

Thermal Aware Clock Synthesis Considering Stochastic Variation and Correlations

Chunchen Liu[†], Ruei-Xi Chen[‡], Jichang Tan[‡], Sharon Fan[§], Jeffrey Fan[§], Kia Makki[§]

[†]Department of Electrical and Computer Engineering, University of California, San Diego, California

[‡]Department of Computer Science and Information Engineering, St. John's University, Taipei, Taiwan

[§]Department of Electrical and Computer Engineering, Florida International University, Miami, Florida

Abstract—In this paper, we have proposed a thermal aware routing based parameterization to generate a clock model that takes the stochastic temperature variation into consideration. The worst-case skew is calculated by using the transient analysis. The proposed Thermal Aware Routing based Clock Tree Optimization (TARCO) is performed level by level via the use of sensitivities from the proposed macromodel. The experimental results show that our algorithm reduces up to 5X worst-case skew in comparison to the existing bonded-skew based Deferred Merge Embedding (DME) approach, and 1.7X worst-case skew compared to Perturbation based Clock Optimization (PECO) and 2.1X compared to Temperature Aware Clocktree Optimization (TACO).

I. INTRODUCTION

The design of the clock distribution network has become an increasingly challenging problem for VLSI designs. The main goal in designing the clock network is to ensure that it is zero skew or low skew, since large skew in a clock network will degrade the chip frequency or even cause functional errors. As we enter the deep sub-micro era, the variations of Process, Supply Voltage, and Temperature (PVT) have a big impact on the circuit performance. The original carefully designed zero skew clock networks may suffer from large skew due to PVT variations. Nevertheless, uniform temperature is assumed in most of existing work on clock optimization [1].

In consideration of the non-uniform on-chip temperature distribution, [2] and [3] are the pioneers to propose the thermal-aware clock tree optimization. The concept of merging diamond was introduced in finding the skew balance region between uniform and non-uniform thermal profiles. However, the temporal variability of on-chip temperature was not considered in details in [2] and [3]. However, the variability of on-chip temperature cannot be ignored, especially when different applications or data sets are applied to the same chip.

In this paper, we propose a clock optimization algorithm to minimize the worst case temperature induced skew consider the on-chip temperature variation. In addition, accurate delay model is assumed by performing transient time analysis at sinks of the impulse respond. Without using merging point perturbation, we use thermal aware routing to balance the skew which minimizes potential serious wire-length overflow.

II. STOCHASTIC THERMAL MODEL

To model the temporal variability of on-chip temperature, we intentionally impose a geological grid onto the chip and

each grid is assigned by a temperature range. This temperature range can be obtained through measurement or thermal simulation. Under such grid, this paper intends to optimize the worst-case temperature induced skew.

Thus, the overall temperature variation can then be extracted based on the temperature profile, say $K^{\circ}C$. The possible temperature range in each sub-region can be obtained and expressed as

$$RT^i = [T_{min}^i, T_{max}^i], \forall \text{ sub-region } R_i \quad (1)$$

Based on [4], in consideration of an interconnect experiencing temperature profile $T(x, y, t)$ along its length in time t , the resistance will change linearly with temperature. The unit length resistance r_{unit} of the clock wire segment can be calculated as

$$r_{unit}(x, y, t) = \rho_0 \cdot [1 + \beta \cdot T(x, y, t)] \quad (2)$$

where ρ_0 is the unit length resistance at $0^{\circ}C$, and β is the temperature coefficient of resistance ($1/^{\circ}C$). Also, note that the unit length capacitance does not change with temperature variations along the interconnect length [4].

In this paper, we are trying to find hotspot regions within K temperature profile statistically and search for the routing path avoiding these regions. That is:

Definition 1: (Thermal Aware Routing based Clock Tree Optimization - TARCO) Given source Src , sinks $s_1 \cdots s_n$, abstract topology of the clock tree and temperature variation profile, find an embedded clock tree which minimizes the worse case skew under the temperature variation.

