WaveMin: A Fine-Grained Clock Buffer Polarity Assignment Combined with Buffer Sizing

Deokjin Joo
School of Electrical Engineering and Computer Science
Seoul National University, Korea
jdj@ssl.snu.ac.kr

Taewhan Kim
School of Electrical Engineering and Computer Science
Seoul National University, Korea
tkim@ssl.snu.ac.kr

ABSTRACT

The clock buffer polarity assignment is one of the effective design schemes to mitigate the power/ground noise caused by the clock signal propagation. This work overcomes two fundamental limitations of the conventional clock buffer polarity assignment methods, which are (1) the unawareness of the signal delay (i.e., arrival time) differences to the leaf buffering elements and (2) the ignorance of the effect of the current fluctuation of non-leaf buffering elements on the total peak current waveform. Clearly, not addressing (1) and (2) in polarity assignment may cause a severe inaccuracy on the peak current estimation, which results in unnecessarily high peak current. To overcome the limitations, we propose a completely new fine-grained approach to the clock buffer polarity assignment combined with buffer sizing, formulating the problem into a multi-objective shortest path problem and solving it effectively. The experimental results show that the proposed method is able to produce designs with 17% lower peak current and 20% lower power noise on average compared the results produced by the best ever known method.

Categories and Subject Descriptors

B.6.3 [**Hardware**]: Logic Design; B.8.1 [**Hardware**]: Performance and Reliability

General Terms

Algorithms, Design, Performance, Reliability

Keywords

Polarity assignment, buffer sizing, power/ground noise

1. INTRODUCTION

As the CMOS process technology scales down, it becomes possible to use much lower supply voltages in VLSI design. The lowered supply voltage in turn reduces the power consumption in the circuit. However, the use of lower supply voltage causes the circuit to be more susceptible to power and ground noise, i.e., voltage fluctuations in the power and ground rails. This noise also adversely affects circuit performance such as the delay of switching signal [1, 2]. The major sources of the voltage fluctuations come from the input and output drivers and the internal logic circuitry, especially those that switch near either rising or falling edge of clock signal [3]. In a synchronous circuit, the buffered clock tree itself consumes a considerable amount of power since its clock signal is one of the most actively switching sources in the circuit. It is reported that the amount of clock power consumed by a clock distribution network with clocked loads typically accounts for one third to one half of the total chip power dissipation [4].

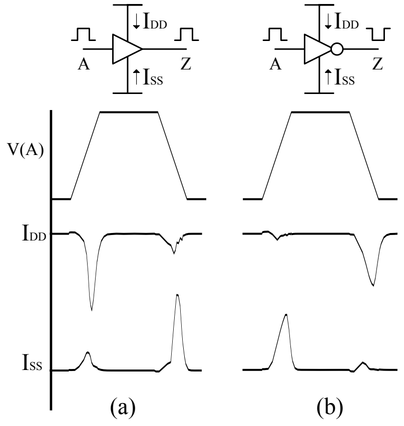


Figure 1: The idea behind buffer polarity assignment. Buffers exhibit high I_{DD}/Gnd current at rising/falling edge of clock signal while inverters emit high I_{DD}/Gnd current at falling/rising edge.

It has been known that selectively assigning (positive or negative) polarities to (initial) clock buffering elements by properly replacing some of the buffering elements with inverters and the other with buffers is an effective way of reducing the power/ground noise.[1] Fig. 1 illustrates the basic idea behind the polarity assignment. A buffer is a chain of unequally sized two inverters and exhibits current noise as

DAC 2011, June 5-10, 2011, San Diego, California, USA.
Copyright 2011 ACM 978-1-4503-0636-2/11/06 ...$10.00.

[1] A buffering element is said to be 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.

shown in Fig. 1(a); at the rising edge of clock signal, the buffer charges drawing a high I_{DD} current while drawing a low I_{SS} current. For inverters, the opposite case happens as shown in Fig. 1(b). Thus, by mixing buffers and inverters in the buffered clock tree, the designer is able to disperse the current noise from/to V_{DD}/Gnd at rising/falling edge of clock signal. By observing the current waveforms in Fig. 1, several techniques of buffer polarity assignment have been proposed [5, 6, 7, 8, 9, 10, 11]. The two critical flaws of all the previous works are (1) the unawareness of the signal delay (i.e., arrival time) differences to the leaf nodes and (2) the ignorance of the effect of non-leaf nodes' current fluctuations on the total peak current waveform. Clearly, not addressing (1) and (2) in polarity assignment may cause a severely inaccurate peak current (or peak power/ground noise) estimation. By addressing the limitations, we propose a completely new solution to the problem of clock buffer polarity assignment with buffer sizing, employing a fine-grained noise estimation technique, rather than using the peak current values only at the four time sampling points of (V_{DD}, rising), (V_{DD}, falling), (Gnd, rising), and (Gnd, falling) as adopted by the previous works.

