on $MS(u, v, P_u)$, there exists a balanced skew point m_i where $i \in \{a, b, c, d\}$ as shown in Fig. 6. The location of such a balanced skew point can be calculated using either binary search or a heuristic parameter K defined as $DIST(i, m_i)/DIST(i, x_i), i \in \{a, b, c, d\}$. For example, in Fig. 6(a), the balanced skew point m_a lies between x_a and a on the routing path which is passing through x_a and a. Similarly, other balanced skew points $(m_b, m_c$ and $m_d)$ are found on their corresponding routing paths. Balanced skew merging diamond $bMD(p)$ is then formed using the balanced skew points m_a, m_b, m_c and m_d.

B. Parent Merging Diamond Construction

After the child balanced skew merging diamonds are constructed, an equal delay merging diamond followed by a balanced skew merging diamond is built at the parent. Let

TABLE II

RESULTS FOR THE INITIAL CLOCK TREE FROM BST [19].

Input	Uniform thermal profile[a]		Worst case thermal profile					Overall
(node)	delay(ns)	skew(ps)	delay(ns)	skew(ps)	temp(°C)[b]	wire(um)	CPU(sec)	worst skew(ps)
r1(256)	1.8	0	1.9	53.9	63-95/83	1320666	2	**53.9**
r2(598)	4.4	0	5.0	228.9	66-104/89	2602908	5	**228.9**
r3(862)	7.0	0	8.1	216.3	66-107/92	3388951	7	**216.3**
r4(1903)	14.4	0	17.6	628.4	70-116/99	6828510	19	**628.4**
r5(3101)	35.3	0	44.9	2328.4	71-126/104	10242660	29	**2328.4**

TABLE III

RESULT FOR THE OPTIMIZED CLOCK TREE FROM TACO.

Input	Uniform thermal profile[a]		Worst case thermal profile					Overall	
(node)	delay(ns)	skew(ps)	delay(ns)	skew(ps)	temp(°C)[b]	wire(um)	CPU(sec)[c]	worst skew(ps)	imprv(%)
r1(256)	1.8	23.1	1.9	11.4	62-94/82	1323174	4	**23.1**	**57.1**
r2(598)	4.5	86.2	4.9	75.7	65-102/88	2615431	14	**86.2**	**62.3**
r3(862)	7.1	107.9	8.1	107.9	65-105/91	3399060	18	**107.9**	**50.1**
r4(1903)	14.7	220.7	17.6	220.7	68-114/98	6845621	31	**220.7**	**64.9**
r5(3101)	35.9	629.1	44.7	675.1	72-124/103	10302650	75	**675.1**	**71.0**

(a) temperature was 80°C for all benchmarks. (b) temperature is expressed as {min-max/avg}. (c) CPU time is the time for one iteration.

$bMD(u)$ and $bMD(v)$ be the balanced skew merging diamonds at points u, v respectively. And let p be the parent of u and v. By choosing one point from $bMD(u)$ and one point from $bMD(v)$, one $bMD(p)$ can be constructed as described in Section IV-A. Similarly, considering all such possible combinations, multiple $bMD(p)$s are constructed. The smallest $bMD(p)$ is then obtained using all the points from all $bMD(p)$s by picking a nearest point to p from each region in $XR(p)$ (line 13 of Algorithm 1). This idea has the same advantages as shrinkage of Section IV-A.2. Fig. 6(b) shows the parent $bMD(p)$ construction. During this step, inferior solutions are pruned out as a result of making the final merging diamond smaller. This step is carried out in bottom-up manner until the root is reached (see Fig. 2(a)).

C. Final Selection and Evaluation

As illustrated in Fig. 2(b), after the balanced merging diamond is constructed at the root, a balanced skew point with the smallest worst case clock skew is selected. The optimized clock tree is then obtained using top-down traversal from the selected point (line 15 of Algorithm 1). As the optimized clock

(a)   (b)

Fig. 6.   (a) the balanced skew merging diamond $bMD(p)$ is constructed with balanced delay points m_a, m_d, m_c, and m_d; (b) parent balanced skew merging diamond $bMD(p)$ is constructed from two child merging diamonds $bMD(u)$ and $bMD(v)$.

tree may affect the thermal profile, an ADI-based linear time thermal simulation [18] is performed to evaluate the worst case clock skew. This procedure, (TACO along with the thermal simulation) is repeated several times to ensure *thermal closure*. More discussion on iterations required to bring thermal closure is in Section V.

