

Buffer Sizing and Polarity Assignment in Clock Tree Synthesis for Power/Ground Noise Minimization

Hochang Jang, *Student Member, IEEE*, Deokjin Joo, and Taewhan Kim, *Senior Member, IEEE*

Abstract—In synchronous systems, clock tree causes high peak current at clock edges, increasing power/ground noise significantly, if the clock tree is not carefully designed. This paper addresses the problem of minimizing power/ground noise in the clock tree synthesis. Contrary to the previous approaches which only make use of assigning polarities to clock buffers to reduce power/ground noise, our approach solves a new problem of simultaneous consideration of assigning polarities to clock buffers and determining buffer sizes to fully exploit the effects of buffer sizing together with polarity assignment on the minimization of power/ground noise while satisfying the clock skew constraint. Specifically, the contributions of this paper are: 1) precisely estimating peak currents by clock buffers and reflecting them on the power/ground noise minimization; 2) proposing a pseudo-polynomial time optimal algorithm based on dynamic programming for solving the integrated problem, together with the proof of intractability of the problem; 3) devising a systematic design flow framework for reducing the power/ground noise over the entire chip; and 4) considering the effect of thermal variation on the clock skew bound and the noise minimization.

Index Terms—Buffer sizing, clock skew, clock tree synthesis, noise minimization, polarity assignment, thermal variation.

I. INTRODUCTION

AS THE SUPPLY voltage decreases in the modern very large scale integration design, the power and ground noise has a crucial effect on the circuit performance, such as the delay of switching signal [1], [2]. The maximum voltage drop of power/ground line is determined by the current peak [3], and the high current peak is driven by wrong or inappropriately dimensioned power/ground routings or peak current sources. The amplitude of peak current increases when numerous signals driven by neighboring sources switch simultaneously. In a synchronous circuit, the buffered clock tree incessantly consumes a considerable amount of current. In addition, the amount of resulting clock power on the clock distribution network and the clocked loads typically accounts

for one third to one half of the total chip power dissipation [4]. Since the clock buffers consume the current at the clock edges, a large amount of current is generated around the clock edges, which makes the clock buffers be one of the major sources of power/ground noise.

There have been several proposed works which have tried to reduce the peak current around the clock edges and the resulting power/ground noise. Benini *et al.* [3] proposed to schedule the clock arrival times of flip-flops (FFs) in order to disperse the peak current. Vittal *et al.* [5] then formulated the clock arrival time scheduling problem as a 0-1 integer linear program. Later, Huang, Chang, and Nieh [6] proposed a refined technique to reduce the computational expense of the 0-1 integer linear program. On the other hand, rather than scheduling the clock arrival times, Nieh, Huang, and Hsu [7] first proposed to assign positive polarity onto a half of clock buffers and negative polarity onto the remaining half of the clock buffers.¹ Fig. 1(a) illustrates the idea proposed in [7]. They equally partitioned the clock tree into two subtrees and replaced the buffer at the root of one subtree [i.e., the lower subtree in Fig. 1(a)] with an inverter so that when the clock signal switches from 0 to 1 (or 1 to 0), all the buffers on the upper subtree charge (or discharge) current from *VDD* (or to *GND*) while all the buffers on the lower subtree discharge (or charge) current to *GND* (or from *VDD*). Note that the FFs connected to the sink buffers in the lower subtree should be replaced with negative-edge triggered FFs. As implied by the clock tree structure in Fig. 1(a), even though this simple modification can reduce the total peak current over the chip upto the limit, it is not able to effectively reduce the power/ground noise in local regions. To overcome this limitation, Samanta, Venkataraman, and Hu [8] used the physical placement information of the buffering elements in determining buffers and inverters so that for local regions, roughly half of the buffering elements are assigned with positive polarity and the other half with negative polarity. Note that although this paper is able to reduce the power/ground noise greatly, sometimes it is likely to cause a long clock skew because the effect of the different delays of inverters and buffers on the clock skew have not been taken into account. [See Fig. 1(b) for illustration.]

Manuscript received January 17, 2010; revised April 23, 2010 and June 11, 2010; accepted July 27, 2010. Date of current version December 17, 2010. This work was supported by the Basic Science Research Program through the National Research Foundation Grant funded by the Korea Ministry of Education, Science and Technology, under Grant 2009-0091236. This paper was recommended by Associate Editor E. Young.

The authors are with the School of Electrical Engineering and Computer Science, Seoul National University, Seoul 151-744, Korea (e-mail: hochang@ssl.snu.ac.kr; jdj@ssl.snu.ac.kr; tkim@ssl.snu.ac.kr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCAD.2010.2066650

¹It is said that a buffer is assigned with a *positive polarity* or a *negative polarity* if its output switches in the same direction as or in the opposite direction to that of the clock source, respectively.

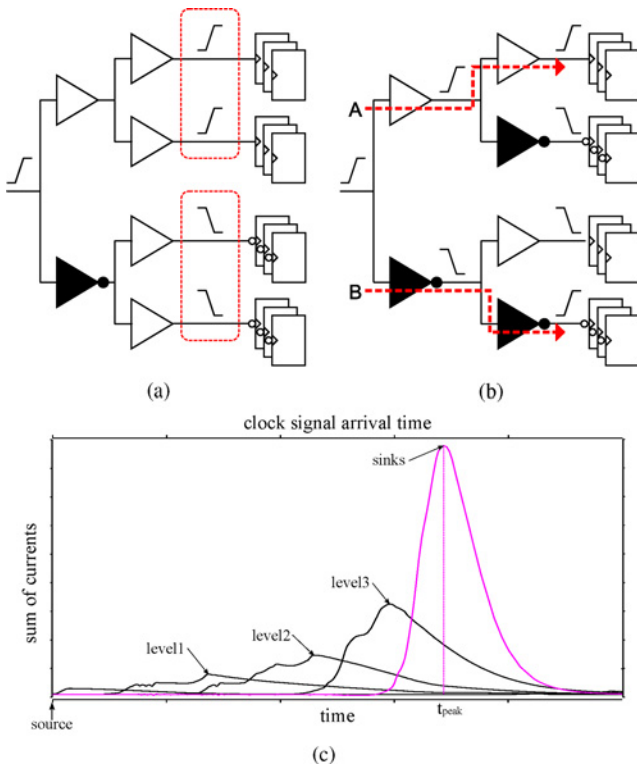


Fig. 1. Polarity assignment methods for reducing power/ground noise. (a) Polarity assignment by inserting one inverter [7]: it can cause a high increase of power/ground noise in some local regions. (b) Polarity assignment using placement information [8]: it can cause clock skew violation. (c) Peak current profile for a buffered clock tree of circuit S5378: the majority of current flow occurs at the time when the sink buffers switch where the x -axis indicates the time instances when a clock signal flows from the clock source to FFs passing through the intermediate buffers in the clock tree. The highest (red) curve on the right is caused by the current flow at the sink buffers while the other curves by the non-sink buffers (i.e., internal buffers).

Chen, Ho, and Hwang [9], [10] observed that the peak current occurs at the time when the clock signal arrives at the buffering elements (called sinks) that are directly incident to FFs, as validated by a SPICE simulation like that in Fig. 1(c). Thus, they proposed a method to assign polarities to the sinks, using the physical placement information of the sinks, with the objective of minimizing the power/ground noise while satisfying a minimum clock skew constraint. The optimization problem described in the work of [9] and [10] is well defined and the proposed solution is effective to overcome the limitations of the prior works [7], [8]. The approach [9], [10] partially handled the buffer sizing in that it has used, for assigning negative polarity, the inverter whose speed is the closest to the corresponding buffer. In addition, the approach by Ryu and Kim [11] placed a more weight on the power/ground noise minimization than the clock tree embedding, thus performed polarity assignment followed by clock tree construction. However, this approach used single types of buffer and inverter. Recently, Kang and Kim [12] considered the delay variations in the polarity assignment. They performed polarity assignment which minimizes the power/ground noise while meeting skew yield constraint. Lu and Taskin [13] performed the polarity assignment to non-sink buffering elements as well as sinks. They reduced the peak

current by using polarity assignment to non-sinks 5.5% further, but the clock skew is significantly sacrificed. The approaches in [12] and [13] still used single types of buffer and inverter.

This paper aims to completely eliminate the limitations of the previous works. Precisely, this paper *addresses a new problem of simultaneous consideration of assigning polarities to the sink buffering elements and determining the sizes of buffers and inverters to fully exploit the effects of the polarity assignment and buffer/inverter sizing on the minimization of power/ground noise while satisfying the clock skew constraint*. The contributions of this paper are summarized as: 1) precisely estimating peak current by clock buffers and inverters; 2) developing a pseudo-polynomial time optimal algorithm based on dynamic programming for the (NP-complete) problem; 3) proposing a framework of systematically utilizing the algorithm of (2) to reduce the power/ground noise over the entire chip; and 4) proposing an extended solution to the problem of considering the effect of thermal variation on the clock skew constraint and power/ground noise minimization. This paper is an extended version of the preliminary work in [14]. The extensions include the proof of the NP-completeness of the polarity assignment problem, the detailed analysis and description on the proposed algorithm, including experimental data, and a solid solution to the thermal-aware polarity assignment.

The remainder of this paper is organized as follows. Section II shows a few examples to illustrate the current profiles of several buffers and inverters, and motivation of the work. In Section III, we formulate the problem of simultaneous buffer sizing and polarity assignment, and show that the problem is NP-complete. In addition, we propose an optimal pseudo-polynomial time dynamic programming algorithm for solving the problem in a local region without considering the clock skew constraint. Then, in Section IV we propose a complete solution to the problem which consists of two phases. The first phase which finds a set of time intervals that satisfies the clock skew constraint for a whole region is presented in Section IV-A and the second phase which applies the algorithm in Section III to the time intervals obtained in Section IV-A to every local region and finds the solution of buffer sizing and polarity assignment with the least peak current is described in Section IV-B. Section IV-E describes an extended approach which exploits the two phases in Sections IV-A and IV-B to support the satisfaction of clock skew constraint due to the thermal variation. Section V then provides experimental results to show the effectiveness of the approach. Finally, concluding remarks are presented.

