Temperature-Aware Clock Tree Synthesis Considering Spatiotemporal Hot Spot Correlations

ChunChen Liu and Junjie Su
Electrical and Computer Engineering Department
University of California, San Diego
cki084@ucsd.edu and jusu@ucsd.edu

Yiyu Shi
Electrical Engineering Department
University of California, Los Angeles
yshi@ee.ucla.edu

ABSTRACT
Temperature variation in microprocessors is a workload dependent problem. In such a design, the clock skew should be minimized with respect to temperature variation. Existing work has studied clock tree embedding perturbation considering time variant temperature variation. There is no existing method that can reduce skew variation. This paper develops an efficient yet effective simultaneous hotspot avoid embedding and thermal aware routing (TMST) method, where hotspot embedding avoid tree topology located in area with high temperature possibility and thermal aware routing reduce skew in tree path with more smooth temperature area. With a thermally tolerable tree structure, our method can reduce not only delay skew but also skew variation (skew violation range). Compared with existing temperature-aware clock tree method, our TMST solution reduces skew variation by 2X compared with the Greedy-DME (GDME) method of Edahiro and existing thermal aware clock synthesis TACO and PECO.

With the scale from 100 down to 1 temperature maps, our TMST also guarantees the smallest wire length overflow. TMST reduces the worst case skew up to 4X than PECO and 5X than TACO.

I. INTRODUCTION
Clock synthesis has been extensively studied. Given a set of sinks (flip-flops), clock synthesis finds an abstract topology, embedding and minimizes mismatch of arrival times (i.e., skew) between sinks. Wire length minimizations under zero or bounded skew constraint were presented in [1], [2], [4]. However, process, supply voltage, and temperature variation induced skews were not considered. The exponential advance of very large scale integration (VLSI) technology has resulted in a high and non-uniform power dissipation over chips [4], which leads to a temperature gradient, i.e., spatial temperature variation over a chip that can cause non-negligible delay differences in both interconnects and devices. As clock signals are globally routed over chip, the temperature gradient can bring significant skew variations [5]. Therefore, the traditional constant zero/bounded skew routing methods become invalid.

The history of clock tree synthesis under thermal variation is short. TACO [5] suggests to construct a tree that balances the skew under the two given static thermal profiles (uniform and worst). The argument is that the skew optimized under one thermal profile will vary significantly under the other profile, thus the need to balance. TACO first builds a nonbuffered, zero-skew clock tree under a uniform thermal profile and refines it under worst-case (non-uniform) profile.

Since the thermal gradient is changing over time while the chip is running, a transient analysis/optimization is an extremely difficult task, if possible. Considering the skew caused by the spatial variation of temperature, PECO [5] is the only existing clock synthesis method considering time-variant variation by perturbation of embedding points.

With further merging points, PECO causes more wire length overhead. PECO has not discussed the skew variation issue, neither.

The skew variation problem can not be ignored in currently very large scale integration (VLSI) technology. A smaller skew variation region, can lead to a more stable system and a more accurate timing analysis. This paper focuses on temperature variation induced skew and skew variation minimization. We propose a simultaneous hotspot avoid embedding and thermal aware routing (TMST) method to minimize skew variation and the worst case skew considering the on-chip temperature variation and spatial correlation with respect to operating time. The main contributions of this paper are as follow: First, without any extra components like crosslinks or buffers insertion, our proposed synthesis guarantee smaller wire-length which leads to small power consumption. Second, we build up clock tree embedding according to temporal and spatial temperature information makes our tree has smaller dynamic skew range (skew variations), our paper is the pioneer...
one discussing this issue. Third, our method considers time-
variant temperature variation using different kind of delay
models. Among all the test, our TMST reaches not only the
smallest skew but the smallest skew variation.

The rest of this paper is organized as below: Section 2
shows the notation and problem statement in this paper.
Section 3 presents temperature modeling and delay model.
Section 4 sums up the overall algorithms. We show our
experimental results in Section 5 and conclude the paper in
Section 6.