III. THERMAL AWARE ROUTING

Routing is a crucial step in clock tree synthesis design. Most of the existing thermal aware clock tree synthesis uses roughly Manhattan distance to simulate and evaluate the delay between each two nodes [2] [3]. In real case, Manhattan distance is not deterministic in consideration of time-variant temperature maps when the objective function is associated with temperature and distance parameters. We use thermal aware routing topology optimization considering both distance and temperature parameters in this paper. The idea is to evaluate the delay between each nodes as well as reduce the skews at the same time.

High temperature variance greatly impacts the delay and skew of zero-skew clock tree. Different routing path involving

temperature gradients will contribute different skews. Therefore, routing path with smooth temperature gradient has small skews because of small temperature variation. In this paper, based on routing topology optimization, we try to locate the routing path with smooth temperature gradient. We firstly construct an initial zero-skew clock tree by DME (Deferred Merged Embedded) algorithm under the uniform temperature profile to minimize the wire length. We then mark some sub-areas with greater possibility of higher temperature variation and avoid route through those areas. After that, we can locally adjust the routing path between two merging points to lower the temperature variance effect and keep short wirelength at the same time.

In this paper, we use maze algorithms [5] to avoid hot-spot areas in this paper. An initial zero-skew clock tree [6] constructed under uniform temperature profile is a good starting point of our proposed algorithm. Given an initial tree, we locally adjust the routing path in bottom-up fashion. A levelization is first performed on the *abstract topology* of the initial tree.

Definition 2: Level i ($i \geq 0$) in a given abstract topology T with source node n_{src} is a node set N_i , in which the length (number of edges in the path) of the path ($n \rightarrow n_{src}$) is i , $\forall n \in N_i$.

In each level N_i of the clock tree (except for the bottom level with all sinks), we try to change the location of the non-sink nodes in N_i within a range.

Definition 3: A Routing Path (RP) of node pair (s_1, s_2) is a new path, which locates at the bound of the cycle with radius r centering at the initial merging point of this node pair.

Algorithm 1 TARCO

Input: Source Src , sinks s_1, \dots, s_n , embedded tree T

Output: A modified embedded clock tree T'

- 1: $(Levels, N_{levels}) \leftarrow Levelize\ T$
 {Bottom up embedding from the second last level to the second level}
 - 2: **for** $i = N_{levels} - 1$ to 1 **do**
 - 3: $PC_i \leftarrow$ Thermal-aware Maze routing in level i
 - 4: **for** $\forall p_i^j \in PC_i$ **do**
 - 5: $R_i^j, C_i^j \leftarrow$ Extract R/C of the updated tree for p_i^j
 - 6: $Volt_{s_k}, \forall sink\ s_k \leftarrow$ Solve the system based on (6)
 - 7: $Dly_{Src \rightarrow s_k}, \forall sink\ s_k \leftarrow$ Calculate Source-Sink Delay based on Back-Euler method
 - 8: Calculate worse case skew under PC_i as follows

$$Skew_j = \max_{\forall Sink\ s_k} Dly_{(Src \rightarrow s_k)} - \min_{\forall Sink\ s_k} Dly_{(Src \rightarrow s_k)}$$
 - 9: **end for**
 - 10: Embedding in level $i \leftarrow (PC_i^* = \min_{\forall PC_i^j} Skew_j)$
 - 11: **end for**
 - 12: **return** T'
-

A. TARCO Routing Path

As mentioned above, the feasible routing path of sub node pairs S_i, S_j can be any point on the cycle centering at M .

Algorithm 2 Thermal-aware Maze Routing

Input: G-partitioned layout N-netlist information Hot regions

Output: A routing path considering Delay and avoid hot regions begin built a timing-driven routing tree for each net N :

- 1: **for** $i = N_{levels} - 1$ to 1 at coarsening stage **do**
 - 2: Hot regions estimation
 - 3: Escape routing
 - 4: **for** $i = N_{levels} - 1$ to 1 the un-coarsening stage **do**
 - 5: Maze routing algorithms()
 - 6: **end for**
 - 7: **end for**
-