2. OBSERVATIONS

Since the leaf buffering elements are the major contributor to the (total) peak current as illustrated by [7], our work also focuses on the polarity assignment on leaf buffering elements. This section illustrates how the previous works on polarity assignment lack the accuracy in estimating peak current.

Let us consider the problem of assigning polarity to the four leaf nodes on the clock tree in Fig. 2(a). All possible combinations of polarity assignment by replacing each node with buffer or inverter and the corresponding value of total peak current obtained by summing the peak current values of the nodes are summarized in the table in Fig. 2(b) where P and N indicate positive and negative polarities, respectively.

From the table, we can see that the fourth assignment (N, N, P, P) produces the lowest value of total peak current, which is 387μA. The dark dotted curve in Fig. 2(c) shows the accumulated current waveform of the leaf nodes for the polarity assignment (N, N, P, P). On the other hand, the blue solid curve in Fig. 2(d) shows the accumulated current waveform of *all* nodes including the two non-leaf buffers, from which we can see that the actual value of total peak current is unbalanced, i.e., skewed to the left (at time = 2.2ps), resulting in the peak current of 691.79μA. However, the dark dotted curve in Fig. 2(d) shows the current waveform of the leaf nodes when the polarity assignment is (N, N, P, N), thus the peak is skewed to the right. The blue solid curve in Fig. 2(d) which shows the resulting waveform of *all* nodes however has much reduced peak current, which is around 542μA. This observation implies that the current fluctuation by non-leaf nodes should be taken into account during the process of polarity assignment of leaf nodes.

Another observation from the current waveforms in Fig. 2(d) indicates that by knowing that some leaf nodes may switch at different times due to unequal clock signal delays, the current fluctuation by the non-leaf nodes contributes differently to the (accumulated) current waveforms at the time when the leaf nodes switch. Thus, any time instance in a certain time interval (e.g., time in $[1.0, 4.0]$ in Fig. 2(d)) can be a time sampling candidate at which peak current may occur.

3. PROBLEM FORMULATION

Problem 1 (WaveMin). *(Polarity assignment/buffer sizing for peak current minimization) Given an available buffer type set B, an inverter type set I, a sub-area that holds set L of leaf buffering elements, time sampling slots S, clock skew constraint κ, find a mapping function $\phi : L \mapsto \{B \cup I\}$ that minimizes the quantity of*

$$\max_{s \in S} \left\{ \sum_{e_i \in L} noise(\phi(e_i), s) \right\} \tag{1}$$

$$s.t\ t_{skew}(\phi) \leq \kappa$$

where $t_{skew}(\phi)$ is the clock skew induced by mapping ϕ and $noise(\phi(e_i), s)$ is the value of peak current estimation at time sampling point s caused by the switch of node e_i when it is assigned with type $\phi(e_i)$. $noise(\phi(e_i), s)$ is assumed to be independent of the mapping choice of $\phi(e_j)$, $i \neq j$.

Note that the set of times S not only represents the discretely sampled times of interest, such as the rising and falling edges of the clock signal, but also the power line of interest, V_{DD} and Gnd. For example, S may have four times, which are the times when V_{DD} and Gnd are on the rising edge of clock tree, and V_{DD} and Gnd on the falling edge. As the size of S increases by including more (meaningful) time sampling points, the peak current estimation would be more accurate.

In the following, we show that WaveMin is NP-complete by reducing the decision version of PeakMin [10], which is also NP-complete, to decision-WaveMin problem which is the decision version of WaveMin.