II. OBSERVATIONS

Each buffering element in a buffered clock tree can be implemented with either a non-inverting clock buffers (i.e., BUF) or an inverting clock buffers (i.e., INV). In general, a cell library provides multiple types of buffers and inverters in the clock tree synthesis to allow designer to choose buffers and inverters according to the design objectives and constraints. Fig. 2 shows the current profiles of two types of buffers BUFCKKHD and BUFCKLHD and two types of inverters

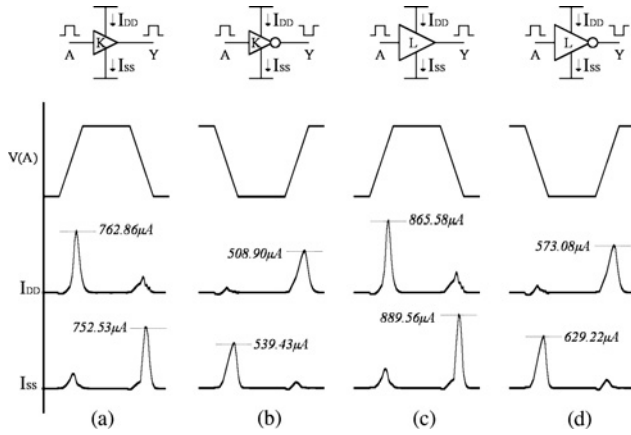


Fig. 2. Current profiles of four types of buffers and inverters in UMC 0.13 μm cell library by running SPICE simulation. $V(A)$ indicates the input waveform applied to each buffer or inverter and the load capacitance is set to 50 fF. (a) Buffer of strength level-K (BUFCKKHD). (b) Inverter of strength level-K (INVCKKHD). (c) Buffer of strength level-L (BUFCKLHD). (d) Inverter of strength level-L (INVCKLHD).

INVCKKHD and INVCKLHD in UMC 0.13 μm cell library produced by the SPICE simulation, in which I_{DD} and I_{SS} represent the amounts of current flows from the power line to the capacitance load and from the capacitance load to the ground line, respectively. Fig. 2(a) and (b) shows the current profiles of the buffer and inverter having the output driving strength level-K, respectively, and Fig. 2(c) and (d) shows the current profile of the buffer and inverter with the output driving strength level-L. (It is known that L denotes the higher strength level than K, implying that the cells with level-L are faster, but larger than the cells with level-K.) From the simulation, the two observations are identified.

A. Observation 1

The difference between the peak values of I_{DD} and I_{SS} of each type of buffers and inverters are non-trivial in that assuming the values are the same, as the previous works did, may lead to solutions of buffer/inverter allocation and polarity assignment that is far from the optimum in minimizing either power noise or ground noise. For example, see the peak value difference of INVCKLHD in Fig. 2(d), which is $629.22 - 573.08 = 54.14 \mu\text{A}$ ($= 9.4\%$ of the peak value of I_{SS}). Thus, when we minimize power/ground noise, the correct values of I_{DD} and I_{SS} should be used.

B. Observation 2

The same levels of output driving strengths of a buffer and an inverter do not mean the identical values of peak current. [See the values of peak currents in Fig. 2(a) and (b), and the values in Fig. 2(c) and (d). There are 18% and 16% differences of peak values between the cells of level-K and the cells of level-L, respectively.] Thus, assuming buffers and inverters expose the same current profiles if their strength levels are the same, as the previous works did, can cause a considerable inaccuracy in minimizing power/ground noise.

The two observations suggest that the buffer sizing and polarity assignment should use the correct values of the peak

TABLE I
NOTATIONS COMMONLY USED IN THE PROBLEM FORMULATION

e_i	A sink buffering element
L	Set of sink buffering elements
B	Set of buffers
I	Set of inverters
κ	Clock skew bound
$peak(x)$	Peak current on $x \in B \cup I$
p_i	A constant peak value

currents as well as the delays of the buffers and inverters, rather than simply treating them all identical only if their strength levels are the same.

III. PROBLEM FORMULATION

Since the power/ground noise is a local effect, we assume to have as input a set of sub-areas in a chip in which the peak current caused by the current flows through the sink buffering elements in each sub-area should be minimized. (The generation of sub-areas will be discussed in Section IV.) Then, solving the problem of minimizing power/ground noise on a chip corresponds to solving the problem of minimizing the peak current on each sub-area. Table I lists the notations which are commonly used in the formulation of the problem.

Problem 1 (PEAK-min): (Polarity assignment and buffer/inverter sizing for noise minimization) For a sub-area that contains a set L of sink buffering elements, a buffer type set B , an inverter type set I , and clock skew bound κ , find a mapping function $\phi : L \mapsto \{B \cup I\}$ that minimizes the quantity of

$$\max \left\{ \sum_{\phi(e_i) \in B} peak(\phi(e_i)), \sum_{\phi(e_i) \in I} peak(\phi(e_i)) \right\} \quad (1)$$

$$s.t. \quad t_{skew} \leq \kappa$$

where $t_{skew} = \max_{i=1, \dots, |L|} (arr_max(\phi(e_i))) - \min_{i=1, \dots, |L|} (arr_min(\phi(e_i)))$, in which $arr_max(\phi(e_i))$ and $arr_min(\phi(e_i))$ represent the latest arrival time and the earliest arrival time from the clock source to FFs that are connected directly to $\phi(e_i)$, respectively, and $peak(\phi(e_i))$ indicates the amount of peak current on $\phi(e_i)$.

In the following, we show that PEAK-min is NP-complete by reducing an NP-complete partitioning problem (PARTITION) to PEAK-min, and propose an optimal algorithm that solves the PEAK-min problem in pseudo-polynomial time.

Problem 2 (PARTITION): For a finite set A and a “size” $s(a) \in \mathbf{Z}^+$ for each $a \in A$, is there a subset $A' \subseteq A$ such that

$$\sum_{a \in A'} s(a) = \sum_{a \in A - A'} s(a).$$

Theorem 1: PARTITION is NP-complete [17].

Problem 3 (decision-PEAK-min): For a PEAK-min instance with (L, B, I, κ) and constant c , is there a mapping ϕ such that the value of expression in (1) is less than or equals c ?

Theorem 2: decision-PEAK-min is NP-complete.

Proof. It is easy to check that decision-PEAK-min is in NP. Let us consider a restricted version of decision-PEAK-min problem

$$B = \{b\} \text{ and } I = \{v\} \quad (2)$$

$$\text{peak}(\phi(e_i) = b) = \text{peak}(\phi(e_i) = v) = p_i \quad \forall i = 1, \dots, |L| \quad (3)$$

$$\kappa = \infty. \quad (4)$$

Then, the sum of the left and right terms in the expression of (1) is $\sum_{\phi(e_i) \in B} \text{peak}(\phi(e_i)) + \sum_{\phi(e_i) \in I} \text{peak}(\phi(e_i)) = \sum_{i=1, \dots, |L|} p_i$. Furthermore, it is true.

Property 1: The quantity of expression in (1) is greater than or equals $\frac{1}{2} \sum_{i=1, \dots, |L|} p_i$, and if the quantity equals $\frac{1}{2} \sum_{i=1, \dots, |L|} p_i$, $\sum_{\phi(e_i) \in B} \text{peak}(\phi(e_i)) = \sum_{\phi(e_i) \in I} \text{peak}(\phi(e_i))$.

Now, we reduce an instance [with A and $s(\cdot)$] of PARTITION problem to an instance (with L, B, I, κ , and c) of the restricted problem of decision-PEAK-min in the following.

- 1) Let $L = A$, so that e_i in L is a_i in A , for $i = 1, \dots, |A|$.
- 2) Let $p_i = s(a_i)$, for $i = 1, \dots, |A|$.
- 3) Let $c = \frac{1}{2} \sum_{i=1, \dots, |L|} p_i$.
- 4) B, I, κ are those that satisfy (2)–(4).

Suppose we have found a mapping ϕ that solves the instance with L, B, I, κ , and c . From the solution, we obtain a solution of the instance of PARTITION problem by setting $A' = \{a_i | \phi(e_i) \in B\}$ and $A - A' = \{a_i | \phi(e_i) \in I\}$. If $\phi(\cdot)$ satisfies the inequality that the value of expression in (1) $\leq c (= \frac{1}{2} \sum_{i=1, \dots, |L|} p_i)$, then, by property 1, $\sum_{\phi(e_i) \in B} \text{peak}(\phi(e_i)) = \sum_{\phi(e_i) \in I} \text{peak}(\phi(e_i))$, which means $\sum_{a \in A'} s(a) = \sum_{a \in A - A'} s(a)$. ■

Our approach to solving an instance of PEAK-min first divides the problem into many subproblems (discussed in Section IV-A). Then, each subproblem is tackled optimally (discussed in Section III), from which a globally optimal solution is derived (discussed in Section IV-B). The generation of subproblems is based on constraining the arrival times of clock signals, thus restricting the types of buffers and inverters to be allocated to the sinks.

More specifically, let us consider a signal arrival time interval $H(t) = [t - \kappa, t]$. Let $C(e_i)$ denote the set of buffers and inverters such that the values of $\text{arr_max}(\cdot)$ and $\text{arr_min}(\cdot)$ for their assignments to sink $e_i \in L$ are in $H(t)$. If $C(e_i)$ contains more than one buffer type, we remove all the buffer types from $C(e_i)$ except the one with the lowest peak current. Similarly, if $C(e_i)$ has more than one inverter type, we remove all the inverter types except the one with the lowest peak current.² (Recall that we want to minimize the total peak current by allocating buffers and inverters to sinks.) Then, we classify the sinks into four groups according to $C(e_i)$ s.

- 1) $G1 = \{e_i \in L | |C(e_i)| = 2\}$.
- 2) $G2 = \{e_i \in L | |C(e_i)| = 1, C(e_i) \text{ has a buffer}\}$.
- 3) $G3 = \{e_i \in L | |C(e_i)| = 1, C(e_i) \text{ has an inverter}\}$.
- 4) $G4 = \{e_i \in L | |C(e_i)| = 0\}$.

²We often simply say “buffer” for buffer type and “inverter” for inverter type if it incurs no confusion.

Definition 1 (Feasible Time Interval): A signal arrival time interval is feasible if each sink has at least one buffer or inverter type to be assigned such that the resulting earliest and latest arrival times are in the arrival time interval, i.e., $G4$ is empty.

Note that since the numbers of buffers and inverters available in B and I are bounded by constants, the number of feasible time intervals are bounded by $|L| \cdot (|B| + |I|)$, where the worst case happens when all the arrival times of buffer/inverter mappings to sinks are distinct.

Once the feasible time intervals are produced, as a next step, we focus on solving the PEAK-min problem for each of the feasible arrival time intervals, which we call PEAK-min-interval problem. Precisely, PEAK-min-interval can be formulated as follows.

Since the sink buffering element $e_i \in G2$ should be assigned to a buffer type, and $e_i \in G3$ should be assigned to an inverter type, we define constants P_f^+ and P_f^- as follows:

$$P_f^+ = \sum_{e_i \in G2} \text{peak}(\phi(e_i)), \quad P_f^- = \sum_{e_i \in G3} \text{peak}(\phi(e_i)).$$

We introduce constant notations p_i^+ and p_i^- , and 0-1 integer variables x_i and y_i , for each sink $e_i \in G1$

$$\begin{aligned} p_i^+ &= \text{peak}(\phi(e_i)) \quad \text{for } \phi(e_i) \in C(e_i) \cap B \\ p_i^- &= \text{peak}(\phi(e_i)) \quad \text{for } \phi(e_i) \in C(e_i) \cap I \\ x_i &= \begin{cases} 1 & \text{if } \phi(e_i) \in B \\ 0 & \text{otherwise} \end{cases} \quad y_i = \begin{cases} 1 & \text{if } \phi(e_i) \in I \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where $x_i + y_i = 1$. Then, the (minimization) cost function in expression in (1) can be reformulated as minimizing

$$\max \left\{ \sum_{e_i \in G1} p_i^+ x_i + P_f^+, \sum_{e_i \in G1} p_i^- y_i + P_f^- \right\}. \quad (5)$$

We formulate the problem of determining the values of variables x_i and y_i that minimizes the quantity of expression in (5) into the KNAPSACK problem [18]. The KNAPSACK problem is stated as follows.