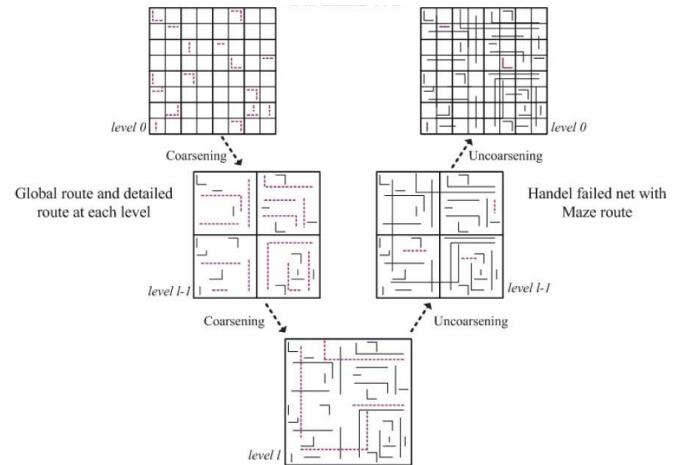


Fig. 1. Thermal-Aware Maze Routing

To reduce complexity, we adopt similar approach as other algorithms by sampling the points in the four Manhattan directions (up, down, left, right) from M [2][3]. The radius of each direction D_i in this Min-thermal Routing Path (MRP) cycle is decided based on the local temperature gradient in D_i . We use the larger radius for the sharper temperature gradient.

Intuitively, this strategy allows more searching space for bigger temperature variations. Basically, the following linear estimation is used to decide the feasible area radius r_p of M in direction D_i .

$$r_p(M) = \alpha \cdot grad(M, D_i) \quad (3)$$

where α is a constant and $grad(M, D_i)$ is the temperature gradient of M in direction D_i , which can be calculated based on the given temperature profiles. In our experiments, we find that it achieves the best runtime-performance tradeoff to choose the corner points M_1, \dots, M_4 in the routing path candidate set to generate MRP.

Given an MRP M' , we then decide the Embedding of Perturbation (EOP) of M' . Due to the combinatorial natural of paths, there exist factorial different EOPs from an MRP M' and a level i node s_i . To consider the on-chip temperature un-uniformity, we seeks an Efficient Path (EP) which has the least sensitivity to temperature variation among all possible paths $M' \rightarrow s_i$.

Since the standard deviations Std_j of each sub-region j can be obtained by the thermal simulation. The desired EOP, $EOP_d(M', s_i)$, is the shortest path in terms of the sum of square of standard deviations of the sub-regions that it passes (see Figure 2) as expressed below.

$$EOP_d(M', s_i) = \min_{\forall P \in M' \rightarrow s_i} \sum_{\forall e \in P} std(i)^2 \quad (4)$$

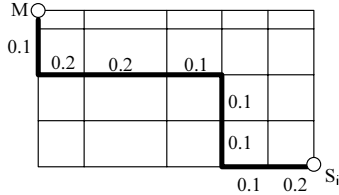


Fig. 2. Finding Efficient Path of Path $M' \rightarrow s_i$

When the EOP ($EOP_d(M', s_i)$) of MRP (M') is fixed, we calculate the new resistance based on (5).

$$R(M', s_i) = \sum_{\forall e \in EOP_d(M', s_i)} E[r_{unit}(e)] \cdot len(e) \quad (5)$$

where $E[r_{unit}(x, y)]$ is the mean value of temperature in edge e and $len(e)$ is the length of this edge.

B. TARCO Skew Calculations

1) *System Transfer Function:* In this section, we need to calculate the worst case skew for each configuration P_i^j (line 4 to line 9 in Algorithm 1). We calculate source to sinks delay based on their impulse voltage responds, which can be obtained as follows:

Given the initial tree T_{init} , the system transfer function of Modified Nodal Analysis (MNA) can be expressed as

$$(G_0 + sC_0)x = Bu \quad (6)$$

$$y = L_0^T x \quad (7)$$

where G_0 is the conductive matrix and C_0 is the capacitive matrix. Suppose there are N nodes in the tree. G_0 and C_0 are both $N \times N$ matrices. G_0 can be obtained by the incident matrix of the tree and C_0 is the diagonal matrix where diagonal elements are the lumped capacitive values of the tree nodes. Since the source node of the tree is the only input of the system, we have

$$B = [1, \overbrace{0, \dots, 0}^{N-1}]^T \quad (8)$$

where B is an $N \times 1$ matrix, u is an $N \times N$ identity matrix, and x is a length N vector corresponding to the voltage of the impulse respond of each tree node, including all sinks and Steiner points. We organize vector x in the following way:

$$x = [x_{s_1}, \dots, x_{s_n}, x_{S_{rc}}, x_{S_1}, \dots, x_{S_t}] \quad (9)$$

where $s_i, i = 1, \dots, n$ are sinks and $S_i, i = 1, \dots, t$ are Steiner points. Since we are interested in the voltage responds at sinks, we have

$$L_0 = \begin{bmatrix} I_{n \times n} \\ 0_{(N-n) \times n} \end{bmatrix} \quad (10)$$

The selected output vector y is

$$y = [y_{s_1}, \dots, y_{s_n}] \quad (11)$$

For each PC, PC_i , in level i , (6) can be re-written as

$$[G_0 + \Delta G_i + s(C_0 + \Delta C_i)] \cdot (x + \Delta x_i) = B \quad (12)$$

2) *Skew Calculations:* The expanded system expressed by (12) in time domain can be written as

$$G_{PC} + C_{PC} \cdot \frac{dX_{PC}(t)}{dt} = B_{PC} \quad (13)$$

$$Y_{PC}(t) = L_{PC} \cdot X_{PC}(t) \quad (14)$$

This time domain transient response can be solved by Back-Euler method. The system equation at time instant t with time step h is

$$G_{PC} + \frac{1}{h} \cdot C_{PC} \cdot X_{PC}(t) = \frac{1}{h} \cdot G_{PC} X_{PC}(t-h) + B_{PC} U(t) \quad (15)$$

$$Y_{PC}(t) = L_{PC} \cdot X_{PC}(t) \quad (16)$$

Following the conventional definition for the propagation delay, $Dly_{(src \rightarrow s_i)}$, from source node src to sinks s_i in a net, $Dly_{(src \rightarrow s_i)}$ is the time required for the node voltage to pass 80% of the peak voltage given the impulse excitation in the source node [7].

Starting from an initial solution, e.g $X = 0$ when $t = 0$, the system expressed by (15) and (16) is solved iteratively. In each iteration i , $t = h \cdot i$, Y_{PC}^i is solved based on the results obtained in the previous iteration. Since the impulse respond increase monotonically before reaching the peak voltage value, the time required to pass 80% of the peak voltage can be identified during iterations.

After obtaining the source to sink delay of each routing path configuration PC_i^j in level i , we can calculate the worst case skew corresponding to PC_i^j (line 8 in Algorithm 1) as follows

$$Skew_j = \max_{\forall sink_{s_k}} Dly_{(src \rightarrow s_k)} - \min_{\forall sink_{s_k}} Dly_{(src \rightarrow s_k)} \quad (17)$$

IV. EXPERIMENTAL RESULTS

The proposed algorithms have been implemented in C++ programming language and Matlab. We report runtime using a Linux workstation with 1.9GHz, P4-CPU, and 2GB memory. We employ standard clock-tree benchmarks r1-r5 in our experiments that are the same as PECO and TACO. The initial zero-skew tree is constructed by the DME method under the Elmore delay model without temperature variations.

TABLE I
WIRE-LENGTH COMPARISON

Input(node)	wirelength(μm)			
	DME	TACO	PECO	TARCO
r1(267)	1.32e06	1.48e06	1.37e06	1.34e06
r2(598)	2.56e06	2.71e06	2.61e06	2.48e06
r3(862)	3.38e06	3.77e06	3.44e06	3.29e06
r4(1903)	6.82e06	6.99e06	6.85e06	6.78e06
r5(3101)	1.02e07	1.22e07	1.05e07	1.04e07

TABLE II
SKEW COMPARISON

Input(node)	w-skew (100-map)(ps)			
	DME	TACO	PECO	TARCO
r1(267)	144.2	132	107.3	61.7
r2(598)	535.3	338	162.8	133.6
r3(862)	587.5	327	200.2	163.2
r4(1903)	1422.2	999	138.2	277.5
r5(3101)	3521.3	911.6	2321	924.6

The interconnect has unit resistance $r_0 = 0.03\Omega/\mu m$, unit capacitance $c_0 = 2.0 \times 10^{-16} F/\mu m$, and the temperature sensitivity of $\beta = 0.0068$. The above interconnect parameters are the same as those in [2].

