Optimized Self-Tuning for Circuit Aging

Evelyn Mintarno[*], Joëlle Skaf[*], Rui Zheng[†], Jyothi Velamala[†], Yu Cao[†],
Stephen Boyd[*], Robert W. Dutton[*], and Subhasish Mitra[*]
[*]Stanford University, CA, USA, [†]Arizona State University, AZ, USA

ABSTRACT

We present a framework and control policies for optimizing dynamic control of various self-tuning parameters over lifetime in the presence of circuit aging. Our framework introduces dynamic cooling as one of the self-tuning parameters, in addition to supply voltage and clock frequency. Our optimized self-tuning satisfies performance constraints at all times and maximizes a *lifetime computational power efficiency (LCPE)* metric, which is defined as the total number of clock cycles achieved over lifetime divided by the total energy consumed over lifetime. Our framework features three control policies: 1. *Progressive-worst-case-aging (PWCA)*, which assumes worst-case aging at all times; 2. *Progressive-on-state-aging (POSA)*, which estimates aging by tracking active/sleep mode, and then assumes worst-case aging in active mode and long recovery effects in sleep mode; 3. *Progressive-real-time-aging-assisted (PRTA)*, which estimates the actual amount of aging and initiates optimized control action. Simulation results on benchmark circuits, using aging models validated by 45nm CMOS stress measurements, demonstrate the practicality and effectiveness of our approach. We also analyze design constraints and derive system design guidelines to maximize self-tuning benefits.

1. INTRODUCTION

This paper is about designing energy-efficient robust systems in the presence of circuit aging, in particular the dominant aging mechanism induced by *Negative Bias Temperature Instability* (*NBTI*). NBTI effects can be severe for sub-65nm ICs. For example, the PMOS threshold voltage may degrade by 50mV over lifetime (e.g., 7-10 years) under worst-case operating conditions due to interface traps accumulated at the Si-SiO_2 interface. Depending on the design and the operating conditions, this may result in more than 20% speed degradation [Borkar 06, Chen 03, Hicks 08, Schroder 03]. NBTI-induced circuit delay degradation depends on several dynamic factors: the amount of time elapsed, temperature, workload, and voltage profiles [Zhang 08, Zheng 09].

In order to prevent delay faults due to aging – especially under worst-case usage conditions – designers traditionally incorporate *one-time worst-case guardbands* (*OWG*) at the beginning of lifetime. OWG examples include clock frequency reduction, supply voltage increase, device over-sizing, or combinations thereof. OWG is pessimistic and expensive because:

1. Transistor aging gets worse in advanced technologies [Borkar 05, Kuhn 08, McPherson 06].
2. Aging of devices on the same chip varies – not every PMOS transistor on a chip is stressed to worst-case level [Wang 07].
3. All systems may not be stressed to worst levels in the field [Agarwal 07, Sylvester 06].

The premise of this paper is: instead of using OWG, design a system that can compensate for NBTI-induced degradation by self-tuning various system parameters progressively over lifetime. Such self-tuning parameters must be adjusted dynamically according to performance demands (which may be time-varying), and adaptively according to actual system aging. The gradual nature of aging and its dependence on dynamic factors enable such a system to achieve better energy-efficiency compared to OWG.

However, self-tuning of various system parameters often leads to conflicting requirements. For example, supply voltage may be increased to compensate for aging-induced delay degradation – however, as a result, dynamic and leakage power will increase, and aging will get worse. Reducing clock frequency can prevent errors and also reduce dynamic power, but system performance degrades – depending on the amount of frequency reduction, system performance requirements may no longer be satisfied. While aging increases delay, it also reduces leakage power. Furthermore, the choice of self-tuning parameters made at any one point in time affects not only the instantaneous but entire future aging, performance, and energy consumption. Hence, there is a need for global optimization of self-tuning parameters.

In this paper, we present a framework for optimizing dynamic control of various self-tuning parameters over system lifetime, quantify achieved benefits, and derive system design guidelines to maximize self-tuning benefits. Our optimized self-tuning achieves the following two specifications:

1. Satisfies performance constraints throughout lifetime while ensuring reliable operation in the presence of circuit aging.
2. Maximizes our *lifetime computational power efficiency (LCPE)* metric which we define as the total number of clock cycles achieved over system lifetime divided by the total energy consumed over lifetime.

We also introduce dynamic cooling as a self-tuning parameter to control aging. Dynamic cooling allows us to reduce temperature by adjusting the amount of input power supplied to the cooling device [Lin 08]. The input power, which we denote as P_{cool}, determines the amount of heat that will be removed by the cooling device. Examples of dynamic cooling mechanisms include hybrid cooling (involving fan, heat sink, and heat pipe), thermoelectric cooling (TEC), and liquid cooling. However, such cooling capability is traditionally used only to maintain system temperature within specified limits. In this paper, we utilize the fact that activating cooling to reduce temperature slows down aging of devices, enhancing lifetime reliability. This can potentially reduce aging compensation that may be required later on. Cooling also considerably improves instantaneous temperature-dependent leakage power and delay. However, cooling increases instantaneous power consumption. We optimize time-varying usage of dynamic cooling jointly with other well-known system parameters (supply voltage and clock frequency) considering the effects on aging, performance and energy consumption over lifetime.

