Aging-Aware Timing Analysis and Optimization Considering Path Sensitization*

Kai-Chiang Wu and Diana Marculescu
Department of Electrical and Computer Engineering
Carnegie Mellon University
{kaichiaw, dianam}@ece.cmu.edu

ABSTRACT

Device aging, which causes significant loss on circuit performance and lifetime, has been a main factor in reliability degradation of nanoscale designs. Aggressive technology scaling trends, such as thinner gate oxide without proportional downscaling of supply voltage, necessitate an aging-aware analysis and optimization flow during early design stages. Since only a small portion of critical and near-critical paths can be sensitized and may determine the circuit delay under aging, path sensitization should also be explicitly addressed for more accurate and efficient optimization. In this paper, we first investigate the impact of path sensitization on aging-aware timing analysis and then present a novel framework for aging-aware timing optimization considering path sensitization. By extracting and manipulating critical sub-circuits accounting for the effective circuit delay, our proposed framework can reduce aging-induced performance degradation to only ***1.21%*** *or one-seventh of the original performance loss with less than* ***2%*** *area overhead.*

1. INTRODUCTION

As discussed in the 2009 International Technology Roadmap for Semiconductor [1], the long-term reliability of nanometer integrated circuits can reach a noteworthy order of 10^3 FITs (failures in 10^9 hours). Some of the main challenges driving the design of reliable systems (*i.e.*, design for reliability) encompass soft errors, process variations, and device aging phenomena. With the continuous scaling of transistor dimensions, device aging, which causes significant loss on circuit performance and lifetime, is becoming increasingly dominant for temporal reliability concerns. Therefore, an early-stage design optimization flow considering aging effects is necessitated as a key factor in guaranteeing consistently reliable operation over a desired lifespan.

Among various aging mechanisms, *negative bias temperature instability* (NBTI) [2] is known for being particularly crucial due to current scaling trends such as shrinking thickness of gate oxide. NBTI is a PMOS aging phenomenon that occurs when PMOS transistors are stressed under negative bias ($V_{gs} = -V_{dd}$) at elevated temperature. As a result of the dissociation of *Si-H* bonds along the *Si-SiO*$_2$ interface, NBTI-induced PMOS aging manifests itself as an increase in the threshold voltage and decrease in the drive current [3], which in turn slow down the rising propagation delay of logic gates. Experiments on PMOS aging [4] indicate that NBTI effects grow exponentially with thinner gate oxide and higher operating temperature, which are the expected trends of technology scaling. If the thickness of gate oxide shrinks down to 4nm, the circuit performance can be degraded by as much as 15% after 10 years of stress and lifetime will be dominated by NBTI [5]. In contrast, the aging mechanism can be partially recovered when the stress condition is relaxed ($V_{gs} = 0$).

In addition to the oxide thickness and operating temperature, NBTI-induced performance degradation strongly depends on the amount of time during which a PMOS transistor is stressed. In [6][7], the increase in threshold voltage has been shown to be a logarithmic function of the corresponding stress time. This parameter, depending on the circuit topology and input vectors, is distributed non-uniformly from transistor to transistor. The asymmetric distribution may lead to 2-5X difference in the degradation rate of threshold voltage [7].

When dealing with the problem of aging-induced performance degradation, it is important to consider path sensitization because (*i*) only a small portion of long paths can determine the delay of a circuit no matter whether aging applies, and (*ii*) a path that is not critical/sensitizable before aging may become critical/sensitizable after aging and affect circuit performance. A path is sensitizable if it can be activated by at least one combination of input transitions. In this paper, by using *timed automatic test pattern generation* [8], we examine the impact of path sensitization on aging-aware timing analysis and also explore the benefits of considering path sensitization for aging-aware timing optimization.

The rest of this paper is organized as follows: Section 2 gives an overview of related work and outlines the contribution of our paper. In Section 3, we review the concept of path sensitization and discuss its impact on aging-aware timing analysis. Section 4 presents an efficient methodology for aging-aware timing optimization considering path sensitization. In Section 5, the experimental results for a set of standard benchmarks are demonstrated. Finally, we conclude our work in Section 6.

2. RELATED WORK AND PAPER CONTRIBUTION

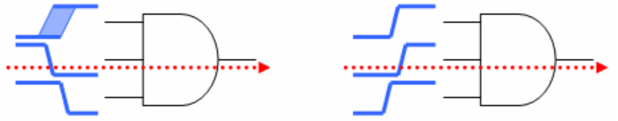
2.1. Previous Work on Aging-Aware Timing Optimization

Traditional design methods add guard-bands or adopt worst-case margins to account for aging phenomena, which in practice refer to over-design and may be expensive. To avoid overly conservative design, the mitigation of aging-induced performance degradation can be formulated as a timing-constrained area minimization problem with consideration of aging effects. Recent aging-aware techniques basically follow this formulation. The authors of [9] proposed a gate sizing algorithm based on Lagrangian relaxation. An average of 8.7% area penalty is required to ensure reliable operation for 10 years. Other methods related to gate or transistor sizing can be found in [10][11].