Problem 4 (KNAPSACK): Given a set of n items with a “gain” $g_i \in \mathbf{Z}^+$ and a “value” $v_i \in \mathbf{Z}^+$ for item i , and a capacity constraint $W \in \mathbf{Z}^+$, select a subset of the items so as to

$$\begin{aligned} &\text{maximize} && \sum_{i=1}^n g_i z_i \\ &\text{subject to} && \sum_{i=1}^n v_i z_i \leq W \\ &\text{where} && z_i = \begin{cases} 1 & \text{if item } i \text{ is selected} \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

we divide expression in (5) into two cases

$$\text{case1: } \sum_{e_i \in G1} p_i^+ x_i + P_f^+ \geq \sum_{e_i \in G1} p_i^- y_i + P_f^-$$

$$\text{case2: } \sum_{e_i \in G1} p_i^+ x_i + P_f^+ < \sum_{e_i \in G1} p_i^- y_i + P_f^-$$

For *case 1*, by replacing x_i with $1 - y_i$, the PEAK-min-interval problem becomes

$$\begin{aligned} & \text{maximize} && \sum_{e_i \in G1} p_i^- y_i \\ & \text{subject to} && \sum_{e_i \in G1} (p_i^+ + p_i^-) y_i \leq \sum_{e_i \in G1} p_i^+ + P_f^+ - P_f^- \\ & && y_i = 0 \text{ or } 1 \end{aligned}$$

Then, the *case 1* problem can be reduced to the KNAPSACK in problem 4 by setting y_i to z_i , p_i^- to g_i , $p_i^+ + p_i^-$ to v_i , $\sum_{e_i \in G1} p_i^+ + P_f^+ - P_f^-$ ³ to W , and $|G1|$ to n . The *case 2* of the PEAK-min-interval problem is also similarly transformed to the KNAPSACK problem. Consequently, an instance of the PEAK-min-interval problem can be solved by solving two instances of the KNAPSACK problem and selecting the solution with the smaller cost value.

We solve each instance of the KNAPSACK problem by formulating it into a dynamic programming (DP) [18]. The subproblems to be solved are, for $j = 1, \dots, n$, the forms of $K(w, j)$, which designates the maximum value achievable using a knapsack of capacity w and items $1, \dots, j$. Then, the answer we look for is $K(W, n)$. We can express subproblem $K(w, j)$ with smaller subproblems

$$K(w, j) = \max\{K(w - w_j, j - 1) + v_j, K(w, j - 1)\}. \quad (6)$$

As a result, our dynamic programming algorithm, called PEAK-min-DP, consists of filling a 2-D table of $W + 1$ rows and $n + 1$ columns. Each entry takes constant time, and overall time takes $O(nW)$. The initializations are $K(0, j) = 0$, $j = 0, 1, 2, \dots, n$ and $K(w, 0) = 0$, for $w = 0, 1, 2, \dots, W$.

Theorem 3: PEAK-min-interval problem can be solved in pseudo-polynomial time.

IV. ALGORITHM FOR NOISE MINIMIZATION

The proposed algorithm called CLK-NOISE accepts, as input, a synthesized clock routing tree that contains buffering elements. The objective of CLK-NOISE is to minimize the power/ground noise on the clock tree by considering buffer sizing and polarity assignment to the sink buffering elements while satisfying the clock skew constraint. CLK-NOISE is performed in two phases. In the first phase, a minimal set of feasible signal arrival time intervals is extracted. In the second phase, CLK-NOISE searches a solution of buffer sizing and polarity assignment with the lowest power/ground noise by exploiting the PEAK-min-DP algorithm repeatedly on the feasible time intervals obtained in the first phase. Fig. 3 shows the overall flow of CLK-NOISE. The following two subsections cover the detail description on the steps of the flow.

A. Phase 1: Generation of a Minimal Set of Feasible Time Intervals

Since selecting a buffer or an inverter from the libraries and assigning it to a sink determines the minimal and maximal

³All p_i^+ , p_i^- , P_f^+ , and P_f^- values are scaled to be positive integer numbers.

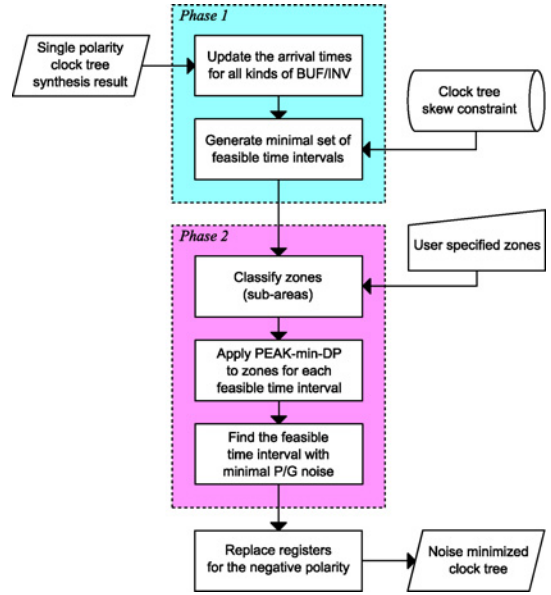


Fig. 3. Flow of CLK-NOISE.

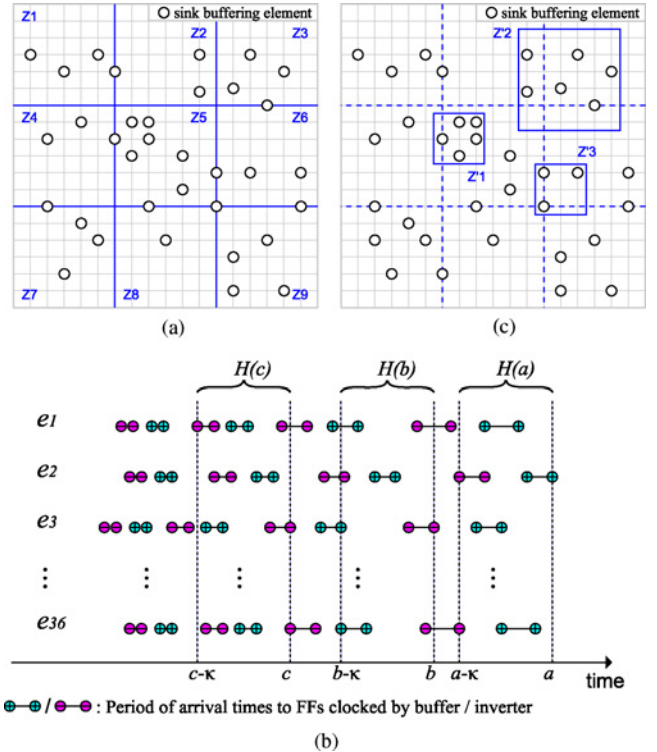


Fig. 4. Design example for explaining the proposed algorithms. (a) Disposition of sink buffering elements and uniformly distributed zones (i.e., sub-areas). (b) Arrival time intervals for sinks with various allocations of buffers and inverters. (c) User specified zones.

arrival times from the clock source to the FFs connected to the sink, we are interested in finding all the *feasible* arrival time intervals of length κ . In other words, we remove from our consideration the time intervals for which there exists at least one sink that is able to be assigned to none of buffers and inverters in the libraries. For example, consider the sink buffering element e_1, \dots, e_{36} that have been disposed

in Fig. 4(a). When we assume that the buffer library B and inverter library I have four types of buffers and inverters, respectively, Fig. 4(b) illustrates the arrival times for all possible assignments of buffers and inverters to sinks where the left and right end-points of each line segment in a sink indicate the minimum and maximum arrival times of clock signal to FFs through the corresponding buffer or inverter assigned to the sink. The assigned buffers and inverters on the left side in the arrival times of Fig. 4(b) are relatively faster, but larger in size and consume more current than those on the right. We search through, from right to left, feasible time intervals [e.g., starting from the interval $H(a)$ in Fig. 4(b)]. The number of distinct time intervals is bounded by the sum of the numbers of line segments because setting the time corresponding to the right end-point, say t_i , of each segment uniquely determines one arrival time interval $H(t_i) = [t_i - \kappa, t_i]$,⁴ and any of other time intervals can be reduced to one of $H(t_i)$'s. Thus, the number of time intervals to check is $|L| \cdot (|B| + |I|)$. (Note that there exists at least one feasible time interval when the skew constraint is ignored or when $\kappa = \infty$.) For each time interval $H(t_i)$, we check if every sink has at least one line segment that is contained in $H(t_i)$. That is, we check if $G4$ is empty or not. If $G4$ is empty, we include $H(t_i)$ in the set of feasible arrival time intervals. For example, $H(a)$ and $H(c)$ are feasible arrival time intervals, but $H(b)$ is not because there is no possible assignment for sink e_1 . The following theorem claims that the search process of feasible time intervals can be greatly simplified by cutting-off the feasible time intervals that will cause higher power/ground noises than some of the others.

Definition 2 (Completely Feasible): An arrival time interval is called completely feasible if it is feasible and its $G1$ contains all sinks, i.e., $|G1| = |L|$.

For example, the arrival time interval $H(c)$ in Fig. 4(b) is completely feasible, but $H(a)$ is not completely feasible.

Theorem 4: If $H(t_i)$ is completely feasible, for each t_j such that $t_j < t_i$ the peak current on $H(t_j)$ is always greater than or equals the minimum peak current on $H(t_i)$.

Proof. Let $\phi_i(\cdot)$ and $\phi_j(\cdot)$ be minimal peak current mapping functions on the completely feasible time interval $H(t_i)$ and a feasible time interval $H(t_j)$, respectively. Let us now generate another mapping $\phi_k(\cdot)$ on $H(t_i)$. Clearly, (statement 1) the peak current by $\phi_k(\cdot)$ is greater than or equals that by $\phi_i(\cdot)$. By noting that $H(t_i)$ is completely feasible, $\phi_k(\cdot)$ is obtained as follows: for each sink buffering element e , if e is assigned to a buffer by $\phi_j(e)$, $\phi_k(e)$ is set to the buffer in $H(t_i)$. Likewise, if e is assigned to an inverter by $\phi_j(e)$, $\phi_k(e)$ is set to the inverter in $H(t_i)$.