D. Overall Algorithm Analysis

This algorithm is analogous to a postorder tree traversal which is $O(N)$ where N denotes the number of edges of the tree. At each level, we need $O(1)$ for each merging diamond construction. For thermal convergence, if we need to repeat the algorithm for C (in practice, this is constant) times, the final time complexity is $O(NC)$. During the bottom-up phase, the exact routing paths for each merging diamond are stored. Hence, the overall memory requirement is $O(N)$.

V. EXPERIMENTAL RESULTS

We implement our algorithm in C++ and run the program on a 1.5GHz Pentium-4 PC. The benchmarks $r1 \sim r5$ are taken from [17]. All simulations use $r_o = 0.03\Omega/\mu m, c_o = 2.0 \times 10^{-16} F/\mu m$ [17] and $\beta = 0.0068(1/°C)$ [20]. To measure the worst case thermal profile, we embed a thermal simulator [18] into our program and use an industry design along with the clock tree for the simulations. The initial clock trees with zero skew under the uniform temperature are generated by the BST-DME obtained from [19].

To demonstrate thermal closure, we test TACO algorithm on $r5$. The first iteration of optimization reduces the worst case clock skew by about 70%. The following nine iterations reduce the skew by only 8% (less than 1% with each iteration). We consistently observe that ten iterations are enough to reach thermal closure.

We also observe the empirical method of using the parameter K (much faster than the binary search method) with $K = 0.4$ yields good overall results which are presented in Table II and III. Table II shows the delay, skew, total wirelength and
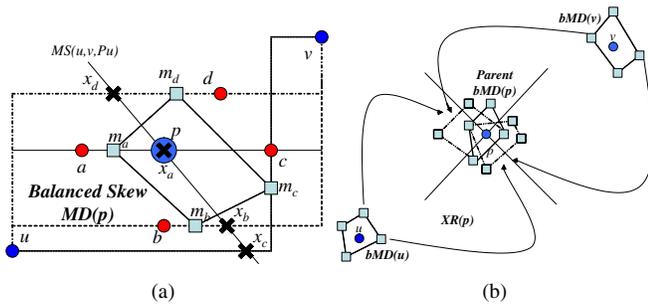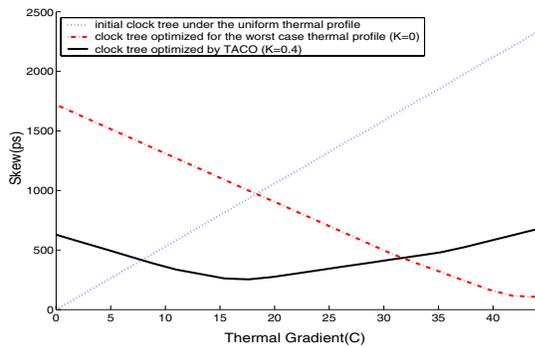
Fig. 7. Skew variations for three kinds of clock trees of $r5$ by the various thermal gradients.

Fig. 8. Initial clock tree (shown in solid line), and optimized clock tree (shown in dotted line) after TACO of $r3$.

CPU time under the uniform and the worst case thermal profile for all the benchmarks before optimization. Last column in Table II shows the worst case clock skew for each of the benchmarks. In Table II, one may observe huge skews under the worst case thermal profile.

The improvements and the overhead of the proposed algorithm are shown in Table III by comparing the delay, skew and wirelength between Table II and Table III. The worst case clock skew is reduced by 50% - 70%. As a penalty, the wirelength is increased by 0.35% on average which is negligible. Besides, the maximum temperature is slightly reduced. The CPU time is increased linearly with the number of nodes.

Regarding the sensitivity of the skew to the varying thermal gradients, Fig. 7 shows the clock skew variation for $r5$ for various thermal gradients. The zero skew tree under the worst case thermal profile is built by setting $K = 0$. It can be observed that the initial clock tree (optimal for the uniform thermal profile) and the clock tree optimized for the worst case thermal profile show huge worst case clock skew for highest and lowest thermal gradients respectively. While the two zero skew trees show huge worst case clock skew, the optimized tree by TACO shows the smallest worst case clock skew. Fig. 8 shows the clock routing changes due to TACO in dotted lines. TACO makes small changes to the initial clock tree.

VI. CONCLUSION

In order to cope with the increasing impact of thermal variation, we present an efficient, linear time temperature aware clock-tree optimization (TACO) algorithm to perform post optimization. Experimental results show that our approach reduces the worst case clock skew under thermal variation significantly (up to 70%) with negligible increase (less than 0.6%) in the total wirelength. We plan to take the clock buffers and IR-drop with thermal variation into account in the future.