A novel technology mapper considering signal probabilities for NBTI was developed in [12]. This technique takes signal probabilities as one of the arguments when searching for the best match in a given standard cell library. On average 10% area recovery and 12% power saving are accomplished, as compared to the most pessimistic case assuming static NBTI on all PMOS transistors in the design. A framework using joint logic restructuring and pin reordering [13] can mitigate NBTI-induced performance degradation with no gate area overhead, while decreasing the number of critical transistors under severe NBTI. In [14], a

* This research was supported in part by NSF Grant CNS-07020653.

978-3-9810801-7-9/DATE11/©2011 EDAA



(a) Earliest controlling transition on the middle pin

(b) Latest non-controlling transition on the middle pin given that all transitions are non-controlling

Figure 1. Criteria of path sensitization

reconfigurable flip-flop design based on time borrowing is introduced for aging detection and correction. Among all of the aforementioned approaches, only the one in [11] considers path sensitization for more accurate optimization. However, the approach involves path enumeration (on a path-wise basis) of exponential complexity and is not scalable for large benchmarks.

Instead of reducing NBTI effects during active mode as described above, an idea of NBTI-aware optimization during standby mode was presented in [15]. Input vectors for minimum standby-mode leakage are selected to minimize PMOS aging. Moreover, for gates that are deep in a large circuit and cannot be well controlled by primary input vectors, internal node control [16] intrusively assigns logic "1" to those gates if they are on the critical paths. The logic "1" relaxes the stress condition and can thus relieve the NBTI impact.

2.2. Paper Contribution

By employing *timed automatic test pattern generation* (timed ATPG) [8] for timing analysis considering path sensitization, we propose a methodology to identify critical gates that are truly necessary to be manipulated for aging-aware timing optimization. Timed ATPG, based on the satisfiability (SAT) problem, is used to generate input patterns activating critical paths. In this way we can efficiently identify those gates along the activated critical paths as critical gates. A small subset of critical gates is finally selected as candidate for aging-aware timing optimization, and more importantly, with path sensitization explicitly addressed. The contributions and advantages of our methodology are threefold:

- ***Runtime efficiency and process variability awareness*:** Unlike existing work [11] which formulates the gate sizing problem on a path-wise basis, we directly identify gates to be resized without enumerating all possible paths. Hence, the proposed algorithm is efficient in terms of runtime and scalable for large-scale designs. Furthermore, our methodology involves iterative optimization and converges perfectly even in the presence of manufacturing process-driven variations.
- ***Low design penalty*:** The framework presented in [13] is implemented and integrated into our methodology. Due to the fact that (*i*) the joint approaches in [13] reduce the number of critical transistors with marginal design penalty and (*ii*) not all of the gates on critical paths require resizing when path sensitization is considered, the integrated framework incurs very little area overhead compared to existing techniques based on sizing alone or without considering path sensitization.

3. IMPACT OF PATH SENSITIZATION ON AGING-AWARE TIMING ANALYSIS

3.1. Sensitizable Paths vs. False Paths

A path is defined as a *sensitizable path* if there is at least one primary input vector activating the path. From the timing perspective, a sensitizable path can propagate a transition (rising or falling) to at least one primary output, which may determine the delay of a circuit. Figure 1 shows two conditions of path sensitization for a 3-input AND gate. As indicated by red dotted lines, a path to be sensitized must hold either the earlier controlling transition (*i.e.*, falling transition for an AND gate, see Figure 1(a)) or the latest non-controlling (rising) transition if all input transitions are non-controlling (see Figure 1(b)). In contrast, a path that is not sensitizable is called a *false path* whose delay, however, cannot affect the circuit performance.

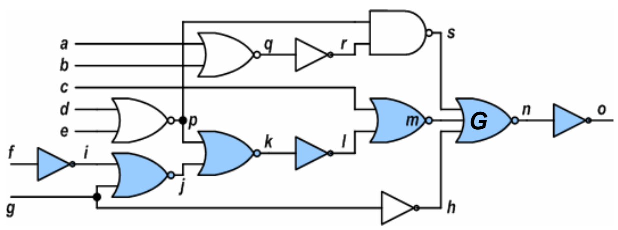


Figure 2. A longest topological path that is false (un-sensitizable)

For example, in Figure 2, the highlighted gates depict the longest topological path (*f – i – j – k – l – m – n – o*). Since there does not exist a combination of input transitions activating the path, the longest topological path is a false path and will not determine the delay of the circuit. Note that, in this case, no other path except the highlighted false path passes through the gates feeding wires *i* and *j*; in other words, they do not lie on any sensitizable path. Therefore, any amount of aging-induced delay increase at these two gates will never reflect performance degradation on the circuit. Speeding up these gates is of no benefit in terms of circuit performance. In order for more accurate and efficient optimization, the basic principle of our methodology is to extract and manipulate the sub-circuit covering only sensitizable paths which are critical or near-critical. The effective circuit delay (*i.e.*, the delay of the longest sensitizable path) can be minimized by focusing on optimizing the sub-circuit and disregarding anything else beyond it.