Problem 2 (decision-WaveMin). *For a* WaveMin *instance with (L, B, I, S, κ) and a constant c, is there a mapping ϕ such that the value of expression 1 is less than or equal to c?*

Problem 3 (decision-PeakMin). *For a sub-area that contains set L of leaf buffering elements, a buffer type set B, an inverter type set I, clock skew bound κ, and a constant c, find a mapping function $\phi : L \mapsto \{B \cup I\}$ such that the quantity in expression 2 is less than or equal to c,*

$$\max \left\{ \sum_{\phi(e_i) \in B} peak(\phi(e_i)), \quad \sum_{\phi(e_i) \in I} peak(\phi(e_i)) \right\} \tag{2}$$

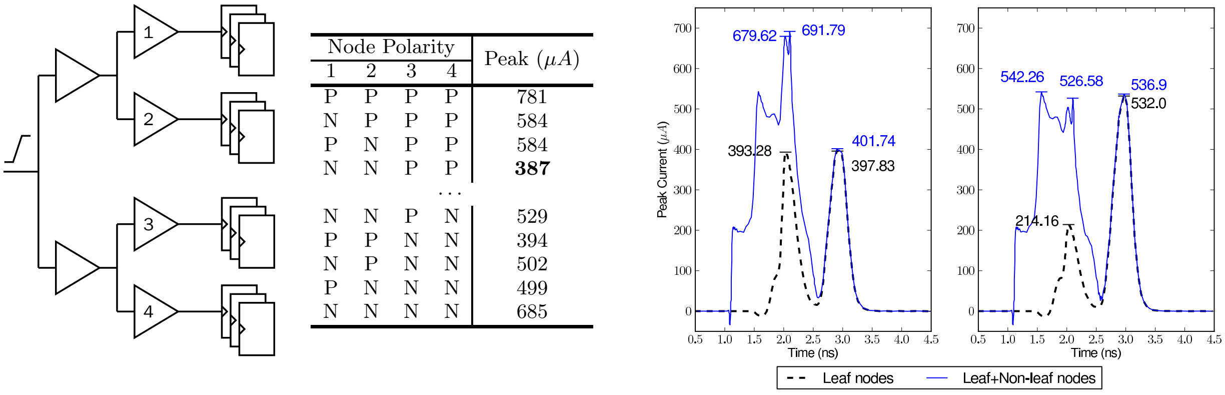
$$s.t.\ t_{skew}(\phi) \leq \kappa$$

where $t_{skew}(\phi)$ is the clock skew induced by mapping ϕ and $peak(\phi(e_i))$ indicates the amount of peak current on $\phi(e_i)$ over time period $[0, \infty)$.

Theorem 1. decision-PeakMin *is NP-complete [10].*

Theorem 2. decision-WaveMin *is NP-complete.*

Proof. We show that decision-PeakMin is a special case of decision-WaveMin: by passing all parameters of decision-PeakMin to decision-WaveMin and letting $|S| =$



Node Polarity				Peak (μA)
1	2	3	4	
P	P	P	P	781
N	P	P	P	584
P	N	P	P	584
N	N	P	P	**387**
			···	
N	N	P	N	529
P	P	N	N	394
N	P	N	N	502
P	N	N	N	499
N	N	N	N	685

(a) (b) (c) (d)

Figure 2: (a) A simple clock tree with four leaf nodes. (b) Its expected peak current value. The fourth assignment (N, N, P, P) produces the lowest value of total peak current of 387μA. (c) Current waveforms by non-leaf nodes' noise unaware optimal polarity assignment (= (N, N, P, P) in (b)) to leaf nodes. Dark dotted line is the current waveform from leaf nodes only while blue solid line shows the total current from all clock nodes. (d) Current waveforms resulting from non-leaf nodes' noise aware optimal polarity assignment (= (N, N, P, N) in (b)) to leaf nodes.

2 for V_{DD} rail at the rising and falling edges of the clock, all instances of DECISION-WAVEMIN can be exactly mapped with DECISION-PEAKMIN. For one time sampling point, the summation term in expression 1 becomes V_{DD} rail at the rising edge of the clock, which is the summation term for buffers in expression 2 and for the other time point, becomes the term for inverters. This is because in PEAKMIN, only V_{DD} rail is considered and switching polarity "migrates" the noise from/to "inverter noise time slot" to/from "buffer noise time slot". □

Since WAVEMIN intends to be a generalized version of PEAKMIN [10], we borrow many concepts from the work in [10] such as feasible time interval to satisfy the skew constraint κ, and zone concept which divides the circuit into tiles to tackle the whole problem tile by tile basis.