II. NOTATION AND PROBLEM STATEMENT

Table 1 shows the notations and definitions used in the
following sections. Clock skew minimization is an important
design task to ensure correct circuit behavior of sequential
circuits, and many works have been done to design zero-
skew clock tree. The main problem of design uncertainty
is error of capacitance estimation. Even if we reduce design
uncertainty by using detailed analysis and optimization, clock
skew occurs due to manufacturing and environmental variabil-
ities. This paper mainly focuses on temperature issue. There
have been different proposed methods such as buffers and
crosslink insertion for reducing skew. However, with further
wire length or analogy elements, both crosslink and buffer
insertion methods significant increase power consumption.
Without any extra elements, our method reduces the skew by
changing the essential routing path to further compensate skew
and skew variation. We build up the entire tree topology and
building merging points on hot spot areas. High correlated area guarantees the same trend of temperature
routing through and building merging points on hot spot areas.

For the SPEC2000 benchmarks [9] from the SPEC2000 benchmark
is error of capacitance estimation. Even if we reduce design
issues. This paper mainly focuses on temperature issue. There
different proposed methods such as buffers and
crosslink insertion for reducing skew. However, with further
wire length or analogy elements, both crosslink and buffer
insertion methods significant increase power consumption.
Without any extra elements, our method reduces the skew by
changing the essential routing path to further compensate skew
and skew variation. We build up the entire tree topology and
building merging points on hot spot areas. High correlated area guarantees the same trend of temperature
routing through and building merging points on hot spot areas.

Given a clock distribution topology generated by GDME,
under the constraints below, we reembedding a new tree which
has high correlated routing paths and merging points.

- 1: We only implement our method with highly correlated
sink pairs and keep the same path with GDME.
- 2: We build up our clock tree using a bottom-up fashion.
- 3: We avoid building our merging points and routing
paths on Hot spot areas.
- 4: All routing paths and merging points are built on the
highly correlated areas.

III. MODELING

A. Stochastic Temperature Model and Correlation Map

The same temperature modeling in [9] is assumed in this
work. In order to make the paper self-contained, we first
review the temperature, gradient and correlation model. The
temperature gradient and correlation are obtained by a micro-
architecture level power and temperature simulation [11],
where the Alpha architecture is used as an example.

The overall chip is divided into a uniform grid with total
$N$ nodes. By applying six different applications (ammp, art,
compress, equake, gzip, gcc) from the SPEC2000 benchmark
in a sequence (each with a time-period $t_p$), the thermal-power
is obtained by averaging the cycle-accurate (scale of ps)
dynamic power in the thermal-constant scale (scale of ms).
Using this time-variant thermal power as input, the transient-
temperature $T(t_i, n_j)$ over the chip is calculated at different
time-instant $t_i$ for each node $n_j$ in the grid. To automatically
extract correlation for temperature variations, the temperatures
at $N$ nodes are modeled by random processes. Each node is
described by a temperature sequence sampled at $N'$ time-
instants,

$$F_{n_1} = \{T(t_1, n_1), ..., T(t_{p', n_1}), T(t_{p'+1, n_1}), ..., T(t_{N', n_1})\}$$
$$F_{n_2} = \{T(t_1, n_2), ..., T(t_{p', n_2}), T(t_{p'+1, n_2}), ..., T(t_{N', n_2})\}$$
$$...$$
$$F_{n_N} = \{T(t_1, n_N), ..., T(t_{p', n_N}), T(t_{p'+1, n_N}), ..., T(t_{N', n_N})\}$$

a temperature (spatial) correlation matrix is defined by

$$C(i,j) = \frac{cov(i,j)}{\sigma_i \cdot \sigma_j}, \quad (1)$$

where

$$cov(i,j) = \frac{\sum_{k=1}^{N'} T(t_k, n_i)T(t_k, n_j) - \sum_{k=1}^{N'} T(t_k, n_i) \sum_{k=1}^{N'} T(t_k, n_j)}{L} \quad (2)$$