As reported in [17], less than **10%** of long (critical and near-critical) paths should be selected for performance optimization if false paths are excluded. Shortening the small portion of long paths, *e.g.*, by speeding up some gates covered, can simply reduce the effective circuit delay, and those long paths that are false can be left un-optimized without affecting the overall circuit performance.

3.2. Aging-Aware Timing Analysis Considering Path Sensitization

Before discussing the impact of path sensitization on aging-aware timing analysis, we briefly introduce the NBTI modeling and analysis framework [7][18] used in our paper. This framework enables us to analyze the long-term behavior of NBTI-induced PMOS degradation. First, the degradation of threshold voltage at a given time *t* can be predicted as:

$$\Delta V_{th} = \left(\frac{\sqrt{K_v^2 \cdot T_{clk} \cdot \alpha}}{1 - \beta_t^{1/2n}} \right)^{2n} \qquad \textbf{(1)}$$

where K_v is a function of temperature, electrical field, and carrier concentration, α is the stress probability, and n is the time exponential constant, 0.16 for the used technology. The detailed explanation of each parameter can be found in [7].

Next, the authors of [18] simplify this predictive model to be:

$$\Delta V_{th} = b \cdot \alpha^n \cdot t^n = b \cdot (\alpha \cdot t)^n \qquad \textbf{(2)}$$

where $b = 3.9\times10^{-3}$ V·s$^{-1/6}$.

Table 1. Aging-aware timing analysis
with and without path sensitization considered

Circuit	Without considering path sensitization (*i.e.* , delays of longest topological paths)			With path sensitization considered (*i.e.* , delays of longest sensitizable paths)		
	No aging (fresh)	10-year aging	%	No aging (fresh)	10-year aging	%
alu2	1128	1237	9.66%	1092	1184	8.42%
alu4	1430	1560	9.09%	1343	1452	8.12%
C1908	1469	1621	10.35%	1366	1502	9.96%
C3540	1850	2022	9.30%	1781	1944	9.15%
C5315	1373	1497	9.03%	1357	1465	7.96%
C7552	1582	1731	9.42%	1561	1717	9.99%
s1238	846	931	10.05%	836	908	8.61%
s9234	1134	1240	9.35%	1071	1151	7.47%
	(ps)	(ps)		(ps)	(ps)	

Finally, the rising propagation delay of a gate through the degraded PMOS can be derived as a first-order approximation:

$$\tau'_p = \tau_p + a \cdot (\alpha \cdot t)^n \qquad \textbf{(3)}$$

where ι_p is the intrinsic delay of the gate without NBTI degradation and a is a constant.

In the remaining of this paper, we apply Equation (3) to calculate the delay of each gate under NBTI, and further estimate the performance of a circuit. The coefficient a in Equation (3) for each gate type and each input pin is extracted by fitting HSPICE simulation results in 70nm, Predictive Technology Model (PTM). The simplified long-term model successfully predicts the PMOS degradation with negligible error.

We exploit the NBTI prediction model on top of *timed automatic test pattern generation* (timed ATPG) [8] to analyze the effective delay of a circuit while accounting for both aging awareness and path sensitization. Timed ATPG itself was presented as a false-path-aware timing analyzer. Given a timing specification (T_{spec}) for a target circuit, the timed ATPG algorithm will construct a corresponding *timed characteristic function* (TCF) in *conjunctive normal form* (CNF). The TCF characterizes the timing behavior of the circuit as a Boolean equation and its on-set specifies input vectors that, when evaluated, can propagate transitions stabilizing later than or equal to T_{spec} at any of the outputs (*i.e.*, with propagation delays greater than or equal to T_{spec}). Because of the CNF (product-of-sum) representation, existing SAT solvers are used to derive one set of input patterns if the TCF is satisfiable; otherwise solvers return nothing, meaning that no such input vector exists to activate a path with delay greater than or equal to T_{spec}. By actually applying the derived input vector to the circuit, the corresponding sensitizable path(s) can be traced. Then, we can identify critical and near-critical sensitizable paths if a timing specification smaller than (but close to) the delay of the longest topological path is chosen.

One major concern for the unified treatment of aging awareness and path sensitization is that, due to the asymmetric rate of aging, a path which is not critical/sensitizable at the beginning of lifetime may become critical/sensitizable and affect circuit performance during the lifetime span (or vice versa). Thanks to the support of timed ATPG, we just need to plug the aging model such that timed ATPG can calculate the change in each pin-to-pin delay based on manufacturing and operating parameters. To obtain the effective delay of a circuit, we use the same stepping method as that in [8] which adjusts T_{spec} dynamically. The maximum T_{spec} achieved for constructing a satisfiable TCF is the effective circuit delay. Table 1 demonstrates the results of aging-aware timing analysis for standard benchmarks whose effective delays are not determined by longest topological paths. We list in the table the values of fresh circuit delay (at time 0) and aged delay under a generic stress condition of 10 years. For circuit *alu2*, the difference in fresh delay between the longest topological path (column 2) and the longest sensitizable path (column 5) is 36ps, while that in 10-year aged delay (columns 3 and 6) is 53ps. As it can be seen, the difference increases (except circuit *C7552*) as a result of aging. Moreover, the percentage of aging-induced performance degradation decreases if path sensitization is taken into account. For more accurate timing analysis and to avoid underestimation of circuit lifetime, it is necessary to consider path sensitization when aging effects are getting severe.