4. THE PROPOSED ALGORITHM

4.1 Overview

Since power/ground noise is a local effect, we first divide the circuit into several zones and apply our algorithm, called CLKWAVEMIN, to the zones one by one to minimize the peak current at each zone, which targets the maximum peak current value as the objective cost to be minimized, among the zones. As a pre-processing step, CLKWAVEMIN requires to prepare some timing information to be used in the polarity assignment of leaf nodes. For each pair of leaf node e_i and buffer/inverter type α_j, $e_i \in L$ and $\alpha_j \in B \cup I$, we measure the arrival time to e_i from clock source when $\phi(e_i) = \alpha_j$. We arrange the arrival times in non-increasing order: $a_1, a_2, \cdots, a_m$, where $m = |L|(|B| + |L|)$ and a_1 and a_m are the latest and earliest arrival times, respectively. Then, for each time interval $[t - \kappa, t]$, $t = a_1, a_2, \cdots, a_m$, we apply CLKWAVEMIN to every zone and compute the peak current values among the zones. This time interval based polarity assignment ensures to meet the clock skew constraint. (Note that for a time interval $[t - \kappa, t]$, the candidate ("feasible") buffers and inverters to be assigned to a node are those whose switchings occur in the time interval.) We then select the solution of the polarity assignment with buffer sizing corresponding to the time interval that gives the minimum peak current among all time intervals.

Now, for the rest of this section we focus on the discussion of CLKWAVEMIN to be applied to a time interval $[t - \kappa, t]$ with zone z_i.

We map the WAVEMIN problem as min-max problem (or sometimes called max ordering problem in some literature [12]), which can be solved by solving the multi-objective shortest path (MOSP) problem. The MOSP problem can be solved by a fully polynomial ϵ-approximation algorithm devised by Warburton [13]. Our formulation of WAVEMIN problem to the MOSP problem is described in 4.2, by which we then use Warburton's approximation algorithm to solve the transformed MOSP problem which is a fully polynomial algorithm in time and space criteria: $O(rn^3(n/\epsilon)^{2r})$ time and $O(rn(n/\epsilon)^r)$ space where r is the arc weight dimension and n is the number of vertices in MOSP graph.

4.2 Mapping WaveMin to MOSP problem

We first formally define the MOSP problem.

PROBLEM 4 (MOSP). *Given a directed graph $G = (V, A)$, r dimensional vector weight $w \in W(a)$ for each arc $a \in A$ and two vertices $s, t \in V$, find all Pareto-optimal paths from s to t, where the cost of a path is defined as the sum of arc weights along the path.*

Even for $r = 2$, it is known that the decision version of MOSP problem is NP-complete [14].

We translate WAVEMIN problem to the MOSP problem with Algorithm 1. The algorithm represents polarity/sizing options for each clock tree nodes as a vertex in the MOSP graph. For the example shown in Fig. 3, node e_2 can be assigned as B_2 or I_1 without violating skew bound in interval $[t_1 - \kappa, t_1]$, thus two vertices e_2B_2 and e_2I_1 are created in the MOSP graph.

Algorithm 1 An algorithm to convert ClkWaveMin problem to MOSP problem.

```
function ClkWaveMin 2MOSP($L, \kappa, T, B, I, S$)
    $V \leftarrow \emptyset$
    $A \leftarrow \emptyset$
    for $e_i \in L$ do                                  ▷ Vertex construction
        for $y \in (B(e_i) \cup I(e_i))$ do             ▷ Feasible type $y$
            $v \leftarrow$ new_vertex()
            row($v$) $\leftarrow i$
            column($v$) $\leftarrow y$
            $V \leftarrow V \cup \{v\}$
        end for
    end for
    sink $\leftarrow$ new_vertex()
    row(sink) $\leftarrow |L| + 1$
    source $\leftarrow$ new_vetex()
    row(source) $\leftarrow 0$
    $V \leftarrow V \cup \{$source, sink$\}$
    for $r \in \{0, \cdots, |L| - 1\}$ do               ▷ Arc construction
        for all $u$ s.t row($u$) $= r$ do
            for all $v$ s.t row($v$) $= r + 1$ do
                $a \leftarrow$ new_arc($u, v$)
                weight($a$) $\leftarrow$ noise(row($v$),column($v$))
                $A \leftarrow A \cup \{a\}$
            end for
        end for
    end for
    for all $u$ s.t row($u$) $= |L|$ do                 ▷ Arcs to sink vertex
        $e \leftarrow$ new_arc($u$,sink)
        weight($a$) $\leftarrow$ noise(non-leaf)
        $A \leftarrow A \cup \{a\}$
    end for
    return $G(V, A)$
end function
```