REFERENCES
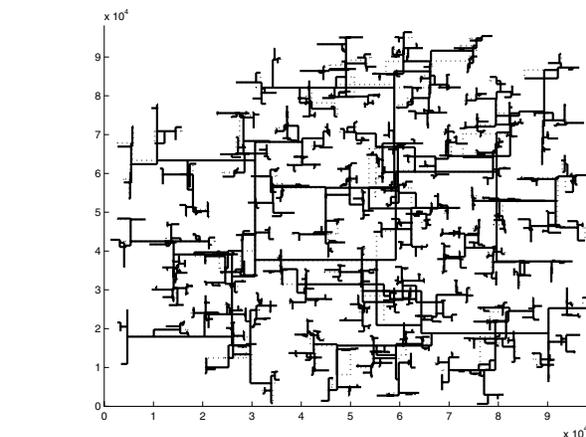
[1] International Technology Roadmap for Semiconductors (ITRS), 2003.
[2] Q. Wu, Q. Qiu, and M. Pedram, "Dynamic power management of complex systems using generalized stochastic Petri nets," in *Proc. Design Automation Conf.*, Jun 2000.
[3] J. Oh and M. Pedram, "Gated clock routing for low-power microprocessor design," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, pp. 715–722, Jun 2001.
[4] R. Puri, L. Stok, J. Cohn, D. Kung, D. Pan, D. Sylvester, A. Srivastava, and S. H. Kulkarni, "Pushing ASIC Performance in a Power Envelope," in *Proc. Design Automation Conf.*, 2003.
[5] A. Srivastava, D. Sylvester, and D. Blaauw, "Concurrent Sizing, Vdd and Vth Assignment for Low-Power Design," in *Proc. Design, Automation and Test in Eurpoe*, vol. 1, Feb 2004, pp. 718–719.
[6] J. Kao, A. Chandrakasan, and D. Antoniadis, "Transistor sizing issues and tool for multi-threshold CMOS technology," in *Proc. Design Automation Conf.*, 1997.
[7] C. Long and L. He, "Distributed sleep transistor network for power reduction," in *Proc. Design Automation Conf.*, 2003.
[8] P. Gronowski *et al.*, "High performance microprocessor design," *IEEE J. Solid-State Circuits*, vol. 33, pp. 676–686, May 1998.
[9] S. Im and K. Banerjee, "Full chip thermal analysis of planar (2-d) and vertically integrated (3-d) high performance ICs," in *Tech. Dig. IEEE International Electron Devices Meeting*, Dec 2000.
[10] T.-Y. Chiang, B. Shieh, and K. Sarawat, "Impact of Joule heating on scaling of deep sub-micron Cu/low-k interconnects," in *Symposium on VLSI Technology, Digest of Technical Papers*, Jun 2002, pp. 38–39.
[11] P. Wilkerson, A. Raman, and M. Turowski, "Fast, automated thermal simulation of three-dimensional integrated circuits," in *Inter Society Conference on Thermal Phenomena*, vol. 1, Jun 2004, pp. 706–713.
[12] F. Chen, J. Gill, and D. Harmon, "Measurements of effective thermal conductivity for advanced interconnect structure with various composite low-k dielectrics," in *International Reliability Physics Symposium*, 2004.
[13] A. H. Ajami, M. Pedram, and K. Banerjee, "Effects of non-uniform substrate temperature on the clock signal integrity in high performance designs," in *Proc. IEEE Custom Integrated Circuits Conf.*
[14] K. Banergee, A. H. Ajami, and M. Pedram, "Analysis and optimization of thermal issues in high-performance VLSI," in *Proc. Int. Symp. on Physical Design*, April 2001, pp. 230–237.
[15] K. D. Boese and A. B. Kahng, "Zero-skew clock routing trees with minimum wirelength," in *Proc. IEEE Int. Conf. ASIC*.
[16] J. Cong, A. B. Kahng, C.-K. Koh, and C.-W. A. Tsao, "Bounded-Skew Clock and Steiner Routing," *ACM Trans. on Design Automation of Electronics Systems*, vol. 4, no. 1, Jan 1999.
[17] R.-S. Tsay, "An Exact Zero Skew Clock Routing Algorithm," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 12, pp. 242–249, Feb 1993.
[18] T. Wang and C. C. Chen, "3D Thermal-ADI: A linear-time chip level transient thermal simulator," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 21, no. 12, Dec 2002.
[19] http://vlsicad.ucsd.edu/GSRC/bookshelf/Slots/BST/.
[20] T. Wang and C. Chen, "Power-Delivery Networks Optimization with Thermal Reliability Integrity," in *Proc. Int. Symp. on Physical Design*, April 2004.