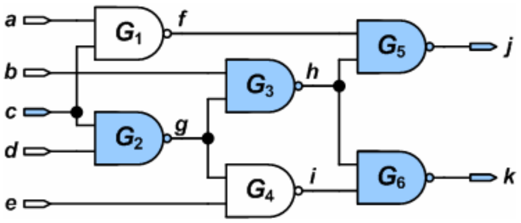


Figure 3. An example circuit (*C17*) for illustrating our methodology

4. PROPOSED METHODOLOGY FOR AGING-AWARE TIMING OPTIMIZATION

The objective of our methodology is to minimize the circuit delay under 10-year NBTI by incurring as little area overhead as possible, while taking into account and taking advantage of the impact of path sensitization. The pre-processing task iteratively performs logic restructuring and pin reordering [13], with minimum area penalty, until no more improvement can be made. As the main procedure, *transistor resizing* is integrated with [13] for further mitigation of NBTI-induced performance degradation, with low area overhead. From the discussion in Section 3.1, it is evident that considering path sensitization can reduce the overall design penalty for timing optimization. Efficient identification of candidates to be manipulated (including gates, transistors, and wires) becomes a more challenging issue. In the sequel, we present an efficient approach to identifying the critical sub-circuit, which consists of potential candidates, for explicit consideration of path sensitization during aging-aware timing optimization.

4.1. Efficient Identification of Critical Sub-Circuits Considering Path Sensitization

We use benchmark circuit *C17* (see Figure 3) from the ISCAS'85 suite to explain the key idea of our proposed methodology based on timed ATPG. Note that the timed ATPG algorithm presented in [8] adopts floating-mode operation where a transition of a node is defined as a switch of its state from an unknown value to a known value. Without loss of generality, we assume that wires do not contribute to the circuit delay and the delay of each node is its intrinsic delay plus the fanout delay (unit fanout delay model). The intrinsic delay of an internal gate is 1, while that of a primary input is 0. The fanout delay is calculated as 0.2X the number of fanout neighbors. The assumption can be easily relaxed for a non-uniform delay model, with wire loads considered.

Under unit fanout delay model, there are two longest topological paths in *C17* (*i.e.*, $c - G_2 - G_3 - G_5 - j$ and $c - G_2 - G_3 - G_6 - k$, as highlighted) with delays of 4.2 (= 0.4+1.4+1.4+1.0). By choosing T_{spec} = 4.2, the on-set of the corresponding TCF specifies input vectors activating these two paths since they are both sensitizable. However, a typical SAT solver derives "one" input vector satisfying the TCF at a time and may not enumerate all satisfying vectors to activate all possible sensitizable paths. To extract the sub-circuit covering all sensitizable paths with delays greater than or equal to T_{spec}, we modify the TCF by adding new clauses into its CNF such that a SAT solver, if used repeatedly, can generate different input vectors and identify possible sensitizable paths in an efficient manner.

Let F be the TCF for *C17* given T_{spec} = 4.2 and CNF(F) be the CNF representation of F. Clearly, CNF(F) is satisfiable due to the existence of two sensitizable paths whose delays are 4.2. By running a SAT solver on CNF(F), we obtain a set of satisfying input patterns $\{a, b, c, d, e\} = \{0, 1, 0, 1, 1\}$, which evaluates F to a "1". The set of input patterns, when actually applied to *C17*, can activate the critical path along $c - G_2 - G_3 - G_5 - j$. Without modifying CNF(F) or the implementation of the SAT solver, it is not possible to obtain a different set of input patterns activating the other critical path along $c - G_2 - G_3 - G_6 - k$. As a naïve solution, we can append a clause $(a + \neg b + c + \neg d + \neg e)$ to CNF(F) so the new CNF, denoted by CNF'(F), is CNF(F) × $(a + \neg b + c + \neg d + \neg e)$. Intuitively, the same vector $\{a, b, c, d, e\} = \{0, 1, 0, 1, 1\}$ evaluates the new clause to a "0", making CNF'(F) unsatisfied. Therefore, the SAT solver will find a different input vector which may or may not activate the other critical path. One may note that the complexity of this naïve strategy grows exponentially with the number of primary inputs. The exponential complexity implies clause explosion of the CNF and an intractable approach with huge runtime for running SAT solvers. To reduce the complexity to a feasible extent, we introduce the following theorem to modify CNF(F). The goal is to find a minimum set of new clauses that, when added one by one, will make CNF(F) un-satisfiable, which means that we can gradually identify critical and near-critical sensitizable paths given a T_{spec} and extract the critical sub-circuit.

Definition 1 (***side input***)**:** For each gate on an activated path, a *side input* is an input pin of the gate through which the activated path does not pass.