There are four "types" of user-inputs to our framework (Fig. 1.1):

1. Models for circuit aging, power consumption, temperature, and performance.
2. Circuit netlist and technology library for extracting design- and process-dependent model coefficients.
3. System constraints, such as performance constraint over lifetime and lifetime target. The performance constraint can also be time-varying.
4. Discrete values of self-tuning parameters available.

Our framework has three built-in control policies (PWCA, POSA, and PRTA) which will be detailed in Sec. 3. However, users can also input their own control policies. The output of our framework is the set of optimized values of self-tuning parameters to be applied online during system operation. The objective is to maximize LCPE under system constraints.

This paper makes the following major contributions:

1. We present a framework and three control policies (PWCA, POSA, and PRTA) to produce optimized dynamic control of self-tuning parameters over lifetime in the presence of circuit aging. Our

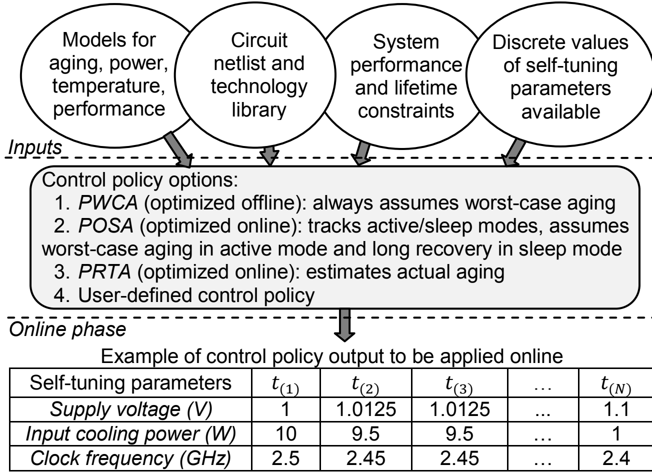
978-3-9810801-6-2/DATE10 © 2010 EDAA

optimized self-tuning satisfies performance constraints at all times and maximizes our lifetime computational power efficiency (LCPE) metric.

2. We introduce dynamic cooling as a self-tuning parameter to control aging. We also optimize its time-varying usage jointly with other self-tuning parameters considering effects on aging, performance, and energy consumption over lifetime.

3. We present simulation results on benchmark circuits, using aging models validated by 45nm CMOS stress measurements, to quantify the benefits of our optimized self-tuning. We also derive a set of system design guidelines to maximize self-tuning benefits.

Section 2 of this paper describes models and terminologies used in our formulation. Section 3 details our optimization framework and control policies. Section 4 presents simulation results. Section 5 discusses related work, followed by conclusions and design guidelines in Sec. 6.



Self-tuning parameters	$t_{(1)}$	$t_{(2)}$	$t_{(3)}$	…	$t_{(N)}$
Supply voltage (V)	1	1.0125	1.0125	…	1.1
Input cooling power (W)	10	9.5	9.5	…	1
Clock frequency (GHz)	2.5	2.45	2.45	…	2.4

Figure. 1.1. Our control system framework.

2. MODELS AND TERMINOLOGIES

2.1 Discrete time-steps

We discretize target lifetime into N uniformly-spaced time-steps: $t_{(i+1)} - t_{(i)} = dt,\ \forall\ i \in \{1,2,\dots,N\}$, where $t_{(i)}$ denotes the amount of time elapsed from the beginning of lifetime until the beginning of the ith time-step, and dt denotes the amount of time elapsed between each time-step. For example, $t_{(1)}$ denotes the time at the beginning of lifetime ($t_{(1)} = 0$), and $t_{(N)}$ denotes the time at the beginning of the last (N^{th}) time-step. At each time step, our control policies decide whether to adjust all or some (or none) of the self-tuning parameters, and, if so, the corresponding tuning-magnitude. Therefore, we do not pre-determine tuning-times. Our time-steps represent "possible" tuning-times. Depending on the control policy, the actual aging and the performance demand, we may or may not perform tuning at a particular time-step, i.e., the self-tuning parameters may stay constant over one or more time-steps. In our formulation, the time-steps do not necessarily have to be uniform. However, as long as each time-step is "fine" enough, the uniformity of time-steps does not compromise optimization results.