is the co-variance matrix between nodes,

$$\sigma_i = \sqrt{\frac{\sum_{k=1}^{N'} T(t_k, n_i)^2}{N'}}, \quad \sigma_j = \sqrt{\frac{\sum_{k=1}^{N'} T(t_k, n_j)^2}{N'}} \quad (3)$$

are the standard deviations for nodes $n_i$ and $n_j$, and

$$\hat{T}_i = \frac{\sum_{k=1}^{N'} T(t_k, n_i)}{N}, \quad \hat{T}_j = \frac{\sum_{k=1}^{N'} T(t_k, n_j)}{N} \quad (4)$$

are mean-temperature for nodes $n_i$ and $n_j$ respectively. The
correlation coefficients $C(i,j)$ can be precomputed and stored
into a table. Figure 1 shows the distribution of one calculated
correlation matrix. The averaged correlation is about 0.8 for
the SPEC2000 benchmarks [9].

![Fig. 1. Temperature correlation matrix from running sequent of SPEC2000 benchmarks.](image-url)
B. Skew Model with Temperature Variation

As suggested by [5], when a clock wire experiences a temperature gradient, the unit-length resistance \( r_{\text{unit}} \) is as follows,

\[
r_{\text{unit}}(x,y,t) = \rho_0 \cdot \left[ 1 + \beta \cdot T(t,x,y) \right],
\]

where \( \rho_0 \) is the unit-length resistance at 0°C, and \( \beta \) is the temperature coefficient of resistance \((1/\text{oC})\). When the embedding path \( d(M^i,s_k) \) is fixed, we calculate the new resistance by

\[
R(M^i,s_k) = \sum_{\forall e \in d(M^i,s_k)} E[r_{\text{unit}}(e)] \cdot \text{len}(e)
\]

where \( E[r_{\text{unit}}(x,y)] \) is the mean value of resistance in edge \( e \) \((M^i,s_k)\).

Following the conventional definition for the propagation delay, the delay from the source node \( s_0 \) to sink \( s_i \), \( D_{(s_0 \rightarrow s_i)} \), is the time required for the node voltage (waveform) to pass 100% of the peak voltage under the impulse excitation in the source node. After obtaining the source to sink delay of \( j \)-th routing configuration \( \text{Conf}_j^i \) in level \( i \), we can calculate the worst case skew corresponding to \( \text{Conf}_j^i \) as follows

\[
\text{Skew}_i = \max_{\forall \text{sink } s_k} D_{(s_0 \rightarrow s_k)} - \min_{\forall \text{sink } s_k} D_{(s_0 \rightarrow s_k)}. \tag{7}
\]

The worst-case skew is then determined by those preserved routing path from all levels.

C. Problem Formulation

The simultaneous hotspot avoid embedding and thermal aware routing (TMST) problem is formulated as follows,

Formulation 1: (Simultaneously hotspot avoid embedding and thermal aware routing (TMST)) Given source \( s_0 \), sinks \( s_1 \cdots s_n \), an initial clock tree embedding, and a set of temperature variation maps, find proper re-embedding (including merging point and re-routing) for the new tree to minimize the worst case skew under the given temperature maps.

IV. ALGORITHMS

A. Overall Algorithm

Given a GDME-initialized clock tree construction, the re-embedding by thermal aware Maze routing is performed. The worst-case skew and re-embedding are determined in a bottom-up fashion. At each level, the merging points and routing paths are picked according to their correlation strength. Then the resulting routing path are routed through with strong correlation area, and only those sink paths that could cause large skew changes (high correlated) are selected for re-embedding.