Definition 2 (***side-input assignment***)**:** For each side input, its value assignment, called *side-input assignment*, is the value evaluated by propagating a particular input vector.

Theorem: For each activated path with side-input assignments $\{x_p, \ldots, x_q, \ldots, x_r\} = \{v_p, \ldots, v_q, \ldots, v_r\}$, a new clause

$$\sum_{i\in p\ldots q\ldots r}(x_i \oplus v_i) = \big((x_p \oplus v_p) + \ldots + (x_q \oplus v_q) + \ldots + (x_r \oplus v_r)\big)$$

can be added into CNF(F) such that different input vectors will be derived for activating critical and near-critical paths which have not been identified yet.

Proof: Omitted due to space limitations. ■

Consider path $c - G_2 - G_3 - G_5 - j$ activated by input vector $\{a, b, c, d, e\} = \{0, 1, 0, 1, 1\}$. By propagating the input vector, the side input assignments for this activated path are $\{b, d, f\} = \{1, 1, 1\}$. According to the theorem, the new clause to be added is $((b \oplus 1) + (d \oplus 1) + (f \oplus 1)) = (\neg b + \neg d + \neg f)$. After adding $(\neg b + \neg d + \neg f)$ into CNF(F) (CNF'(F) = CNF(F) × $(\neg b + \neg d + \neg f)$), input vectors which evaluate b to a "1", d to a "1", **and** f to a "1" cannot satisfy CNF'(F) and thus will not be generated. The next set of input patterns derived by the solver will be $\{a, b, c, d, e\} = \{1, 1, 1, 1, 0\}$, which activates the other critical path along $c - G_2 - G_3 - G_6 - k$ whose side-input assignments are $\{b, d, i\} = \{1, 1, 1\}$. Finally, by adding the corresponding clause, $(\neg b + \neg d + \neg i)$, the resulting CNF of F becomes un-satisfiable, meaning that all critical sensitizable paths have been identified. Note that a single input vector may activate several paths and for each activated path, a new clause should be added. For example, input vector $\{a, b, c, d, e\} = \{0, 1, 0, 1, 0\}$ can activate the two critical sensitizable paths in *C17* simultaneously.

Compared to the naïve approach of exponential complexity, the proposed methodology significantly decreases the number of added clauses and the number of SAT runs. In the case of *C17* under unit fanout delay model, only two additional clauses are added and three runs of the SAT solver are needed. Hence, we can efficiently extract the sub-circuit consisting of critical and near-critical sensitizable paths. The extracted sub-circuit, called *critical sub-circuit*, is the main focus of our integrated framework using logic restructuring, pin reordering, and transistor resizing for aging-aware timing optimization. Anything beyond the critical sub-circuit is either non-critical or un-sensitizable. On these non-critical/un-sensitizable portions, timing optimization may not be effective and consequently, they can be excluded for lowering the design penalty.

Let us use the circuit in Figure 2 to summarize our methodology for aging-aware timing optimization. Assuming unit fanout delay model, the delay of the longest topological path ($f - i - j - k - l - m - n - o$) in the circuit is 8.4 (= 0.2+1.2*6+1.0). As mentioned, it is a false path and will not be identified as part of the critical sub-circuit given T_{spec} = 8.4. The delay of the circuit is determined by two longest sensitizable paths from d and e, via $k - l - m - n$, to o with delays of 7.4 (= 0.2+1.4+1.2*4+1.0). By choosing T_{spec} = 7.4, the critical sub-circuit consisting of these two paths can be extracted to be manipulated by logic restructuring, pin reordering, and transistor resizing. For logic restructuring [13], we will swap c and p, instead of c and j as path sensitization is not considered. Here, wires c, j, and p are functionally symmetric and any two of them can be swapped with each other while maintaining the circuit functionality. For pin reordering, we may change the input order of gate G (wires h, m, and s) to minimize the circuit delay under aging. For transistor resizing, we apply a similar algorithm to that in [19] on the critical sub-circuit and will not touch the transistors connected to wires f and i that are on the longest topological (but a false) path.

4.2. Guaranteeing Full Coverage of Sensitizable Paths

Up to this point, the efficient methodology for critical sub-circuit extraction does not guarantee to identify all critical and near-critical sensitizable paths. In fact, identifying all sensitizable paths given a T_{spec} is not necessary for our concern of extracting the critical sub-circuit as long as the extracted sub-circuit has covered all of them already. This is usually the case because a large fraction of those paths overlap and share many segments. In a few cases, missing sensitizable paths may lead to incomplete extraction of critical sub-circuits. Figure 4 shows a case where a sensitizable path may be missed. In this example, input vector V_1 activates paths P_1 and P_2 while V_2 activates P_2 and P_3. Note that P_2 can be activated by both V_1 and V_2. Suppose V_1 is generated by the SAT solver based on timed ATPG, P_1 and P_2 will be activated and their corresponding clauses C_1 and C_2 will be added into the CNF. However, after C_2 has been added, V_2 will no longer satisfy the new CNF and thus, P_3 will not be identified – a miss.