A chip is divided into a uniform grid with 100×100 regions to obtain the distributed temperature map by a micro-architecture level cycle-accurate power/temperature simulator. Our experiments use six *SPEC*2000 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 $50^\circ C$ over the 100 temperature maps.

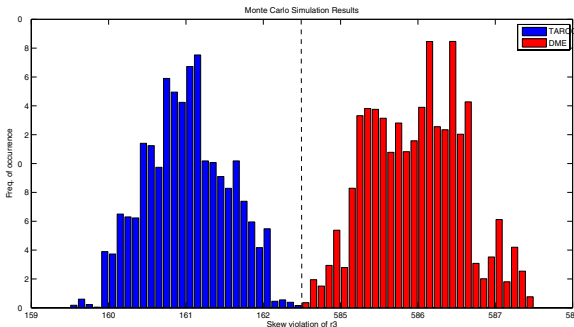


Fig. 3. Given 100 temperature maps, compare skew distribution of TARCO (blue) on benchmarks r3 with DME (red)

Table 1 and table 2 compare our proposed TARCO algorithm with the existing techniques DME, TACO [2], and perturbation based re-embedding PECO [3]. TARCO obtains the smallest worst-case skews for all r1-r5 circuits, and reduces the worst-case skew by up to 5.01X compared to DME, by up to 2.1X compared to TACO, and by up to 1.72X in comparison to the PECO algorithm.

All clock trees optimized by proposed TARCO algorithm

have smaller skews than those obtained by DME. The averaged skew reduction by TCO is 6.4X, and the skew distribution by TARCO is also much narrower.

Table 1 also reports wirelength and runtime for different algorithms, where runtime includes the time to build macro-model. The three skew reduction algorithms all have small amount of wirelength increase compared to DME. TARCO has almost the same wirelength as DME. But more importantly, TARCO has over 0.5% shorter wirelength than TACO and PECO. As to runtime, TARCO has a runtime longer than DME using Elmore model, but has a shorter runtime compared with TACO and PECO.

V. CONCLUSION

Existing clock tree synthesis did not consider the extra skew caused by the temperature variation in details. In this paper, we have developed a minimal skew clock tree embedding that considers the time-variant temperature variation with correlations. The proposed Thermal Aware Routing based Clock Tree Optimization is used to avoid hot-spots of temperature variation. The experimental results show that our algorithm reduces up to 5X worst-case skew compared to the existing bonded-skew based approach DME, 1.7X worst-case skew compared to PECO and 2.1X compared to TACO. In the future, we plan to extend our method to clock skew induced by process and supply-voltage variations.

REFERENCES

- [1] M. Edahiro, "Delay minimization for zero-skew routing," in *Proc. Intl. Conf. Computer-Aided Design*, pp. 563–566, 1993.
- [2] M. Cho, S. Ahmed, and D. Z. Pan, "Taco: Temperature aware clock-tree optimization," in *Proc. Intl. Conf. Computer-Aided Design*, 2005.
- [3] H. Yu, Y. Hu, C. Liu, and L. He, "Minimal skew clock embedding considering temperature variation," in *Proc. Intl. Symp. Physical Design*, 2007.
- [4] K. Banerjee, A. H. Ajami, and M. Pedram, "Analysis and optimization of thermal issues in high-performance vlsi," in *ISPD*, pp. 230–237, Apr. 2001.
- [5] c. Lee, "An algorithm for path connections and its applications," in *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, 1961.
- [6] A. B. Kahng and C.-W. A. Tsao, "More practical bounded-skew clock routing," in *Proc. Design Automation Conf.*, pp. 594–599, 1997.
- [7] D. S. Gao and D. Zhou, "Propagation delay in rlc interconnection networks," in *Proc. Int. Sym. on Circuits and Systems*, 1993.