In summary, the overall algorithm is shown in Figure 2, and the algorithm’s pseudo code is as presented in algorithm 1:

B. Correlation Cofactor

Smoother routing path guarantees more stable skew variation since the dynamic temperature variation over time generate different skews. The Thermal Aware Routing Topology Optimization (TMST) is an effective algorithm that considers the time variant temperature variations with spatial and temporal correlation. TMST generates a temperature correlation map by analyzing time variant temperature maps, and avoid the hot spots that are indicated in the temperature correlation map of the clock tree structure. Without using merging point perturbation, we use thermal aware routing to balance the skew, which can still keep the same or similar Manhattan distance. First, we build a macro model for temperature variations to get temperature maps for various timestamps. To model such on-chip time variant temperature, we impose a grid onto the chip and each grid is assigned a temperature range. This temperature range can be obtained by measurement or thermal simulation. A complete instruction set is tested and the corresponding \( K \) temperature profiles are obtained. The overall temperature variation can then be obtained based on
A modified embedded clock tree

Output: A modified embedded clock tree $T'$ (Bottom up embedding from the second last level to the second level)

1: Acquire temporal variant spatial correlation map
2: while $L_i! = \text{Level do}$
3: Generate initial embedding tree by GDME
4: while $p_i! = PC_i$ do
5: Find the merging segment
6: Choose the largest correlation weight area but not hotspot as the merging point
7: Apply thermal aware Maze routing to find the highest correlated path.
8: $j = j + 1$
9: end while
10: $i = i + 1$
11: end while
12: return $T'$

Algorithm 1 TMST

Input: Source $s_0$, sinks $s_1, \cdots, s_n$

Output: A modified embedded clock tree $T'$ (Bottom up embedding from the second last level to the second level)

1: Acquire temporal variant spatial correlation map
2: while $L_i! = \text{Level do}$
3: Generate initial embedding tree by GDME
4: while $p_i! = PC_i$ do
5: Find the merging segment
6: Choose the largest correlation weight area but not hotspot as the merging point
7: Apply thermal aware Maze routing to find the highest correlated path.
8: $j = j + 1$
9: end while
10: $i = i + 1$
11: end while
12: return $T'$

C. Thermal Aware Maze With Dynamic Window

High temperature variability greatly impacts the delay and skew of the zero-skew clock tree. Figure 3 shows that different routing paths with different temperature gradients will have different skews. The high temperature variance increases the delay by 100%, i.e., from 2ns to 4ns. In Figure 3, on the one hand, we can see the subregion with high temperature. On the other hand, we also notice that we can find the path with smooth temperature gradient but keep the short wire length as well.

Routing is a crucial step in clock tree synthesis design. Most of existing thermal aware clock tree synthesis uses roughly Manhattan distance to simulate and evaluate the delay between each two nodes [6], [9]. In real case, Manhattan distance is not exactable when considering time-variant temperature maps if our objective function consider both weight of temperature and distance parameters. We use thermal aware routing Maze Routing with considering both distance and temperature correlation cofactor in this paper, which not only is an accurate method for evaluating the delay between each node but also can reduce the skews and skew variation significantly. However, not all path with high correlation lead better thermal tolerance ability. With two fully components temperature difference path sometimes cause better skew reduction. In order to choose suitable path candidates, we only target path according to sink point correlation strength in this paper.

D. Overview

As shown in Figure 4(a), we get the initial tree from GDME topology. Unlike GDME, we build the small dynamic window with each candidate pair before designing merging point and routing path. Dynamic windows are small grids with weight of spatiotemporal information. We determine the routing paths and merging paths based on dynamic windows level by level as shown in Figures 4(c) and (d).

Figure 5 presents a short overview on how to build up the dynamic windows. The green spots are the sinks of the clock tree, and the stars are the hotspot areas.