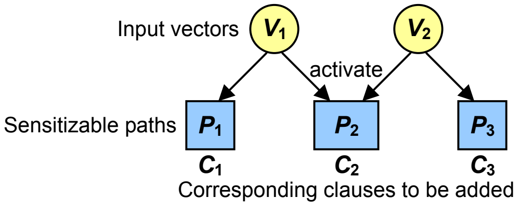


Figure 4. A case of missing sensitizable paths

To deal with this issue, we apply the same T_{spec} on timed ATPG repeatedly until we extract all possible critical sub-circuits and optimize them. That is to say, if there are indeed some missed sensitizable paths for a given T_{spec}, we use timed ATPG with the same T_{spec} for another run of critical sub-circuit extraction. Due to the fact that the number of unidentified sensitizable paths decreases drastically after each run of extraction and optimization, this strategy for guaranteeing full coverage of sensitizable paths works well and will not impose significant runtime overhead.

4.3. Proposed Algorithm Description

Our overall algorithm for aging-aware timing optimization, including all ideas presented in Sections 4.1-4.2, is given in Figure 5. As a pre-processing procedure, joint logic restructuring (LR) and pin reordering (PR), which introduce no gate area overhead, are performed to topologically shorten the circuit delay under aging without considering path sensitization for reduced computational complexity. Then, we iterate the proposed methodology based on timed ATPG with decreasing T_{spec} until a specified performance target is met or no further improvement can be made. In each iteration, transistor resizing, as well as joint LR and PR, are applied on the extracted critical sub-circuit to optimize the effective circuit delay, while explicitly considering path sensitization. Lines 16-17 are used for guaranteeing full coverage of sensitizable paths by not decreasing T_{spec} if there are still sensitizable paths identified during the current run of timed ATPG. The complexity of our algorithm is bounded by satisfiability-based ATPG, which is a known NP-complete problem but can be addressed efficiently by existing solvers using a wide combination of techniques. In the worst case, the algorithm is of exponential complexity. In reality, it is absolutely more scalable than other approaches based on path enumeration, whose average-case complexity is exponential.

4.4. Impact of Process Variability

By extracting the corresponding critical sub-circuit before performing each run of optimization, the proposed algorithm can exclude gates that do not need to be manipulated for lower optimization effort and design penalty. A gate must be excluded from the critical sub-circuit if it is on un-sensitizable or non-critical paths only, where "non-critical" paths are in contrast to "critical" and "near-critical" paths. In the presence of process variations, the fresh threshold voltage of each transistor (before aging, *i.e.*, at time 0) is no longer a fixed value but a random variable, which makes the problem of aging-aware timing optimization non-deterministic across silicon instances of a design. More precisely, the circuit delay may be different from one silicon instance to another because of different fresh threshold voltages, different behaviors of transistor aging, and different patterns of path sensitization. However, as indicated in [7], the impact of process variability can be compensated by the NBTI effect. Due to the compensation effect of device aging on process variability, a non-critical path will hardly dominate the circuit delay in the long term unless process variations incur a significant delay increase on the path. This is particularly uncommon when the focus is, as proposed, on the minimization of long-term (10-year) circuit performance. In addition, since our algorithm involves an *iterative* process of exploiting timed ATPG with decreasing T_{spec} to gradually reduce the effective circuit delay, a gate which is not covered by the critical sub-circuit in the previous run will be covered in the current run if it is now on a critical sensitizable path (but not previously). Hence, all potential candidate gates are guaranteed to be identified, earlier or later, for the purpose of aging-aware timing optimization considering path sensitization.

```
Input: circuit netlist, delay model, and performance target
Output: optimized circuit netlist
Algorithm: aging-aware timing optimization
01 Apply joint LR and PR without considering path sensitization
02 $D \leftarrow$ delay of the longest topological path, without aging applied
03 $D' \leftarrow$ delay of the longest topological path, with aging applied
04 $\Delta \leftarrow (D' - D) / n$ // $n$ is specified, the number of iterations, usually 10
05 $T_{spec} \leftarrow D' - \Delta$
06 DO {
07     $C \leftarrow \varnothing$ // critical sub-circuit, a set of "gates" instead of "paths"
08     $F \leftarrow$ construct TCF given $T_{spec}$
09     WHILE (CNF($F$) is satisfiable) {
10         $V \leftarrow$ derive a satisfying input vector
11         $P \leftarrow$ trace sensitizable path(s) by propagating $V$
12         $C \leftarrow C \cup$ (gates along $P$) // not on a path-wise basis
13         Add corresponding clause(s) into CNF($F$)
14     }
15     Apply transistor resizing and LR/PR on $C$ only
16     IF (no clause is added) THEN // for guaranteeing full coverage
17         $T_{spec} \leftarrow T_{spec} - \Delta$
18 } WHILE (performance target is met)
   // Also terminates if no improvement for consecutive 2 iterations.
```