For each pair of vertices, arc (u, v) is created if $row(v) - row(u) = 1$, where $row(e_iX_j) = i$. The multi-dimensional distance $w(u, v)$ is assigned as the estimated noise value when option v is selected for the final assignment, hence the distance of path $s \rightsquigarrow t$ represents the (accumulated) noise, and the vertices in between the path indicate the corresponding assignments. For example, if vertex e_2B_2 is on path $s \rightsquigarrow t$, node e_2 should be assigned with a buffer of type B_2. Fig. 3 summarizes this idea of WaveMin formulation into MOSP problem. The graph degree is $O(|B|+|I|)$ since a node can have at most $|B|+|I|$ incoming and at most $|B|+|I|$ outgoing arcs. Therefore, $O(|A|) = O(2(|B| + |I|)|L| + 2) = O(|L|)$, since there are only limited available types of buffers and inverters, meaning that $|B| + |I|$ is a constant. Lastly, arc weight dimension r equals $|S|$.

The resulting problem is solved with Warburton's algorithm [13] and all approximated Pareto-optimal paths from s to t are found. Among the retrieved paths, we take the path with the minimum worst distance as our ClkWaveMin solution. The path is a valid solution to ClkWaveMin problem because the MOSP graph is directed acyclic since arc (u, v) exists between vertices u and v only if $row(v) - row(u) = 1$. The overall runtime of Warburton's approximation algorithm is given as $O(rn^3(n/\epsilon)^{2r})$ and substituting r and n yields $O(|S||L|^3(|L|/\epsilon)^{2|S|})$. The final selection of min-max solution among Pareto-optimal solutions has execution time of $O(r \times r(n/\epsilon)^r + r(n/\epsilon)^r) = O(|S|^2(|L|/\epsilon)^{|S|})$.

4.3 A Heuristic Algorithm

Algorithm 2 A fast version of ClkWaveMin.

```
procedure GreedyMOSP($G = (V, A)$)
    sum $\leftarrow$ noise(non-leaf)
    while $|V| \neq 0$ do
        best $\leftarrow \infty$
        best_v $\leftarrow$ nil
        for $(u, v) \in A$ do                    ▷ Get least worsening $v$.
            next_sum $\leftarrow$ sum + $w(u, v)$
            if max(next_sum) < max(best) then
                best $\leftarrow$ next_sum
                best_v = $v$
            end if
        end for
        $e_i \leftarrow$ row(best_v)
        $y \leftarrow$ col(best_v)
        Remove nodes in row $e_i$ from $V$
        Assign clock tree node $e_i$ as type $y$
        sum $\leftarrow$ best
    end while
end procedure
```