For grid $j$, its correlation coefficient related to sink $k$ is 0.25. Its correlation coefficient related to sink $i$ is also 0.25. The sum of these two coefficients is the correlation cofactor between these two sinks, which is 0.5 in this case. Only the correlation coefficients of the grids that are inside of the dynamic window of two sinks are calculated. After obtaining the correlation coefficients, we find the merging segment by calculating the Manhattan distance. We determine the merging point by avoiding the hot spots and search for the grid with the highest correlation coefficient on the merging segment. In this grid map, the highest correlation coefficient
The interconnect has unit resistance under the Elmore delay model with no temperature variation. The initial zero-skew tree is constructed by the GDME [4] method. Industry cases JPEG1 and JPEG3 are tested in this paper. The benchmarks are the same as those in TACO [6]. Two more benchmarks r1-r5 which are the same benchmarks in TACO and PECO. We employ the standard clock-tree benchmarks r1-r5 which are the same benchmarks in TACO and PECO. Two more industry cases JPEG1 and JPEG3 are tested in this paper. The initial zero-skew tree is constructed by the GDME [4] method under the Elmore delay model with no temperature variation. The interconnect has unit resistance $r_0 = 0.1\Omega/\mu m$, unit capacitance $c_0 = 2.0 \times 10^{-16} F/\mu m$, and temperature sensitivity $\beta = 0.0068$. The above interconnect parameters are the same as those in TACO [6].

A chip with size of 6cm² is divided into a uniform grid to obtain the distributed temperature map by a macro-architecture level cycle-accurate power /temperature simulator [11]. Our experiments use six SPEC2000 applications (art, ammp, compress, equake, gcc, and gzip). We collect 100 temperature maps by simulating these applications in a sequence and recording temperature maps for every 10 million clock cycles after fast forwarding of 1 billion cycles. These applications lead to a temperature variation about 50°C over the 100 temperature maps. We then find regions of hot spot with high temperature variation (variance), and extract correlation matrix, i.e., covariance for pairwise regions. Below, we report the minimum-cost bounded-skew routing tree problem under the pathlength (linear) delay models in table II. Compared with existing temperature-aware clock tree method, our TMST solution reduces skew variation by 2X compared with the Greedy-DME (GDME) method of Edahiro and existing thermal aware clock synthesis TACO and PECO. With the scale from 100 down to 1 temperature maps, our TMST also guarantees the smallest wirelength overflow. TMST reduces worst case skew up to 4X than PECO and 5X than TACO.

### A. Comparison With Existing Techniques

We show the skew comparison with linear delay model in table II. GDME gets zero skew since it build up tree in H shape. TMST gets the best pathlength skew delay in all comparison. Table 4 shows the run-time comparison between different algorithms. We do detail routing with each two connecting pairs instead of roughly Manhattan distance, and further accurate high order delay model, TMST consumes much run time cost compared with other existing methods.

Table V compares our proposed TMST with the existing techniques GDME [3], and thermal aware clock synthesis PECO [9] and TACO [6] with SPICE. We scale from 100...
TABLE V