Figure 5. The overall algorithm

5. EXPERIMENTAL RESULTS

We have implemented the integrated framework for aging-aware timing optimization considering path sensitization. Experiments are conducted on a subset of benchmarks from the ISCAS and MCNC suites. The technology used is 70nm, Predictive Technology Model (PTM). The supply voltage is 1.2V and the operating temperature is assumed to be 300K. Our framework aims at mitigating performance degradation under 10-year NBTI, with minimum design penalty. For each benchmark, logic simulation with 10000 random patterns, assuming that the 0-probabilities of all primary inputs are 0.5, is applied to calculate the probability of each signal. In the case of real applications with various workloads, we can apply different sets of input probabilities and use average signal probabilities instead. Given signal probability α of the input to a PMOS, the 10-year threshold degradation of the PMOS can be predicted by Equation (2). For each gate type and input pin (PMOS), HSPICE simulations with its nominal and degraded threshold voltages are performed for a discrete set of signal probabilities from 0 to 1. We fit these HSPICE results to obtain coefficients a's in Equation (3). Therefore, the gate delay and circuit timing under NBTI can be accurately estimated.

Table 2 reports the experimental results of our proposed methodology for aging-aware timing optimization. All baseline circuits, listed in column one, are pre-optimized and mapped in terms of delay, and their nominal delays (without consideration of aging effects) are shown in column two. Columns three and four show the circuit delays under aging and percentages of degradation compared to the nominal cases. Columns five and six demonstrate the improved delays and corresponding percentages after

Table 2. Aging-aware timing optimization with path sensitization considered

Circuit	No aging (fresh)	10-year aging	%	LR+PR	%	TR & LR+PR	%	Area over-head	Run-time
alu2	1092	1184	8.42%	1127	3.20%	1086	-0.52%	2.35%	19s
alu4	1343	1452	8.12%	1391	3.57%	1356	0.98%	2.52%	28s
C1355	921	994	7.93%	964	4.71%	924	0.32%	4.60%	1m2s
C1908	1366	1502	9.96%	1436	5.11%	1385	1.41%	0.93%	32s
C3540	1781	1944	9.15%	1857	4.25%	1807	1.43%	2.59%	2m5s
C5315	1357	1465	7.96%	1400	3.18%	1359	0.17%	0.68%	2m39s
C7552	1561	1717	9.99%	1649	5.66%	1602	2.62%	3.83%	15m3s
s713	876	944	7.76%	895	2.18%	872	-0.50%	1.19%	8s
s832	564	607	7.62%	590	4.58%	582	3.15%	0.62%	9s
s1196	748	812	8.56%	785	4.98%	768	2.63%	2.12%	40s
s1238	836	908	8.61%	872	4.36%	849	1.50%	1.03%	37s
s9234	1071	1151	7.47%	1101	2.77%	1085	1.32%	0.65%	58s
AVG.	(ps)	(ps)	**8.46%**	(ps)	**4.05%**	(ps)	**1.21%**	**1.93%**	
			1.00		**0.48**		**0.14**		

the pre-processing procedure using joint logic restructuring (LR) and pin reordering (PR). Columns seven and eight demonstrate those after the integrated framework using transistor resizing as well as joint LR and PR. Columns nine and ten show the area overheads and runtimes. The runtimes, including the times spent on logic simulation and the whole algorithm in Figure 5, are measured on a 3GHz Pentium 4 workstation running Linux. Every delay number in Table 2 is found with path sensitization considered, *i.e.*, the delay of the longest sensitizable path in a circuit (denoted by D), which is the maximum T_{spec} achieved for constructing a satisfiable TCF in timed ATPG. Any T_{spec} greater than D fails to derive a satisfiable TCF after our algorithm finishes, meaning that no path with delay greater than D can be sensitized and D determines the circuit performance accordingly.

For example, the nominal delay of circuit *alu2* is 1,092ps and the delay considering 10-year NBTI effects is 1,184ps, which means 8.42% performance degradation. The pre-processing LR and PR can reduce the circuit delay to 1,127ps (3.20% degradation). After being optimized by the proposed methodology as shown in Figure 5, the circuit delay becomes 1,086ps and we can even achieve a performance improvement of 0.52% while incurring 2.35% area overhead. On average across all listed benchmarks, aging-induced performance degradation can be recovered to 1.21%, which is about only one-seventh of the un-optimized case, with less than 2% area overhead. When compared to existing sizing techniques accounting for aging, our methodology is not only more cost-efficient than [9][10], which do not address path sensitization, but also more runtime-efficient than [11], which addresses path sensitization on a path-wise basis. The runtimes for the proposed framework range from <10 seconds to 15 minutes, as opposed to [11] whose largest ISCAS benchmark that can be handled is *C880*.

Figure 6 depicts the incremental recovery of aging-induced performance degradation by our iterative optimization algorithm. For circuit *C1908* (*C5315*), it takes six (seven) iterations of joint LR and PR to reduce performance degradation to 5.11% (3.18%) and takes another five (four) iterations (Lines 6-18 in Figure 5) to reach 1.41% (0.17%). We employ the same perturbation techniques as those in [13][19] to prevent the algorithm from being trapped in the local optimum. The effect of perturbation has been included in the results even though Figure 6 exhibits monotonic decreases in the overall degradation because we keep track of only the best solution during each iteration.