Since we have assumed that a faster buffer (or inverter) causes a higher peak current than the slower buffer (or inverter), $peak(\phi_j(e)) \geq peak(\phi_k(e))$ for each sink e , which means that (statement 2) the peak current by $\phi_j(\cdot)$ is greater than or equals that by $\phi_k(\cdot)$. Then, by combining statements 1 and 2, the peak current by $\phi_j(\cdot)$ is greater than or equals that by $\phi_i(\cdot)$. ■

⁴Note that the time interval needs to be a bit tighter in length than $[t_i - \kappa, t_i]$ to consider the sibling buffer sizing effect on slew. We have observed from experiments that the difference between the minimum and maximum slews is around 3.7%.

CLK-NOISE: Phase 1

Inputs: (L, B, I, κ)

Output: a set, \mathcal{H} , of feasible time intervals

```

•  $\mathcal{H} = \emptyset$ ;
• sort  $arr\_max(\phi(e_i) = \varepsilon)$ ,  $\forall e_i \in L$ ,  $\forall \varepsilon \in B \cup I$ 
  in decreasing order;
for (each  $arr\_max(\phi(e) = \varepsilon)$  in the sorted list) {
  •  $D_{max} = arr\_max(\phi(e) = \varepsilon)$ ;
  •  $D_{min} = D_{max} - \kappa$ ;
  •  $isValid = \text{true}$ ;
  •  $count = 0$ ;
  for (each  $e_i \in L$ ) {
  •  $C(e_i) = \{\varepsilon \mid D_{min} \leq arr\_min(\phi(e_i) = \varepsilon)$ 
    &&  $arr\_max(\phi(e_i) = \varepsilon) \leq D_{max}, \varepsilon \in B \cup I\}$ ;
  if ( $C(e_i) = \emptyset$ ) {
  •  $isValid = \text{false}$ ; break; /* exit inner loop */
  }
  else if ( $C(e_i) \cap B \neq \emptyset$  &&  $C(e_i) \cap I \neq \emptyset$ )
  •  $count++$ ;
  }
  if ( $isValid = \text{true}$ ) • include  $H(D_{max})$  in  $\mathcal{H}$ ;
  /* check if completely feasible */
  if ( $count = |L|$ ) break;
  }
return  $\mathcal{H}$ ;
        
```

Fig. 5. Summary of phase 1 in CLK-NOISE: generating a minimal set of feasible time intervals.

zones	Feasible time intervals in \mathcal{H} ($ \mathcal{H} = n$)					current constr.	
	$H(t_1)$	$H(t_2)$	$H(t_3)$...	$H(t_n)$		
Z'	z'_1	26	19	24	...	16	≤ 20
	z'_2	39	29	34	...	27	≤ 31
	z'_3	17	13	15	...	12	≤ 15
Z	z_1	-	12	-	...	10	
	z_2	-	10	-	...	13	
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
	z_5	-	42	-	...	37	
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
$\mathcal{P}H(\cdot)^{max}$	-	42	-	...	37		

Fig. 6. Illustration of selecting a feasible interval with minimum noise in phase 2 for the example in Fig. 4(c).

According to theorem 4, we can terminate the search process when a completely feasible time interval is encountered. Fig. 5 summarizes the code of *phase1* in CLK-NOISE. Note that if the output \mathcal{H} of *phase1* is empty (i.e., there is no feasible time interval), it means the clock skew constraint is too tight. Thus, it may be needed to relax the skew constraint by increasing the value of κ and repeat the phase.

B. Phase 2: Finding a Feasible Interval of Minimum Noise

Since the power/ground noise is a local effect, it is assumed that designer is given a set of circuit sub-areas, which we call *zones*, for which their peak currents should be minimized. We consider two strategies of zone generation: 1) *generating zones by uniform partitioning*, which partitions the circuit into small pieces of equal size with no overlaps, and 2) *specifying zones by designer*, in which according to the designer's experience

and knowledge on designing the circuit, she or he locates zones where the peak currents should be minimized. For example, Fig. 4(a) shows a uniform partition of circuit with nine zones. On the other hand, Fig. 4(c) shows three designer-specified zones. [Note that some zones may be overlapped. In that case, priorities among the zones for minimizing peak currents should be given. In addition, a peak current constraint (i.e., current bound) on each specified zone is given.] First, we consider the simple strategy of uniform partition. Handling the strategy of user-specified zones will be described subsequently.

C. Handling Zones Generated by Uniform Partitioning

The objective of CLK-NOISE is to find a feasible interval that leads to a minimum among the maximum values of peak currents of zones of feasible intervals. For a feasible interval $H(t)$, the peak current values of zones can be obtained by applying PEAK-min-DP to each zone, from which the maximum peak current value, $p_{H(t)}^{max}$, can be determined. Then, CLK-NOISE chooses the feasible interval with the smallest value of $p_{H(t)}^{max}$. Note that the selection of feasible interval and the application of PEAK-min-DP solves the combined problem of buffer sizing and polarity assignment accordingly.

D. Integrating Designer-Specified Zones

For each feasible interval, PEAK-min-DP is applied to the designer-specified zones first. Then, CLK-NOISE selects the feasible intervals for which the peak currents of the designer-specified zones all satisfy their peak current constraints. For example, the three rows labeled z'_1 , z'_2 , and z'_3 in Fig. 6 illustrate the peak current values for the zones z'_1 , z'_2 , and z'_3 in Fig. 4(c) for the feasible intervals $H(t_1), \dots, H(t_n)$ obtained in phase 1. The values in the last column show the peak current constraints of the zones. The values with underscore indicate that the corresponding zones satisfy the peak current constraints. For example, feasible intervals $H(t_1)$ and $H(t_3)$ violate the peak current constraints, but $H(t_2)$ and $H(t_n)$ can be considered as candidates for further noise minimization. If there are additional zones to reduce noise, the process is repeated for the set of selected feasible intervals while preserving the results of buffer sizing and polarity assignment done in the previous iteration(s). For example, the rows labeled z_1, \dots, z_9 show the peak current values for the uniform partitioned zones for the selected feasible intervals. If the zones are from uniform partition of circuit, CLK-NOISE computes $p_{H(t)}^{max}$ values for the feasible intervals, and chooses the feasible interval with minimum value of $p_{H(t)}^{max}$. For the example in Fig. 6, $H(t_n)$ is selected.

Fig. 7 summarizes the code of *phase2* in CLK-NOISE for a set of mixed zones specified by designer and by uniform partition. In phase 2, the number of calls to PEAK-min-DP equals the number of zones times the number of feasible intervals in \mathcal{H} . The PEAK-min-DP algorithm with a zone z and a time interval $H(t)$ takes a run time proportional to the value of the sum of the peak currents of the buffers and inverters that can be assigned to the buffering elements in z for $H(t)$.

CLK-NOISE: Phase 2
<i>Inputs:</i> ($\mathcal{H}, Z', \Delta, Z$) /* \mathcal{H} : feasible intervals, */ /* Δ : zones' peak current constraints, */ /* Z' : user specified zones, */ /* Z : uniformly partitioned zones */
<i>Output:</i> a mapping function ϕ
for (each interval $H(t) \in \mathcal{H}$) { • <i>isValid</i> = true ; for (each zone $z' \in Z'$) { apply PEAK-min-DP to buffer elements in z' ; if ((peak current in z') > $\delta_{z'}$) { /* current constr. */ • <i>isValid</i> = false ; } } if (<i>isValid</i> = false) continue ; /* go on next $H(t)$ */ for (each $z \in Z$) apply PEAK-min-DP to z ; • $p_{H(t)}^{max}$ = maximum peak current in $z \in Z$; } } return $\phi(\cdot)$ of $H(t)$ with the least value of $p_{H(\cdot)}^{max}$;

Fig. 7. Summary of phase 2 in CLK-NOISE: finding a feasible interval of minimum noise.

E. Considering the Effect of Thermal Variation

The nonuniform temperatures on a chip as well as the significant on-chip thermal gradient which occur during the execution of chip circuits of high power density are the main cause of high delay variations [19]. Since the clock nets are one of the most sensitive signals to the delay variations caused by the thermal variation [20], [21], it is important to consider the effect of thermal variation on the polarity assignment in the clock tree synthesis. There are a couple of works which have considered the clock tree synthesis under the thermal variation. TACO [22] constructed a tree that balances the clock skew under the two given static thermal profiles, one uniform and the other worst. The reason of choosing only the two thermal profiles is that analyzing and optimizing all the transient thermal profiles between the two profiles is an extremely difficult task. BURITO [23] then extended the TACO's work to the clock tree synthesis in the 3-D IC designs. The major difference between our paper, named CLK-NOISE-t, and the works in TACO and BURITO is that the task of TACO and BURITO is to restructure the initial clock tree routing with the objective of minimizing the additional clock wirelength while balancing and minimizing the clock skew of the worst thermal profile, whereas the task of CLK-NOISE-t is to determine the buffer sizing and polarity assignment with the objective of minimizing the power/ground noise while satisfying the clock skew constraint under the thermal variation. That is, CLK-NOISE-t preserves the routing of the initial clock tree.

Let us suppose that we are given M chip thermal profiles P_1, P_2, \dots, P_M which are extracted during the execution of the circuits in the chip, where we assume P_1 is the uniform (lowest) temperature profile of the circuit just before the execution. Then, the thermal profiles may produce different clock skews on the same clock tree, causing clock skew variation. This means that our thermal aware polarity assignment and

buffer sizing requires to satisfy the clock skew constraint under every thermal profile. (Note that the value of peak current, for the same buffer sizing and polarity assignment, may go down as the chip temperature goes up due to the increase of the delay.) However, since the places in which the buffer sizing and polarity assignment are considered are confined to the relatively small and short-distance regions that contain the sink buffering elements [see Fig. 1(c)] we assume that the peak current for a solution of polarity assignment and buffer sizing is invariant with respect to the temperature.⁵ The problem we want to solve for a sub-area on a chip can be stated as follows.

Problem 5 (Thermal aware polarity assignment and buffer/inverter sizing for noise minimization): For a sub-area that contains a set L of sink buffering elements, a buffer type set B , an inverter type set I , thermal profiles P_1, P_2, \dots , and P_M , and clock skew bound κ , find a mapping function $\phi : L \mapsto \{B \cup I\}$ that minimizes the quantity of

$$\max \left\{ \sum_{\phi(e_i) \in B} \text{peak}(\phi(e_i)), \sum_{\phi(e_i) \in I} \text{peak}(\phi(e_i)) \right\} \quad (7)$$

s.t. $t_{skew,j} \leq \kappa, \forall j = 1, \dots, M$

where $t_{skew,j} = \max_{i=1, \dots, |L|} (\text{arr_max}_j(\phi(e_i))) - \min_{i=1, \dots, |L|} (\text{arr_min}_j(\phi(e_i)))$, in which $\text{arr_max}_j(\phi(e_i))$ and $\text{arr_min}_j(\phi(e_i))$ represent the latest arrival time and the earliest arrival time from the clock source to FFs that are connected directly to $\phi(e_i)$ under thermal profile P_j , respectively, and $\text{peak}(\phi(e_i))$ indicates the amount of peak current on $\phi(e_i)$ under P_1 .