2.2 Control variables

The output of our control policies is a set of values for self-tuning parameters to be applied during the ith time-step (from the beginning until the end of the ith time-step). This set is denoted by u_i:

$$u_i = \{V_{\text{dd}(i)}, P_{\text{cool}(i)}, f_{(i)}\} \tag{1}$$

where V_{dd} denotes supply voltage, P_{cool} denotes user-input power for cooling, and f denotes clock frequency.

In light of concerns regarding the limited effectiveness of body-bias in advanced technologies [Borkar 04, Narendra 99], we do not consider body-bias in this paper, although our framework can include body-bias or any other self-tuning parameters.

2.3 Lifetime computational power efficiency (LCPE)

Our optimization objective is to maximize LCPE, which can be expressed as the total number of clock cycles achieved over all time-steps divided by the total energy consumed over all time-steps. Higher LCPE values indicate better overall energy efficiency over lifetime. The number of clock cycles achieved during the ith time-step is $f_{(i)}\ dt$. Due to aging, leakage power at the beginning is higher than at the end of each time-step. Since aging is a slow process, $P_{(i)}\ dt$ provides a tight upper bound for the energy consumed during the ith time-step, where $P_{(i)}$ denotes the total power consumption at the beginning of the ith time-step. Therefore, LCPE can be expressed as

$$LCPE = \frac{\sum_{i=1}^{N} f_{(i)}}{\sum_{i=1}^{N} P_{(i)}}\,. \tag{2}$$

2.4 Constraints

A system is expected to satisfy performance constraints over a target lifetime. The lower bound on the clock frequency during the ith time-step is determined by an application-dependent performance constraint $f_{\text{c}(i)}$ which can be time-varying. Aging during the ith time-step causes the circuit delay at the end of ith time-step $D_{(i)}$ to be greater than the delay at the beginning of the time-step. Hence, the upper bound on the clock frequency at the ith time-step is determined by the delay at the end of the time-step:

$$f_{\text{c}(i)} \le f_{(i)} \le\ 1/\left(D_{(i)} + \Delta\right) \tag{3}$$

where Δ is necessary to account for setup time, clock skew, jitter, and noise guardbands. Although we assume a lifetime constraint in this paper, our framework can be expanded to include the possibility of trading-off lifetime with energy-efficiency and performance.

2.5 Threshold voltage

The threshold voltage of a transistor at the beginning of the ith time-step, $V_{\text{th}(i)}$, is affected by the aging effect and the drain-induced barrier lowering (DIBL) effect [BSIM 09]. The DIBL effect can be approximated as a linear decrease in threshold voltage with increase in supply voltage [Martin 02, Hong 08]. Hence,

$$V_{\text{th}(i+1)} = V_{\text{th}(i)} + V_{\text{IT}(i+1)} - V_{\text{IT}(i)} -\ K_{\text{dibl}}\left(V_{\text{dd}(i+1)} - V_{\text{dd}(i)}\right) \tag{4}$$

Here, $V_{\text{IT}(i)}$ is the cumulative aging-induced shift in threshold voltage from time 0 up to the beginning of the ith time-step. Note that, $V_{\text{IT}(1)} = 0$ for a fresh circuit. Since PMOS threshold voltage is negative, every time we refer to "threshold voltage" we actually mean the "magnitude of threshold voltage."

From [Zheng 09], $V_{\text{IT}(i+1)}$ can be expressed as a function of $V_{\text{IT}(i)}$ and dynamic operating conditions between i^{th} and $(i+1)^{th}$ time-step. During active mode when V_{dd} is turned on, a system experiences dynamic-stress condition, where both stress (i.e., logic 0 at the input of a PMOS transistor) and recovery phases (i.e., logic 1 at the input of a PMOS transistor) impact aging. On the other hand, during sleep mode, when V_{dd} is turned off ($V_{\text{dd}} = 0$), only the recovery phase impacts aging.

As an example, consider the case when the ith time-step starts with active mode and is then followed by sleep mode until the end of the time-step, with $\eta_{(i)}$ as the fraction of time in active mode. Aging during the active mode increases the aging-induced threshold voltage shift from $V_{\text{IT}(i)}$ to $V_{\text{IT,m}(i)}$ at the end of the active mode:

$$V_{\text{IT,m}(i)}{}^{1/n} = V_{\text{IT}(i)}{}^{1/n} + \Phi_{(i)} \tag{5}$$

$$\Phi_{(i)} = K_{\text{aging}(i)}\left(V_{\text{dd}(i)} - V_{\text{th}(i)}\right)^2 e^{\frac{V_{\text{dd}(i)} - V_{\text{th}(i)}}{0.25\ E_0\, T_{\text{ox}}}} e^{-\frac{E_a}{K\, T_{(i)}}}\, \eta_{(i)}\left(t_{(i+1)} - t_{(i)}\right). \tag{6}$$