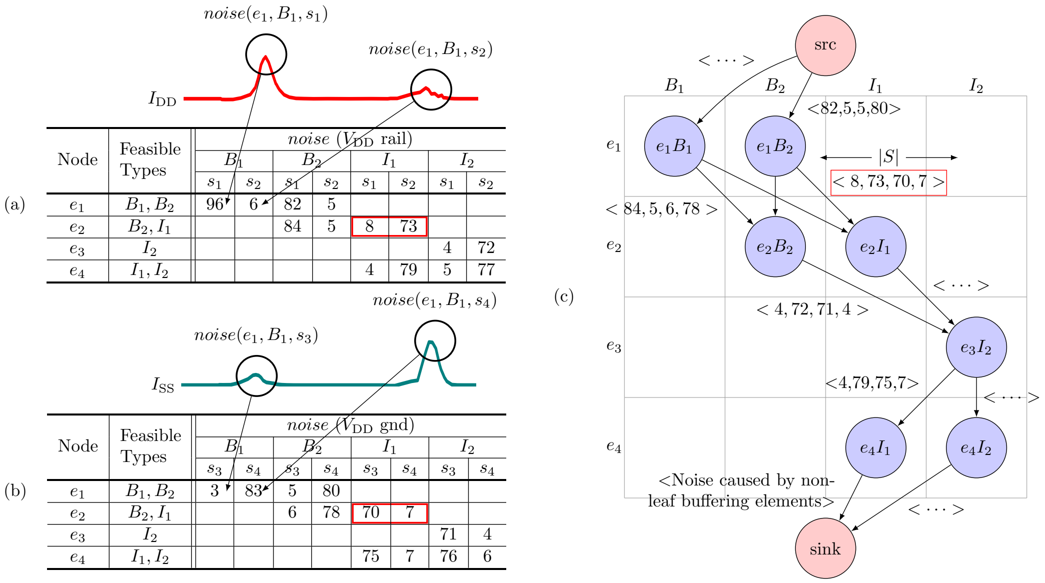
In addition to using Warburton's approximation algorithm we propose a fast version with lower time and space complexity of ClkWaveMin as shown as Algorithm 2. It performs the polarity assignment node by node basis iteratively, by selecting and assigning buffer/inverter with the least noise-worsening first from its current state. The memory of Algorithm 2 is $O(|S||L|)$ since there are $O(|L|)$ arcs in the MOSP graph and running time is $O(|S||L|^2)$.

5. EXPERIMENTAL RESULTS

The proposed algorithm ClkWaveMin has been implemented in C++ language on a Linux machine and tested on ISCAS'95 benchmark circuits. The benchmarks were synthesized using Synopsys' Design Compiler and clock trees were synthesized with IC Compiler, using Nangate 45nm Open Cell Library [15]. RC extractions were performed on IC Compiler and HSPICE simulation was done on the clock trees.

We also implemented the best ever known polarity assignment algorithm [10], which we call ClkPeakMin for the comparison with our ClkWaveMin. We profiled the noise waveform by simulating the clock tree multiple times while varying the buffer/inverter assignment to leaf nodes. All leaf nodes were attempted to be assigned to any of BUF_X8, BUF_X16, INV_X8, and INV_X16. For each assignment of a leaf node, I_{SS} and I_{DD} waveforms were probed with options to allow ASCII output with static sampling time interval, from which we can construct weight vectors of the same dimension. Clock signal arrival times for each case were also measured. The benchmark circuits were partitioned into a square grid of zones. On average, each zone contained 4.3 nodes. Benchmark design s35932 has 7.1 nodes in each zone on average.

Table 1 summarizes the comparison of the results produced by ClkPeakMin [10] and ClkWaveMin when clock



(a)

Node	Feasible Types	*noise* (V_{DD} rail)							
		B_1		B_2		I_1		I_2	
		s_1	s_2	s_1	s_2	s_1	s_2	s_1	s_2
e_1	B_1, B_2	96	6	82	5				
e_2	B_2, I_1			84	5	8	73		
e_3	I_2							4	72
e_4	I_1, I_2					4	79	5	77

(b)

Node	Feasible Types	*noise* (V_{DD} gnd)							
		B_1		B_2		I_1		I_2	
		s_3	s_4	s_3	s_4	s_3	s_4	s_3	s_4
e_1	B_1, B_2	3	83	5	80				
e_2	B_2, I_1			6	78	70	7		
e_3	I_2							71	4
e_4	I_1, I_2					75	7	76	6

Figure 3: An example instance of ClkWaveMin in one interval $[t_1 - \kappa, t_1]$ and its conversion to MOSP problem. For each fixed time interval, the feasibility of buffer and inverter types for each node can be calculated and the corresponding *noise* values for each noise slot can be determined. In this example, there are four slots $s_1, \cdots, s_4$, where s_1, s_2 are sampling slots for I_{DD} noise waveform and s_3, s_4 are for I_{SS}. The MOSP graph has vertices with "row" and "column" properties. The vertex located at row e_2 column I_1 corresponds to the option of assigning node e_2 with inverter type I_1. In the graph, there is no vertex of e_1I_1 because I_1 is not a feasible type for e_1 in given ClkWaveMin instance. A vertex in row i has incoming arcs from all the vertices in row $i-1$. For an arc (u,v) where v is at row r column c, the arc weight is defined as $w(u,v) = (noise(r,c,s_1), \cdots, noise(v,c,s_{|S|}))$. For example, any arc which is directed to vertex e_2I_1 has arc weight of $w(\cdot, e_2I_1) = (noise(e_2,I_1,s_1), noise(e_2,I_1,s_2), noise(e_2,I_1,s_3), noise(e_2,I_1,s_4)) = (8,73,70,7)$. One exception is the sink vertex that does not belong to any column. For the arcs that are directed to the sink vertex, the arc weights are assigned to reflect the noise caused by non-leaf clock tree buffering elements.