We solve the problem of satisfying all the clock skew constraints under all thermal profiles P_1, P_2, \dots , and P_M by manipulating feasible time intervals as follows. For each P_j , we apply phase 1 of CLK-NOISE to generate all the feasible time intervals. Let us denote the set of feasible time intervals corresponding to P_j by $\mathcal{H}_j = \{H(t_{(j,1)}), H(t_{(j,2)}), \dots\}$. In addition, let $C_{(j,k)}(e_i)$ denote the set of buffers and inverters such that the values of $\text{arr_max}(\cdot)$ and $\text{arr_min}(\cdot)$ for their assignments to sink $e_i \in L$ are in $H(t_{(j,k)})$ in set \mathcal{H}_j . Then, by the definition of feasible time intervals, $C_{(j,\cdot)}(e_i) \neq \emptyset$, for each sink $e_i \in L$ and time interval $H(t_{(j,\cdot)}) \in \mathcal{H}_j$. Each feasible time interval is characterized by its $C_{(j,\cdot)}(e_i)$'s.

Definition 3 (Intersection of Feasible Time Interval Sets): The intersection, denoted as $\mathcal{H}_{(j,l)}$, of two feasible interval sets \mathcal{H}_j and \mathcal{H}_l is defined as the set of the intersection, denoted as $H(t_{(j,l,\cdot)})$, of every pair of their elements $H(t_{(j,\cdot)}) \in \mathcal{H}_j$ and $H(t_{(l,\cdot)}) \in \mathcal{H}_l$. (The intersection of two feasible time intervals $H(t_{(j,\cdot)})$ and $H(t_{(l,\cdot)})$ is characterized by the set intersection $C_{(j,\cdot)}(e_i) \cap C_{(l,\cdot)}(e_i)$ for every sink $e_i \in L$), and satisfying that $H(t_{(j,l,\cdot)})$ is a feasible time interval. ($H(t_{(j,l,\cdot)})$ is called

⁵The total current flow which occurs when a buffer or an inverter is switching is sensitive to the load capacitance but insensitive to the temperature. However, the peak value of the current will be somewhat lowered as the temperature increases because of the increase of switching delay. In our paper, the peak current value under the initial thermal profile P_1 is used as the representative value of peak currents over all thermal profiles, which can in fact be used as an upper bound of the peak currents under P_2, \dots, P_M because the temperature on P_1 is the lowest.

	Profile P_1		Profile P_2		Profile P_3
	$H_1(43)$	$H_1(44)$	$H_2(59)$	$H_2(53)$	$H_3(74.3)$
e_1	b_1 i_1	b_1	i_1	i_1	b_1, b_2 i_1
e_2	b_1, b_2, b_3 i_1, i_2	b_2, b_3, b_4 i_2, i_3	b_2, b_3 i_1, i_2, i_3	b_3, b_4 i_3	b_2, b_3 i_1
e_3	b_2, b_3 i_1, i_2, i_3	b_3, b_4 i_2, i_3	b_3, b_4 i_2, i_3	b_3, b_4 i_3	b_3 i_2
e_4	b_1, b_2, b_3 i_2, i_3, i_4	b_2, b_3 i_3, i_4	b_1, b_2 i_3, i_4	b_2 i_3	b_1
e_5	b_1, b_2 i_1	b_1, b_2 i_1	b_1, b_2	b_1, b_2 i_1	b_1, b_2 i_4

(a)

$\mathcal{H}_{(1,2)}$	$H_{(1,2)}(43,59)$	$H_{(\cdot)}(43,53)$	$H_{(\cdot)}(44,59)$	$H_{(\cdot)}(44,53)$
e_1	i_1	i_1	\emptyset	\emptyset
e_2	b_2, b_3 i_1, i_2	b_3	b_2, b_3 i_2, i_3	b_3, b_4 i_3
e_3	b_3 i_2, i_3	b_3 i_3	b_3, b_4 i_2, i_3	b_3, b_4 i_3
e_4	b_1, b_2 i_3, i_4	b_2 i_3	b_2 i_3, i_4	b_2 i_3
e_5	b_1, b_2	b_1, b_2 i_1	b_1, b_2	b_1, b_2 i_1
	feasible	feasible		

(b)

$\mathcal{H}_{(1,2,3)}$	$H_{(1,2,3)}(43,59,74.3)$	$H_{(1,2,3)}(43,53,74.3)$
e_1	i_1	i_1
e_2	b_2, b_3 i_1	b_3
e_3	b_3 i_2	b_3
e_4	b_1	\emptyset
e_5	b_1, b_2	b_1, b_2
	feasible	

(c)

Fig. 8. Example illustrating the derivation of feasible time intervals under multiple thermal profiles. (a) Example of the sets, $\mathcal{H}_1, \mathcal{H}_2$, and \mathcal{H}_3 , of feasible time intervals for the clock trees under P_1, P_2 and P_3 . (b) Set, $\mathcal{H}_{(1,2)}$, of time intervals produced by intersecting the feasible time intervals in \mathcal{H}_1 with those in \mathcal{H}_2 . (c) Set, $\mathcal{H}_{(1,2,3)}$, of time intervals produced by the intersection of the feasible time intervals in $\mathcal{H}_{(1,2)}$ with that in \mathcal{H}_3 .

a feasible time interval for $H(t_{(j,\cdot)})$ and $H(t_{(l,\cdot)})$ of thermal profiles P_j and P_l if $C_{(j,\cdot)}(e_i) \cap C_{(l,\cdot)}(e_i) \neq \emptyset$ for every $e_i \in L$.)

CLK-NOISE-t will compute the intersection of all feasible time interval sets of P_1, P_2, \dots, P_M incrementally: $\mathcal{H}_{(1,2)}$ is obtained from \mathcal{H}_1 and \mathcal{H}_2 . $\mathcal{H}_{(1,2)}$ is then intersected with \mathcal{H}_3 to produce $\mathcal{H}_{(1,2,3)}$. This process is repeated until the intersection produces an empty set or $\mathcal{H}_{(1,2,3,\dots,M)}$ is produced. The generation of empty set in the process of intersection means that there is no feasible time interval which satisfies

CLK-NOISE-t: Thermal aware polarity assignment and buffer sizing
<i>Inputs:</i> $(L, B, I, \kappa, P_1, \dots, P_M)$ <i>/*</i> P_1, \dots, P_M : thermal profile <i>*/</i> <i>Output:</i> a mapping function ϕ
apply phase 1 of CLK-NOISE to each P_1, \dots, P_M to produce feasible interval sets $\mathcal{H}_i, i = 1, \dots, M$; produce $\mathcal{H}_{(1,2)}$ by $\mathcal{H}_1 \cap \mathcal{H}_2$; for (each $\mathcal{H}_i, i = 3, \dots, M$) { produce $\mathcal{H}_{(1,2,\dots,i)}$ by $\mathcal{H}_{(1,2,\dots,i-1)} \cap \mathcal{H}_i$; if $(\mathcal{H}_{(1,2,\dots,i)} = \emptyset)$ return "no solution"; }
apply phase 2 of CLK-NOISE to $\mathcal{H}_{(1,2,\dots,M)}$; return $\phi(\cdot)$ of $H(t_{(\cdot)}) \in \mathcal{H}_{(1,2,\dots,M)}$ with minimum $p_{H(\cdot)}^{max}$.

Fig. 9. Procedure of CLK-NOISE-t: considering the effect of thermal variation.

the clock skew constraint under all thermal profiles P_1, P_2, \dots, P_M . In that case, it may be needed to relax the clock skew constraint by increasing the value of κ and repeat the intersection operation. The next step is then to apply phase 2 of CLK-NOISE with the feasible time intervals in $\mathcal{H}_{(1,2,3,\dots,M)}$.

The example in Fig. 8 illustrates the intersection of feasible time intervals. Suppose we have extracted three thermal profiles P_1, P_2 , and P_3 where it is assumed that they are the thermal instances at the beginning, in the middle, and at the end of the execution of chip circuit, respectively. Further, suppose that there are five sinks e_1, \dots, e_5 , three types of buffer b_1, b_2, b_3 , and three types of inverter i_1, i_2, i_3 . Fig. 8(a) shows an example of the sets $\mathcal{H}_1, \mathcal{H}_2$, and \mathcal{H}_3 , of feasible time intervals produced by the application of the first phase of CLK-NOISE for the clock trees of P_1, P_2 , and P_3 under the same clock skew bound.⁶ Fig. 8(b) then shows the result of $\mathcal{H}_1 \cap \mathcal{H}_2$, which is $\mathcal{H}_{(1,2)}$, where two time intervals are feasible. Then, by intersecting each of the two feasible time intervals with that in \mathcal{H}_3 , we produce the two time intervals as shown in Fig. 8(c), in which the first one is feasible. Finally, the second phase of CLK-NOISE will be applied to the feasible interval.

Fig. 9 summarizes the procedure of CLK-NOISE-t which consists of three steps: (Step 1) applying the first phase of CLK-NOISE to compute all the feasible intervals of thermal profiles; (Step 2) iteratively intersecting the feasible time intervals to produce a set of feasible time intervals that satisfy the clock skew constraint under all thermal profiles; and (Step 3) applying the second phase of CLK-NOISE to find a solution of polarity assignment and buffer sizing with least peak current among the feasible time intervals obtained in Step 2. Since each \mathcal{H}_j contains at most $|L| \cdot (|B| + |I|)$ number of feasible time intervals and the intersection of two feasible time intervals can be computed by $O(|L| \cdot (|B| + |I|))$ with $O(|B| + |I|)$ time for set operation of each $e_i \in L$. The computation time of $\mathcal{H}_{(j,l)}$ is bounded by $O(|L|^3 \cdot (|B| + |I|)^3)$. Thus, the total computation time to $\mathcal{H}_{(1,2,\dots,M)}$ is bounded by $O(|L|^{(M+1)} \cdot (|B| + |I|)^{(M+1)})$.

⁶We can see that as the thermal profile changes by execution of circuit the number of candidate buffers and inverters on the feasible time intervals is reduced. This is because of the increase of clock delay variation.