<table>
<thead>
<tr>
<th>Input(node)</th>
<th>wirelength(m)</th>
<th>w-skew-SPICE (100-map)</th>
<th>wirelength over(%)</th>
<th>TMST skew imprv(X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>r1(267)</td>
<td>1.32</td>
<td>1.32</td>
<td>1.33</td>
<td>145.1</td>
</tr>
<tr>
<td>r2(598)</td>
<td>2.59</td>
<td>2.62</td>
<td>2.62</td>
<td>534.5</td>
</tr>
<tr>
<td>r3(862)</td>
<td>3.38</td>
<td>3.43</td>
<td>3.43</td>
<td>582.3</td>
</tr>
<tr>
<td>r4(1903)</td>
<td>6.81</td>
<td>6.84</td>
<td>6.84</td>
<td>1424</td>
</tr>
<tr>
<td>r5(3101)</td>
<td>10.2</td>
<td>10.3</td>
<td>10.3</td>
<td>3533</td>
</tr>
<tr>
<td>JPEG1(3603)</td>
<td>9.31</td>
<td>9.33</td>
<td>9.33</td>
<td>1447</td>
</tr>
<tr>
<td>JPEG2(10.3K)</td>
<td>43.1</td>
<td>44.3</td>
<td>44.3</td>
<td>4755</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input(node)</th>
<th>wirelength(m)</th>
<th>w-skew-SPICE (50-map)</th>
<th>wirelength over(%)</th>
<th>TMST skew imprv(X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>r1(267)</td>
<td>1.32</td>
<td>1.33</td>
<td>1.33</td>
<td>144.2</td>
</tr>
<tr>
<td>r2(598)</td>
<td>2.59</td>
<td>2.62</td>
<td>2.61</td>
<td>527.8</td>
</tr>
<tr>
<td>r3(862)</td>
<td>3.38</td>
<td>3.44</td>
<td>3.44</td>
<td>581.4</td>
</tr>
<tr>
<td>r4(1903)</td>
<td>6.81</td>
<td>6.85</td>
<td>6.82</td>
<td>1423</td>
</tr>
<tr>
<td>r5(3101)</td>
<td>10.2</td>
<td>10.5</td>
<td>10.3</td>
<td>3521</td>
</tr>
<tr>
<td>JPEG1(3603)</td>
<td>9.31</td>
<td>9.34</td>
<td>9.33</td>
<td>1438</td>
</tr>
<tr>
<td>JPEG2(10.3K)</td>
<td>43.1</td>
<td>43.3</td>
<td>43.3</td>
<td>4750</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input(node)</th>
<th>wirelength(m)</th>
<th>w-skew-SPICE (20-map)</th>
<th>wirelength over(%)</th>
<th>TMST skew imprv(X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>r1(267)</td>
<td>1.32</td>
<td>1.35</td>
<td>1.35</td>
<td>143.2</td>
</tr>
<tr>
<td>r2(598)</td>
<td>2.59</td>
<td>2.62</td>
<td>2.61</td>
<td>519.1</td>
</tr>
<tr>
<td>r3(862)</td>
<td>3.38</td>
<td>3.44</td>
<td>3.39</td>
<td>579.5</td>
</tr>
<tr>
<td>r4(1903)</td>
<td>6.81</td>
<td>6.85</td>
<td>6.85</td>
<td>1420</td>
</tr>
<tr>
<td>r5(3101)</td>
<td>10.2</td>
<td>10.5</td>
<td>10.3</td>
<td>3500</td>
</tr>
<tr>
<td>JPEG1(3603)</td>
<td>9.31</td>
<td>9.34</td>
<td>9.33</td>
<td>1427</td>
</tr>
<tr>
<td>JPEG2(10.3K)</td>
<td>43.1</td>
<td>43.3</td>
<td>43.3</td>
<td>4755</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input(node)</th>
<th>wirelength(m)</th>
<th>w-skew-SPICE (2-map)</th>
<th>wirelength over(%)</th>
<th>TMST skew imprv(X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>r1(267)</td>
<td>1.32</td>
<td>1.35</td>
<td>1.35</td>
<td>142.8</td>
</tr>
<tr>
<td>r2(598)</td>
<td>2.59</td>
<td>2.62</td>
<td>2.62</td>
<td>511.1</td>
</tr>
<tr>
<td>r3(862)</td>
<td>3.38</td>
<td>3.43</td>
<td>3.43</td>
<td>579.5</td>
</tr>
<tr>
<td>r4(1903)</td>
<td>6.81</td>
<td>6.88</td>
<td>6.88</td>
<td>1431</td>
</tr>
<tr>
<td>r5(3101)</td>
<td>10.2</td>
<td>10.8</td>
<td>10.3</td>
<td>3523</td>
</tr>
<tr>
<td>JPEG1(3603)</td>
<td>9.31</td>
<td>9.34</td>
<td>9.32</td>
<td>1427</td>
</tr>
<tr>
<td>JPEG2(10.3K)</td>
<td>43.1</td>
<td>43.4</td>
<td>43.4</td>
<td>4732</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input(node)</th>
<th>wirelength(m)</th>
<th>w-skew-SPICE (1-map)</th>
<th>wirelength over(%)</th>
<th>TMST skew imprv(X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>r1(267)</td>
<td>1.32</td>
<td>1.34</td>
<td>1.34</td>
<td>144.1</td>
</tr>
<tr>
<td>r2(598)</td>
<td>2.59</td>
<td>2.61</td>
<td>2.63</td>
<td>521.1</td>
</tr>
<tr>
<td>r3(862)</td>
<td>3.38</td>
<td>3.41</td>
<td>3.41</td>
<td>586.3</td>
</tr>
<tr>
<td>r4(1903)</td>
<td>6.81</td>
<td>6.86</td>
<td>6.86</td>
<td>1397</td>
</tr>
<tr>
<td>r5(3101)</td>
<td>10.2</td>
<td>10.4</td>
<td>10.3</td>
<td>3500</td>
</tr>
<tr>
<td>JPEG1(3603)</td>
<td>9.31</td>
<td>9.35</td>
<td>9.32</td>
<td>1427</td>
</tr>
<tr>
<td>JPEG2(10.3K)</td>
<td>43.1</td>
<td>44.3</td>
<td>44.1</td>
<td>4741</td>
</tr>
</tbody>
</table>