Table 1: Comparison of results by ClkPeakMin [10] and ClkWaveMin when $\kappa = 20$ps, $\epsilon = 0.01$, $|S| = 158$.

Benchmark Circuit	ClkPeakMin [10]			ClkWaveMin			Improvement		
	V_{DD} noise (mV)	Gnd noise (mV)	Peak curr. (mA)	V_{DD} noise (mV)	Gnd noise (mV)	Peak curr. (mA)	V_{DD} noise (%)	Gnd noise (%)	Peak curr. (%)
s13207	3.8	6.1	6.5	4.2	6.6	7.2	-9.2	-9.2	-12.4
s15850	1.5	2.0	3.0	1.5	2.0	3.0	0.0	0.0	0.0
s35932	7.5	8.7	21.6	4.4	8.4	15.6	41.7	3.0	27.8
s38417	7.8	7.5	19.8	3.8	7.0	11.9	51.5	6.8	40.1
s38584	5.4	6.5	16.9	4.3	7.9	11.6	20.0	-21.4	31.6
Average							**20.8**	**-4.2**	**17.4**

Table 2: Comparison with ClkWaveMin ($\epsilon = 0.01$) varying the number of time points and fast ClkWaveMin ($\kappa = 20$ps).

Benchmark circuit	Fast ClkWaveMin		$\|S\| = 4$		$\|S\| = 8$		$\|S\| = 158$	
	Peak curr. (mA)	Exec. time (ms)	Peak curr. (mA)	Exec. time (ms)	Peak curr. (mA)	Exec. time (ms)	Peak curr. (mA)	Exec. time (ms)
s13207	7.25	<0.01	7.2	<0.01	7.2	<0.01	7.2	<0.01
s15850	3.01	<0.01	3.0	<0.01	3.0	<0.01	3.0	<0.01
s35932	18.8	0.01	16.9	0.19	15.6	1.07	15.6	1.02
s38417	11.4	<0.01	13.0	0.08	11.9	0.51	11.9	0.49
s38584	10.3	0.01	13.6	1.02	11.6	0.7	11.6	0.66

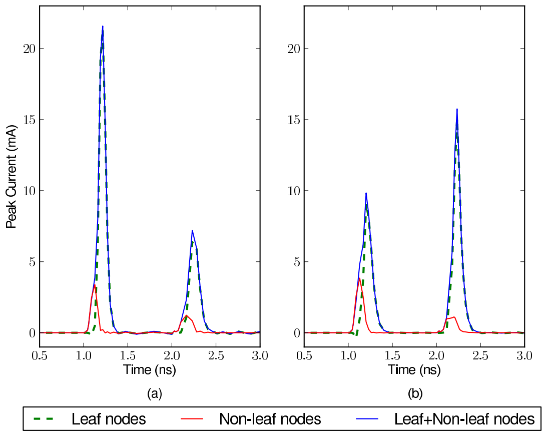


Figure 4: I_{SS} current waveforms for s35932. (a) Result by ClkPeakMin [10]. (b) Result of ClkWaveMin.

skew bound is set to $\kappa = 20$ps. In summary, ClkWaveMin reduces the power noise and peak current by 20.8% and 17.4% on average, respectively. However, ClkPeakMin sometimes outperforms ClkWaveMin; by examining the noise profiles that ClkWaveMin relies on, it is found that I_{SS} current that affects Gnd line does not have the clear peaks as I_{DD} does, which can cause ClkWaveMin to focus on V_{DD} line rather than Gnd line.