V. EXPERIMENTAL RESULTS

The proposed algorithm CLK-NOISE for solving the combined problem of polarity assignment and buffer sizing with the objective of minimizing power/ground noise has been implemented in C on a Linux machine and tested on ISCAS89 and two of ISPD09 benchmark circuits. We obtained the locations of the FFs of the circuits in [24], by performing synthesis using Berkeley SIS and placement using UCLA Dragon. The clock trees were then generated by using the algorithm in [25]. We combined the cluster based algorithm in [26] with the clock tree generation to produce sink buffering elements. We used four pairs of buffer and inverter types taken from the UMC (0.13 μm) standard cell library. The pairs have different driving strength levels G, H, I, and J. (Level-J indicates the fastest delay and the largest current consuming level whereas level-G indicates the slowest delay and the least current consuming level.) The model parameters of the buffers and inverters were taken from [27]. For the SPICE simulation to measure the peak current, we used the power grid model in [16].

We compared the results produced by CLK-NOISE with that by the approach, we name *Polarity-only*, proposed by Chen, Ho, and Hwang [9], which heuristically solves the polarity assignment problem only, under the assumption that the peak currents of a buffer and an inverter are the same. The experimentations were performed in four-fold to assess the effectiveness of CLK-NOISE: 1) how much effectively CLK-NOISE solves the problem of polarity assignment compared to that of *Polarity-only* approach; 2) how much effectively CLK-NOISE solves the combined problem of polarity assignment and buffer sizing on reducing power/ground noise as well as total peak current; 3) how much effectively CLK-NOISE explores the design space by varying the clock skew budget; and 4) how much effectively CLK-NOISE-t takes into account the thermal variations.

- 1) Assessing the effectiveness of CLK-NOISE for polarity assignment on reducing P/G noise. Table II summarizes the simulation results of the designs produced by *Polarity-only* which uses level-I BUF/INV and the designs by CLK-NOISE which also uses level-I BUF/INV. Since *Polarity-only* produces minimum skews of 24.6 ps–29.8 ps when the level-I BUF/INV are used, we set the skew budget to 30 ps in CLK-NOISE for a fair comparison. The column $|L|$ represents the number of sink buffering elements on each circuit. The last three columns in each column section labeled *Polarity-only* and CLK-NOISE represent the values of total peak current, maximum power noise, and maximum ground noise of the designs produced by *Polarity-only* and CLK-NOISE, respectively. Column $|Z|$ indicates the number of zones that are used in CLK-NOISE.⁷ The last three columns labeled *Improvement* show the improvements by CLK-NOISE over *Polarity-only*. We can see that the average improvements of the maximum power and ground noises by CLK-NOISE are 11.9% and 12.8%, respectively. Fig. 10(a) and (b) shows the power

⁷The zone size is set approximately to $500 \mu\text{m} \times 500 \mu\text{m}$ for each circuit.

TABLE II

COMPARISON OF RESULTS (Polarity-Only [9]: USING **Level-I** BUF/INV, CLK-NOISE: USING **Level-I** BUF/INV, SKEW_BUDGET ≤ 30 PS)

Benchmark Info		Polarity-Only [9] (Using Level-I)					CLK-NOISE (Using Level-I)						Improvement		
Circuit	L	Skew (ps)	Run Time (s)	Peak Curr. (mA)	Power Noise (mV)	Ground Noise (mV)	Skew (ps) (≤30)	Z	Run Time (s)	Peak Curr. (mA)	Power Noise (mV)	Ground Noise (mV)	Peak Curr. (%)	Power Noise (%)	Ground Noise (%)
s5378	40	28.9	<0.01	15.2	10.7	11.3	28.9	12	<0.01	13.2	9.6	9.9	13.2	10.3	12.4
s9234	50	28.9	<0.01	19.9	16.5	16.8	29.3	16	<0.01	18.5	13.6	14.0	7.0	17.6	16.7
s13207	134	29.4	0.01	50.7	41.6	42.1	29.5	36	0.03	46.4	36.6	37.9	8.5	12.0	10.0
s15850	133	26.9	0.01	49.4	41.5	43.2	28.4	49	0.04	42.3	36.9	36.7	14.4	11.1	15.0
s35932	407	29.6	0.26	114.1	114.0	101.0	29.8	121	0.27	97.0	90.0	92.8	15.0	21.1	8.1
s38417	343	29.8	0.16	101.7	94.1	103.0	29.9	81	0.16	92.9	87.3	92.0	8.7	7.2	10.7
s38584	330	29.5	0.14	105.5	95.2	91.0	29.7	100	0.15	96.0	83.5	86.5	9.0	12.3	4.9
ispd09f22	91	24.6	<0.01	9.4	8.0	10.2	28.8	10	<0.01	9.1	7.8	8.0	3.3	2.5	21.6
ispd09f32	190	25.4	<0.01	21.6	22.1	22.8	25.7	49	<0.01	18.9	19.3	19.1	12.5	12.7	16.2
Average												10.2	11.9	12.8	

TABLE III

COMPARISON OF RESULTS (CLK-NOISE: USING **Level-I** BUF/INV, CLK-NOISE: USING **Level-G, H, I, J** BUF/INV, SKEW_BUDGET ≤ 30 PS)

Benchmark Info		CLK-NOISE (Using Level-I)					CLK-NOISE (Using Level-G, H, I, J)					Improvement		
Circuit	L	Skew (ps) (≤30)	Run Time (s)	Peak Curr. (mA)	Power Noise (mV)	Ground Noise (mV)	Skew (ps) (≤30)	Run Time (s)	Peak Curr. (mA)	Power Noise (mV)	Ground Noise (mV)	Peak Curr. (%)	Power Noise (%)	Ground Noise (%)
s5378	40	28.9	<0.01	13.2	9.6	9.9	29.0	0.01	10.1	9.3	7.3	23.8	3.1	26.7
s9234	50	29.3	<0.01	18.5	13.6	14.0	29.1	0.01	12.5	12.0	9.6	32.5	11.8	31.7
s13207	134	29.5	0.03	46.4	36.6	37.9	29.6	0.03	30.0	36.8	30.7	35.2	-0.5	18.9
s15850	133	28.4	0.04	42.3	36.9	36.7	28.7	0.04	29.7	37.2	32.3	29.9	-0.8	12.1
s35932	407	29.8	0.27	97.0	90.0	92.8	29.0	0.27	40.4	54.5	44.1	58.3	39.4	52.5
s38417	343	29.9	0.16	92.9	87.3	92.0	29.3	0.16	68.0	97.0	83.6	26.8	-11.1	9.1
s38584	330	29.7	0.15	96.0	83.5	86.5	29.9	0.15	69.0	65.8	80.6	28.1	21.2	6.9
ispd09f22	91	28.8	<0.01	9.1	7.8	8.0	27.3	<0.01	9.1	7.1	7.3	0.0	9.0	8.8
ispd09f32	190	25.7	<0.01	18.9	19.3	19.1	29.7	<0.01	19.0	19.7	19.4	-0.5	-2.1	-1.6
Average												26.0	7.8	18.3

noise maps of the designs produced by Polarity-only and CLK-NOISE for s35932. (s35932 has the largest number of sinks in ISCAS89 benchmarks.) Tables IV and V show the numbers of feasible intervals without and with the application of Theorem 4, respectively. The numbers in parentheses indicate the run times by CLK-NOISE using the feasible intervals. In addition, Table VI shows peak powers (mV) by CLK-NOISE with different zone sizes (μm^2). The size $l \times l$ of each zone is denoted by l in the table. The next l is scaled up to $l \times 2^{\frac{1}{4}}$. Since power/ground noise is a local effect, the selected zone size will affect the value of power/ground noise. For example, in Table VI for s9234 and s35932 the peak power is lowest when $l = 500$ while for s38417 the peak power is lowest when $l = 594$. To find the lowest peak noise, CLK-NOISE needs to be applied iteratively while increasing the zone size incrementally and stop when there is no more reduction on the measured peak noise. Finally, Table VII shows the comparison of results by the phase 1 of Ryu’s polarity assignment algorithm and CLK-NOISE. Since phase 1 of Ryu’s algorithm performs a polarity assignment only to sinks without considering skew constraint, the peak current is absolutely small, but the skew could be very large, which is shown to be 6.2–7.4 times larger

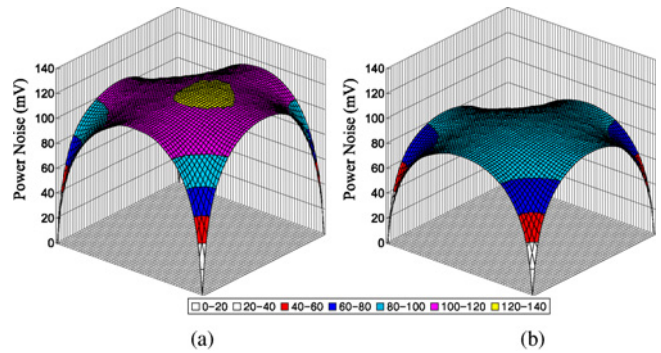


Fig. 10. Comparison of the power noise map for S35932 in Table II. (a) Power noise map by Polarity-only. (b) Power noise map by CLK-NOISE.

than that of CLK-NOISE. [Note that phase 2 of Ryu’s algorithm is practically unacceptable because in phase 2, the algorithm puts the same noise weight (i.e., the same amount of contribution to peak current) on the polarity assignment to non-sinks as that on the polarity assignment to sinks in phase 1.]

- Assessing the effectiveness of CLK-NOISE for polarity assignment and buffer sizing on reducing P/G noise. We relaxed the buffer and inverter library to have level-G, H, I, J BUF/INV for CLK-NOISE. The skew bound

TABLE IV

NUMBER OF ALL FEASIBLE INTERVALS EXTRACTED WITHOUT USING THEOREM 4 AND THE RUN TIMES (S) USING THE INTERVALS ARE IN PARENTHESES

Skew Budget (ps)	Circuit								
	s5378	s9234	s13207	s15850	s35932	s38417	s38584	ispd09f22	ispd09f32
20	46(0.01)	86(0.01)	173(0.09)	190(0.06)	566(0.62)	438(0.53)	432(0.38)	61(0.00)	72(0.01)
30	66(0.01)	145(0.02)	253(0.14)	285(0.10)	849(1.15)	639(0.86)	639(0.66)	89(0.00)	108(0.02)
40	104(0.03)	218(0.04)	439(0.37)	469(0.28)	1370(3.10)	1119(2.45)	1090(1.79)	141(0.01)	170(0.04)
50	109(0.04)	293(0.05)	449(0.50)	512(0.38)	1418(3.94)	1150(3.34)	1131(2.37)	145(0.01)	180(0.05)

TABLE V

NUMBER OF FEASIBLE INTERVALS REDUCED BY USING THEOREM 4 AND THE RUN TIMES (S) USING THE INTERVALS ARE IN PARENTHESIS

Skew Budget (ps)	Circuit								
	s5378	s9234	s13207	s15850	s35932	s38417	s38584	ispd09f22	ispd09f32
20	46(0.01)	86(0.01)	173(0.09)	190(0.06)	566(0.63)	438(0.54)	432(0.38)	61(0.00)	72(0.01)
30	9(0.01)	8(0.01)	85(0.03)	122(0.04)	316(0.27)	222(0.16)	224(0.15)	33(0.00)	41(0.00)
40	9(0.01)	17(0.02)	204(0.17)	34(0.02)	599(1.40)	513(1.13)	479(0.79)	7(0.00)	80(0.00)
50	2(0.01)	54(0.01)	20(0.03)	56(0.05)	77(0.25)	49(0.19)	60(0.16)	5(0.00)	16(0.01)