VI. CONCLUSIONS AND FUTURE WORK

Existing clock tree and hybrid clock synthesis did not consider extra skew caused by temperature variation, or [8] needed to assume a time-invariant worst-case temperature map to find the worst-case skew. These papers also use Elmore delay model to evaluate clock skew. In this paper we have made the first attempt to reduce skew variation considering time variant temperature variation. We have developed a minimal skew clock tree embedding that considers temporal and spatial correlation issue. Thermal aware routing topol-ogy optimization is used to avoid hot-spots and compensate temperature variation. With the entire tree in the smoother temperature areas, our tree synthesis can significant reduce the skew variation and worst case skew. Compared with existing temperature-aware clock tree methods, our TMST solution reduces skew variation by 2X compared with the Greedy-DME (GDME) method of Edahiro and existing thermal aware clock synthesis TACO and PECO. With the scale from 100 down to 1 temperature maps, our TMST also guarantees the smallest wirelength overflow. TMST reduces worst case skew up to 4X than PECO and 5X than TACO.

maps down to 1 temperature map. Our TMST has the best skew reduction in all of testing cases.

Table V also reports wirelength for different algorithms, where runtime includes the time to build macromodel for PECO and TMST.

We present the skew violation distribution of r4 in Figure 7 and Figure 8, note that our TMST has not only small worst case skew but also small skew variation.

Figure 9 shows the clock tree built by our TMST method in JPEG1 case. Significant bending in the route between nodes can be seen.

VI. CONCLUSIONS AND FUTURE WORK

Existing clock tree and hybrid clock synthesis did not consider extra skew caused by temperature variation, or [8] needed to assume a time-invariant worst-case temperature map to find the worst-case skew. These papers also use Elmore delay model to evaluate clock skew. In this paper we have made the first attempt to reduce skew variation considering time variant temperature variation. We have developed a minimal skew clock tree embedding that considers temporal and spatial correlation issue. Thermal aware routing topology optimization is used to avoid hot-spots and compensate temperature variation. With the entire tree in the smoother temperature areas, our tree synthesis can significant reduce the skew variation and worst case skew. Compared with existing temperature-aware clock tree methods, our TMST solution reduces skew variation by 2X compared with the Greedy-DME (GDME) method of Edahiro and existing thermal aware clock synthesis TACO and PECO. With the scale from 100 down to 1 temperature maps, our TMST also guarantees the smallest wirelength overflow. TMST reduces worst case skew up to 4X than PECO and 5X than TACO.
VII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the contribution of professor Andrew Kahng’s research group from UCSD, professor Lei He from UCLA and reviewers’ comments.

REFERENCES