Table 2 shows comparison with results by ClkWaveMin using various time sampling points and our fast ClkWaveMin ($|S| = 158$). For $|S| = 4$, from I_{SS} and I_{DD} waveform, two values from each current profile were extracted by extracting the maximum value from the first and the second half of the waveform. We can see that the use of more sampling points leads to a further reduction in peak current. Further, our fast greedy algorithm produces result close to that by ClkWaveMin with 158 sampling points, but run time is significantly fast. For the five benchmark circuits, our algorithm found near-optimal assignments which were expected to have lower noise than the greedy algorithm. But the results from more accurate HSPICE simulation show that the greedy algorithm sometimes yields superior polarity assignments than ClkWaveMin. This is mainly caused by the modeling inconsistency between HSPICE and our noise model: $noise(\phi(e_i), s)$ is affected by the polarity/sizing of neighboring nodes.

Figs. 4(a) and (b) shows the I_{SS} current waveform produced by ClkPeakMin [10] and ClkWaveMin ($|S|$=158) for design s35932 in Table 2, respectively. Since ClkPeakMin is unaware of the noise from the non-leaf nodes, the produced waveform exhibits a large peak current at the rising edge of the clock signal while ClkWaveMin successfully migrates some noise from the rising edge of the clock to the falling edge so that the resultant noise is dispersed across time slots.

6. CONCLUSION

We address a new problem of clock buffer polarity assignment combined buffer sizing to overcome two critical limitations of the previous clock polarity assignment methods, which are the unawareness of the signal delay (i.e., arrival time) differences to the leaf buffering elements and the ignorance of the effect of the current fluctuation of non-leaf buffering elements on the total peak current waveform. We showed that the two limitations may cause a severe inaccuracy on the peak current estimation, resulting in unnecessarily high peak current. To overcome the limitations, we proposed a completely new (fine-grained) approach to the clock buffer polarity assignment combined with buffer sizing. We formulated the problem into multi-objective shortest path problem, and proposed fast as well as approximation algorithms to solve the problem.

Acknowledgement: This work was supported by Basic Science Research Program through National Research Foundation (NRF) grant funded by the Korea Ministry of Education, Science and Technology (No. 2009-0091236).

7. REFERENCES

[1] S. Chowdhury and J. Barkatullah, “Estimation of maximum currents in MOS IC logic circuits,” *IEEE TCAD*, vol. 9, no. 6, 1990.

[2] L. Chen, M. Marek-Sadowska, and F. Brewer, “Buffer delay change in the presence of power and ground noise,” *IEEE TVLSI*, vol. 11, no. 3, 2003.

[3] K. Tang and E. Friedman, “Simultaneous switching noise in on-chip CMOS power distribution networks,” *IEEE TVLSI*, vol. 10, no. 4, 2002.

[4] N. H. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*, 3rd ed. USA: Addison-Wesley Publishing Company, 2005.

[5] Y.-T. Nieh, S.-H. Huang, and S.-Y. Hsu, “Minimizing peak current via opposite-phase clock tree,” in *DAC*, 2005.

[6] R. Samanta, G. Venkataraman, and J. Hu, “Clock buffer polarity assignment for power noise reduction,” in *ICCAD*, 2006.

[7] P.-Y. Chen, K.-H. Ho, and T. Hwang, “Skew-aware polarity assignment in clock tree,” *TODAES*, vol. 14, no. 2, 2009.

[8] Y. Ryu and T. Kim, “Clock buffer polarity assignment combined with clock tree generation for power/ground noise minimization,” in *ICCAD*, 2008.

[9] M. Kang and T. Kim, “Clock buffer polarity assignment considering the effect of delay variations,” in *ISQED*, 2010.

[10] H. Jang, D. Joo, and T. Kim, “Buffer sizing and polarity assignment in clock tree synthesis for power/ground noise minimization,” *IEEE TCAD*, vol. 30, no. 1, 2011.

[11] J. Lu and B. Taskin, “Clock buffer polarity assignment considering capacitive load,” in *ISQED*, 2010.

[12] Z. Tarapata, “Selected multicriteria shortest path problems: An analysis of complexity, models and adaptation of standard algorithms,” *Int. J. Appl. Math. Comput. Sci.*, vol. 17, pp. 269–287, June 2007.

[13] A. Warburton, “Approximation of pareto optima in multiple-objective, shortest-path problems,” *Oper. Res.*, vol. 35, pp. 70–79, February 1987.

[14] M. Ehrgott, *Multicriteria Optimization*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.

[15] “Open cell library v2009 07, 2009,” http://www.nangate.com/openlibrary, 2009.