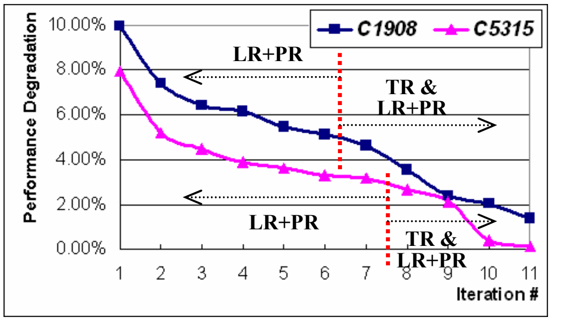


Figure 6. Incremental recovery of aging-induced performance degradation

6. CONCLUSION

In this paper, we present an efficient methodology for aging-induced timing analysis and optimization considering path sensitization. The analysis results reveal the importance and benefit of considering path sensitization for aging-aware timing optimization. Based on timed ATPG, we can identify the critical sub-circuit of a target circuit, which is truly necessary to be manipulated, and then apply transistor resizing as well as joint LR and PR to mitigate aging-induced performance degradation. Experiments demonstrate that our framework successfully recovers benchmark circuits from performance degradation with marginal cost. Lastly, the proposed methodology is scalable for large-scale designs due to the runtime efficiency.

REFERENCES

[1] ——, International Technology Roadmap for Semiconductor, 2009.

[2] D. K. Schroder and J. A. Babcock, "Negative bias temperature instability: road to cross in deep submicron silicon semiconductor manufacturing," *Journal of Applied Physics*, vol. 94, no. 1, pp. 1-18, July 2003.

[3] J. H. Stathis and S. Zafar, "The negative bias temperature instability in MOS devices: a review," *Microelectronics Reliability*, vol. 46, no. 2-4, pp. 270-286, Feb.-April 2006.

[4] S. Chakravarthi *et al.*, "A comprehensive framework for predictive modeling of negative bias temperature instability," in *Proc. of Int'l Reliability Physics Symp.* (IRPS), pp. 273-282, April 2004.

[5] N. Kimizuka *et al.*, "The impact of bias temperature instability for direct-tunneling ultra-thin gate oxide on MOSFET scaling," in *Proc. of Symp. on VLSI Technology*, pp. 73-74, June 1999.

[6] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "An analytical model for negative bias temperature instability," in *Proc. of ICCAD*, pp. 493-496, Nov. 2006.

[7] W. Wang *et al.*, "The impact of NBTI effect on combinational circuit: modeling, simulation, and analysis," *IEEE TVLSI*, vol. 18, no. 2, pp. 173-183, Feb. 2010.

[8] Y.-M. Kuo, Y.-L. Chang, and S.-C. Chang, "Efficient Boolean characteristic function for timed automatic test pattern generation," *IEEE TCAD*, vol. 28, no. 3, pp. 417-425, March 2009.

[9] B. C. Paul *et al.*, "Temporal performance degradation under NBTI: estimation and design for improved reliability of nanoscale circuits," in *Proc. of DATE*, pp. 780-785, March 2006.

[10] K. Kang *et al.*, "Efficient transistor-level sizing technique under temporal performance degradation due to NBTI," in *Proc. of ICCD*, pp. 216-221, Oct. 2006.

[11] X. Yang and K. Saluja, "Combating NBTI degradation via gate sizing," in *Proc. of ISQED*, pp. 47-52, March 2007.

[12] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "NBTI-aware synthesis of digital circuits," in *Proc. of DAC*, pp. 370-375, June 2007.

[13] K.-C. Wu and D. Marculescu, "Joint logic restructuring and pin reordering against NBTI-induced performance degradation," in *Proc. of DATE*, pp. 75-80, April 2009.

[14] H. Dadgour and K. Banerjee, "Aging-resilient design of pipelined architectures using novel detection and correction circuits," in *Proc. of DATE*, pp. 244-249, March 2010.

[15] Y. Wang *et al.*, "Temperature-aware NBTI modeling and the impact of input vector control on performance degradation," in *Proc. of DATE*, pp. 546-551, April 2007.

[16] D. R. Bild, G. E. Bok, and R. P. Dick, "Minimization of NBTI performance degradation using internal node control," in *Proc. of DATE*, pp. 148-153, April 2009.

[17] H.-C. Chen, D. H.-C. Du, and L.-R. Liu, "Critical path selection for performance optimization," *IEEE TCAD*, vol. 12, no. 2, pp. 185-195, Feb. 1993.

[18] W. Wang *et al.*, "An efficient method to identify critical gates under circuit aging," in *Proc. of ICCAD*, pp. 735-740, Nov. 2007.

[19] O. Coudert, "Gate sizing for constrained delay/power/area optimization," *IEEE TVLSI*, vol. 5, no. 4, pp. 465-472, Dec. 1997.