TABLE VI

PEAK POWERS (MV) BY CLK-NOISE WITH DIFFERENT ZONE SIZES (μM)

Zone Size (l)	Circuit								
	s5378	s9234	s13207	s15850	s35932	s38417	s38584	ispd09f22	ispd09f32
250	9.6	13.2	38.2	39.3	58.8	99.3	68.3	7.7	20.5
297	9.5	13.1	37.5	38.7	57.2	98.9	67.3	7.2	20.2
353	9.2	12.6	36.6	37.6	56.6	97.7	66.3	6.5	20.5
420	9.0	12.4	36.5	37.6	55.2	97.3	66.1	6.9	19.9
500	9.3	12.0	36.8	37.2	54.5	97.0	65.8	6.4	19.6
594	9.2	12.6	36.5	37.4	54.5	96.9	65.6	7.1	19.4
707	9.2	12.6	36.5	37.1	54.7	97.2	65.5	6.1	19.4
840	9.2	12.2	36.4	37.0	55.8	97.0	65.5	6.1	19.4
1000	9.2	12.7	36.5	37.1	55.9	98.1	65.5	6.1	19.4
1189	9.2	12.7	36.5	37.0	57.3	97.2	65.5	6.1	19.4

The size $l \times l$ of each zone is denoted by l (μm) in the table. The next l is scaled up to $l \times 2^{\frac{1}{4}}$.

TABLE VII

COMPARISON OF RESULTS BY RYU'S ALGORITHM 11 AND CLK-NOISE USING LEVEL-I BUF/INV

Circuits	Ryu's Algorithm [11]		CLK-NOISE	
	Peak Current (mA)	Skew (ps)	Peak Current (mA)	Skew (ps)
s5378	9.9	218.5	13.2	28.9
s9234	12.5	222.1	18.5	29.3
s13207	13.5	223.9	46.4	29.5
s15850	1.5	208.0	42.3	28.4
s35932	0.2	224.4	97.0	29.8
s38417	0.6	219.8	92.9	29.9
s38584	0.6	223.9	96.0	29.7
ispd09f22	7.0	186.5	9.1	28.8
ispd09f32	2.6	191.3	18.9	25.7

was also set to 30ps for a fair comparison with the result by CLK-NOISE using level-I BUF/INV only. Table III summarizes the simulation results. We can see the improvements of power and ground noises are 7.8% and 18.3% on average, respectively.

- Assessing the effectiveness of CLK-NOISE for exploring design space. The curves in Fig. 11(a) and (b) show the changes of improvements by CLK-NOISE on

maximum power noise and total peak current by varying clock skew constraint, respectively. It is true that as the skew bound is relaxed, it is more likely that buffers and inverters with low current consumption are to be selected and allocated (i.e., by buffer/inverter sizing). Consequently, CLK-NOISE can be more effective as the skew bound increases,⁸ as validated by the curves in Fig. 11. We can notice from the slopes of the curves in Fig. 11 that the improvements are saturated at the skew bound around 40ps for all tested designs. This is because for different values of clock skew bound, the best, in terms of power/ground noise, buffer and inverter types to be used can be different, but beyond the skew bound around 40 ps, there are no such "new" buffers and inverters in the library that are best suited to the clock skew bound.

- Assessing the effectiveness of CLK-NOISE on considering thermal variations. To produce a set of thermal map instances, we performed thermal simulation by using the ADI-based thermal simulator package in [28]. For testing ISCAS89 benchmark circuits, the power density of each thermal node is randomly assigned to a value

⁸This is due to enlarged feasible time intervals.

TABLE VIII
TEMPERATURE INFORMATION UNDER THERMAL PROFILES P_2 , P_3 , P_4 , P_5 , AND P_6 PRODUCED BY THERMAL SIMULATOR ADI [29]

Circuit	Profile P_2		Profile P_3		Profile P_4		Profile P_5		Profile P_6	
	Min/Max	Avg/Stdev	Min/Max	Avg/Stdev	Min/Max	Avg/Stdev	Min/Max	Avg/Stdev	Min/Max	Avg/Stdev
S5378	27.0/27.3	27.2/0.1	27.5/29.0	28.6/0.4	29.1/34.1	32.4/1.3	33.2/44.5	40.6/2.9	41.7/60.2	53.8/4.8
S9234	27.1/27.3	27.2/0.1	27.9/29.0	28.6/0.3	30.1/34.3	32.7/1.1	35.4/46.2	41.8/2.8	45.9/65.9	57.7/5.1
S13207	27.0/27.3	27.2/0.1	27.6/29.2	28.7/0.4	29.4/35.0	33.2/1.5	33.9/49.4	43.8/4.2	44.0/77.3	64.2/8.9
S15850	27.0/27.3	27.2/0.0	27.5/29.0	28.7/0.3	29.2/34.3	33.3/1.0	33.5/48.1	44.2/3.2	42.9/76.8	66.0/7.6
S35932	27.1/27.3	27.3/0.0	27.8/29.1	28.9/0.2	29.8/34.6	33.9/0.9	35.5/49.2	46.6/2.8	50.7/82.1	74.5/6.9
S38417	27.0/27.3	27.3/0.0	27.7/29.1	28.9/0.3	29.5/34.7	33.8/1.0	34.0/49.5	46.0/3.3	44.4/82.3	71.7/8.5
S38584	27.0/27.3	27.2/0.0	27.6/29.1	28.8/0.3	29.3/34.6	33.6/1.1	33.8/48.9	45.7/3.3	45.0/80.9	71.7/8.4

The units are in degrees Celsius(deg).

TABLE IX
VALUES OF PEAK CURRENT AND CLOCK SKEW PRODUCED BY CLK-NOISE- t WITH THE CONSTRAINT OF SKEW BOUND = 50 PS

Circuit	Thermal Profiles											
	(P_1)		(P_1, P_2)		(P_1, P_2, P_3)		(P_1, P_2, P_3, P_4)		$(P_1, P_2, P_3, P_4, P_5)$		$(P_1, P_2, P_3, P_4, P_5, P_6)$	
	Peak Current (mA)	Clock Skew (ps)	Peak Current (mA)	Clock Skew (ps)	Peak Current (mA)	Clock Skew (ps)	Peak Current (mA)	Clock Skew (ps)	Peak Current (mA)	Clock Skew (ps)	Peak Current (mA)	Clock Skew (ps)
S5378	4.12	47.04	4.12	47.04	4.12	47.04	4.12	47.04	4.09	44.85	4.04	48.12
			4.12	47.14	4.12	47.14	4.12	47.14	4.10	44.83	4.04	48.03
					4.13	47.51	4.13	47.51	4.10	44.76	4.04	47.74
							4.13	48.52	4.10	44.83	4.04	47.20
S9234	4.67	48.68	4.67	48.68	4.67	48.68	5.17	49.81	6.03	47.31	6.74	47.55
			4.67	48.62	4.67	48.62	5.17	49.68	6.03	47.02	6.74	47.17
					4.67	49.85	5.17	49.20	6.03	45.97	6.73	45.75
							5.17	49.71	6.03	44.80	6.73	41.68
S13207	8.46	49.96	8.47	49.32	8.36	49.23	8.98	49.23	11.48	49.23		
			8.47	49.29	8.36	49.09	8.98	49.09	11.48	49.09		
					8.35	49.18	8.97	49.01	11.46	49.01		
							8.96	49.56	11.44	48.98		
S15850	0.65	49.85	0.65	49.85	0.65	48.88	0.65	48.88	0.64	48.88		
				49.96	0.65	48.73	0.65	48.63	0.65	48.63		
					0.65	49.89	0.65	47.68	0.64	47.68		
							0.65	49.60	0.64	47.36		
S35932	0.02	49.74	0.02	49.74	0.02	48.24	0.02	49.15				
				49.81	0.02	47.32	0.02	48.46				
					0.02	49.78	0.02	48.21				
							0.02	49.93				
S38417	0.11	49.98	0.11	49.54	0.11	48.75	0.11	49.77				
			0.11	49.86	0.11	48.02	0.11	48.72				
					0.11	49.79	0.11	45.66				
							0.11	49.20				
S38584	0.12	49.89	0.12	49.89	0.12	49.87	0.11	49.68				
				49.65	0.12	48.74	0.11	48.24				
					0.12	50.00	0.11	46.46				
							0.12	48.91				

The vertical, top to bottom, arrangement of the multiple values at an entry matches the horizontal, left to right, arrangement of the thermal profiles on the corresponding column.

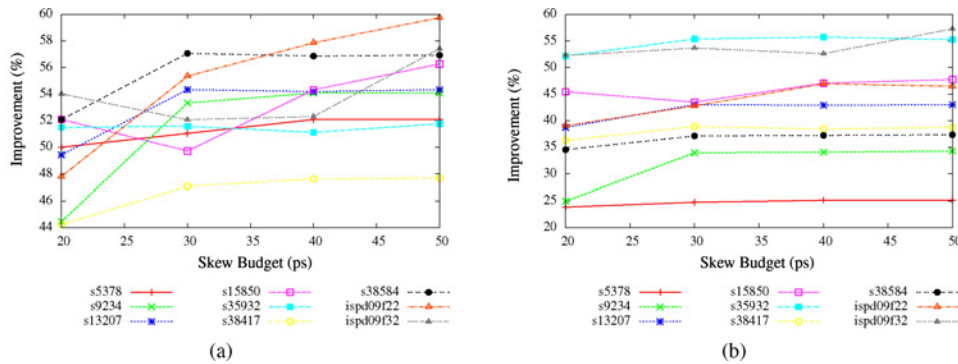


Fig. 11. Curves showing the changes of improvement, compared to the unassigned tree, by varying clock skew bound. (a) Maximum power noise. (b) Total peak current.

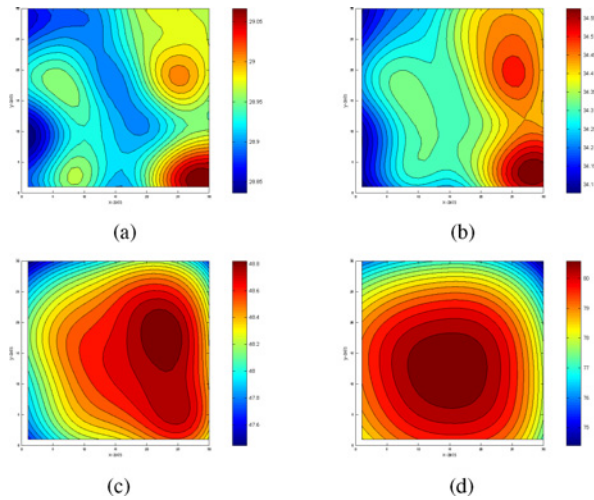


Fig. 12. Maps of thermal profiles P_2 , P_3 , P_4 , and P_5 for S38584. (a) P_2 ($t=1.4\text{ms}$). (b) P_3 ($t=2.5\text{ms}$). (c) P_4 ($t=5.0\text{ms}$). (d) P_5 ($t=10\text{ms}$).

in between $1.85 \times 10^{14} \text{ W/m}^3$ and $5.54 \times 10^{14} \text{ W/m}^3$, as suggested by the example input specification [29] of the simulator. In addition, the position and geometric information is given to the simulator by $\Delta x = 100 \mu\text{m}$ and $\Delta y = 100 \mu\text{m}$, and the size to contain circuit by $6000 \mu\text{m} \times 6000 \mu\text{m}$. We extract the thermal simulation profiles at the times of 0, 13, 25, 50, 100 and 200 iterations of simulation where we labeled the profiles as P_1, P_2, \dots, P_6 , respectively. We set the time increment parameter Δt in the simulator [28] to 100 ns, thus the duration of circuit execution for the last profile P_6 being $t = 20 \text{ ms}$. For example, Fig. 12 shows the thermal maps of P_2, P_3, P_4 , and P_5 for circuit s38584. The minimum and maximum temperature, and the average and standard deviation of the temperature under each thermal profile are shown in Table VIII. With the assumption that the thermal variance has negligible effect on unit length capacitance, we calculate the interconnection wire resistance per unit length by [21]

$$r = \rho_0 \{1 + \beta \cdot T(x, y)\} \quad (8)$$

where ρ_0 is the unit resistance per unit at 0 degC, β is the temperature coefficient of resistance (1/degC),

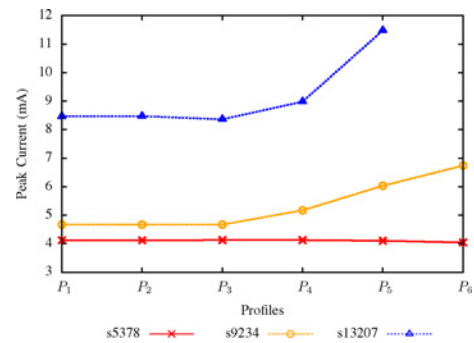


Fig. 13. Curves showing the changes of peak current values as the number of thermal profiles considered increases from P_1 only (marked as P_1), P_1 and P_2 only (marked as P_2), \dots , finally P_1 through P_6 (marked as P_6 with skew bound = 50 ps (i.e., results in Table IX).

and $T(x, y)$ is the temperature at point (x, y) . In this experiment $\beta = 0.0068(1/\text{degC})$ [30]. For wire model, the π network is used for simulation as TACO algorithm [22] does.

CLK-NOISE-t is then applied to each of thermal profiles, followed by performing SPICE simulation to produce the noise data. Table IX shows the values of peak currents and clock skews for different sets of profiles under the clock skew bound of 50 ps. The red colored number in each entry of peak current column indicates the worst peak value among the profiles in the corresponding column. From the two tables, we observe a consistent trend: the peak current increases (or decreases) as more (or less) thermal profiles are considered. Finally, Fig. 13 shows how the peak current values change as the circuit execution is performed, starting from considering P_1 only, considering P_1 and P_2 only, \dots , finally considering P_1 through P_6 for circuits s5378, s9234, and s13207 with skew bound = 50ps.

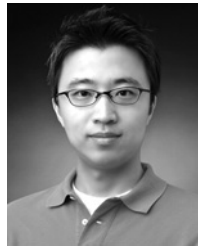
VI. CONCLUSION

This paper proposed a comprehensive solution to the integrated problem of buffer sizing and polarity assignment for minimizing power/ground noise. Precisely, the key contributions of this paper were: 1) the proof of intractability of the

problem; 2) a precise estimation of peak current by clock buffers/inverters; 3) a practically efficient optimal algorithm based on dynamic programming for the problem; 4) a systematic design flow for reducing power/ground noise using two types of “zone” concept; and 5) considering the effect of thermal variations.

REFERENCES

- [1] S. Chowdury and J. Barkatullah, “Estimation of maximum currents in MOS IC logic circuits,” *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 9, no. 6, pp. 642–654, Jun. 1990.
- [2] L. H. Chen, M. M. Sadowska, and F. Brewer, “Buffer delay change in the presence of power and ground noise,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 11, no. 3, pp. 461–473, Jun. 2003.
- [3] L. Benini, P. Vuillod, A. Bogliolo, and G. D. Micheli, “Clock skew optimization for peak current reduction,” *J. VLSI Signal Process.*, vol. 16, nos. 2–3, pp. 117–130, 1997.
- [4] N. H. E. Weste and D. Harris, *CMOS VLSI DESIGN, A Circuits and Systems Perspective*, 3rd ed. Reading, MA: Addison-Wesley, 2005.
- [5] A. Vittal, H. Ha, F. Brewer, and M. M. Sadowska, “Clock skew optimization for ground bounce control,” in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 1996, pp. 395–399.
- [6] S.-H. Huang, C.-H. Chang, and Y.-T. Nieh, “Fast multi-domain clock skew scheduling for peak current reduction,” in *Proc. IEEE Asia South Pacific Des. Autom. Conf.*, 2006, pp. 254–259.
- [7] Y.-T. Nieh, S.-H. Huang, and S.-Y. Hsu, “Minimizing peak current via opposite-phase clock tree,” in *Proc. IEEE/ACM Des. Autom. Conf.*, 2005, pp. 182–185.
- [8] R. Samanta, G. Venkataraman, and J. Hu, “Clock buffer polarity assignment for power noise reduction,” in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 2006, pp. 558–562.
- [9] P.-Y. Chen, K.-H. Ho, and T. Hwang, “Skew aware polarity assignment in clock tree,” in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 2007, pp. 376–379.
- [10] P.-Y. Chen, K.-H. Ho, and T. Hwang, “Skew-aware polarity assignment in clock tree,” *ACM Trans. Des. Autom. Electron. Syst.*, vol. 14, no. 2, pp. 31:1–31:17, Mar. 2009.
- [11] Y. Ryu and T. Kim, “Clock buffer polarity assignment combined with clock tree generation for power/ground noise minimization,” in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 2008, pp. 416–419.
- [12] M. Kang and T. Kim, “Clock buffer polarity assignment considering the effect of delay variations,” in *Proc. IEEE Int. Symp. Quality Electron. Design*, 2010, pp. 69–74.
- [13] J. Lu and B. Taskin, “Clock buffer polarity assignment considering capacitive load,” in *Proc. IEEE Int. Symp. Quality Electron. Des.*, 2010, pp. 765–770.
- [14] H. Jang and T. Kim, “Simultaneous clock buffer sizing and polarity assignment for power/ground noise minimization,” in *Proc. IEEE/ACM Des. Autom. Conf.*, 2009, pp. 794–799.
- [15] K. D. Bosse, A. B. Kahng, “Zero-skew clock routing with minimum wirelength,” in *Proc. IEEE Int. ASIC Conf.*, 1992, pp. 111–115.
- [16] Q. Zhu, *Power Distribution Network Design for VLSI*. New York: Wiley, 2004.
- [17] M. R. Garey and D. S. Johnson, *Computers and Intractability, A Guide to the Theory of NP-Completeness*. San Francisco, CA: Freeman, 1979.
- [18] S. Martello and P. Toth, *Knapsack Problems*. New York: Wiley, 1990.
- [19] P. Gronowski, W. J. Bowhill, R. P. Preston, M. K. Gowan, and R. L. Allmon, “High performance microprocessor design,” *IEEE J. Solid-State Circuits*, vol. 33, no. 5, pp. 676–686, May 1998.
- [20] 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 Integr. Circuits Conf.*, 2001, pp. 233–236.
- [21] K. Banerjee, A. H. Ajami, and M. Pedram, “Analysis and optimization of thermal issues in high-performance VLSI,” in *Proc. ACM Int. Symp. Phys. Design*, 2001, pp. 230–237.
- [22] M. Cho, S. Ahmed, and D. Z. Pan, “TACO: Temperature aware clock tree optimization,” in *Proc. ACM/IEEE Int. Conf. Comput.-Aided Des.*, 2005, pp. 582–587.
- [23] J. Minz, X. Zhao, and S. K. Lim, “Buffered clock tree synthesis for 3-D ICs under thermal variations,” in *Proc. IEEE Asia South Pacific Design Autom. Conf.*, 2009, pp. 504–509.
- [24] *Placement of ISCAS89 Benchmark Circuits* [Online]. Available: <http://www.ece.wisc.edu/vlsi/tools/iscas-placement/index.html>
- [25] R. Chaturvedi and J. Hu, “Buffered clock tree for high quality IC design,” in *Proc. IEEE/ACM Int. Symp. Quality Electron. Design*, 2004, pp. 381–386.
- [26] M. Edahiro, “A clustering-based optimization algorithm in zero-skew routing,” in *Proc. IEEE/ACM Des. Autom. Conf.*, 1993, pp. 612–616.
- [27] *FSCOH_D 0.13 μm Standard Cell Databook*, Faraday Technology, Hsinchu, Taiwan, 2004.
- [28] T. Wang and C. C. Chen, “3-D thermal-ADI: A linear-time chip level transient thermal simulator,” *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 21, no. 12, pp. 1434–1445, Dec. 2002.
- [29] *3-D Thermal-ADI Simulator, the Binary Executable File and Sample Input* [Online]. Available: http://www.ece.wisc.edu/~vlsi/3D_Thermal_ADI.htm
- [30] T. Wang and C. Chen, “Power-delivery networks optimization with thermal reliability integrity,” in *Proc. ACM Int. Symp. Phys. Design*, 2004, pp. 124–131.



Hochang Jang (S'08) received the B.S. and M.S. degrees in electrical engineering from Seoul National University, Seoul, Korea, in 2007 and 2009, respectively.

He is currently a Research Student with the School of Electrical Engineering and Computer Science, Seoul National University. His current research interests include computer-aided design with an emphasis on clock network related analysis and optimization, including skew and noise.



Deokjin Joo received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 2009. He is currently working toward the M.S. degree in the School of Electrical Engineering and Computer Science, Seoul National University.

His current research interests include low power and thermal resilient design.



Taewhan Kim (SM'08) received the B.S. degree in computer science and statistics and the M.S. degree in computer science from Seoul National University, Seoul, Korea, and the Ph.D. degree in computer science from the University of Illinois at Urbana-Champaign, Urbana, in 1993.

Currently, he is a Professor with the School of Electrical Engineering and Computer Science, Seoul National University. His current research interests include embedded systems and computer-aided design of integrated circuits.