Recovery during the following sleep mode decreases the aging-induced threshold voltage shift from $V_{\text{IT,m}(i)}$ to $V_{\text{IT}(i+1)}$ at the end of the sleep mode:

$$V_{\mathrm{IT}(i+1)} = V_{\mathrm{IT,m}(i)}\left(1+\xi\left(1-\eta_{(i)}\right)\left(t_{(i+1)}-t_{(i)}\right)/t_{(i+1)}\right)^{-0.5} \quad (7)$$

where $K_{\mathrm{aging}(i)}$ represents workload (probability that PMOS is on) and $T_{(i)}$ is temperature at the ith time-step. E_a is the activation energy of interface bonds, K is the Boltzman's constant, and T_{ox} is the gate oxide thickness. n, E_{o}, ξ are process-dependent parameters. Worst-case aging during the ith time-step implies that $\eta_{(i)} = 1$ and $K_{\mathrm{aging}(i)} = \mathrm{WC\text{-}K_{aging}}$. $\mathrm{WC\text{-}K_{aging}}$ can be computed using worst-case signal probability for a given circuit, for example 0.95, which provides a tight upper bound on aging during dynamic-stress [Agarwal 08].

2.6 Power

At the beginning of the ith time-step, the instantaneous power consumption $P_{(i)}$ consists of dynamic power, leakage power, and user-input power for cooling:

$$P_{(i)} = P_{\mathrm{dyn}(i)} + P_{\mathrm{leak}(i)} + P_{\mathrm{cool}(i)}. \quad (8)$$

Dynamic power can be approximated as

$$P_{\mathrm{dyn}(i)} = K_{\mathrm{dyn}} V_{\mathrm{dd}(i)}{}^2 f_{(i)}. \quad (9)$$

Leakage power is approximated, following [Zhang 09], as

$$P_{\mathrm{leak}(i)} = T_{(i)}{}^2 K_{\mathrm{leak1}}\, V_{\mathrm{dd}(i)}\, e^{\frac{K_{\mathrm{leak2}} V_{\mathrm{dd}(i)}}{T_{(i)}}}\; e^{\frac{K_{\mathrm{leak3}} V_{\mathrm{th}(i)}}{T_{(i)}}} \quad (10)$$

where K_{dyn}, K_{leak1}, K_{leak2}, K_{leak3} are design- and process-dependent parameters.

2.7 Temperature

After adjustment of self-tuning parameters, interdependencies between temperature and leakage power cause a feedback loop which will converge to steady-state in less than one second [Skadron 03]. This is extremely short compared to a time-step in our formulation, since as will be demonstrated later, our time step is of the order of 5 days for NBTI-induced aging. Therefore, the use of steady state temperature and leakage power values introduces negligible error. Steady-state temperature at the beginning of the ith time-step can be approximated as in [Lin 08]:

$$T_{(i)} = T_o + R_{\mathrm{therm}}\left(P_{\mathrm{dyn}(i)} + P_{\mathrm{leak}(i)}\right) - R_{\mathrm{cool}} P_{\mathrm{cool}(i)} \quad (11)$$

where T_o, R_{therm}, R_{cool} depend on system thermal characteristics: T_o is the ambient temperature, R_{therm} is the thermal resistance, and R_{cool} is the coefficient of active cooling.

2.8 Delay

The delay $D_{(i)}$ at the end of the ith time-step (i.e., just before any tuning is applied at the beginning of the $(i+1)^{th}$ time-step), can be approximated as in [Sakurai 90]:

$$D_{(i)} = K_{\mathrm{delay1}} V_{\mathrm{dd}(i)}\left(1 + K_{\mathrm{delay2}} T_{(i)}\right) / \left(V_{\mathrm{dd}(i)} - V_{\mathrm{th,end}(i)}\right)^{\alpha} \quad (12)$$

where K_{delay1} and K_{delay2} are process- and design-dependent parameters. $V_{\mathrm{th,end}(i)}$ is the threshold voltage at the end of the ith time-step. Due to aging within the ith time-step, $V_{\mathrm{th,end}(i)}$ can be expressed as

$$\left(V_{\mathrm{th,end}(i)} - V_{\mathrm{th}(i)}\right) = \left(V_{\mathrm{IT}(i+1)} - V_{\mathrm{IT}(i)}\right). \quad (13)$$

Note that, our framework is not limited to use only the particular models in Sec. 2. It can be used in conjunction with other aging models that may incorporate other aging mechanisms, or any other power, temperature, and performance models.

3. CONTROL POLICIES

3.1 Progressive-worst-case-aging (PWCA)

PWCA acquires its benefits over OWG by applying self-tuning progressively over lifetime rather than just one-time at the beginning of lifetime. Such progressive self-tuning can adapt to gradual aging more efficiently than one-time guard-banding at the beginning of lifetime which accounts for worst-case aging effects at the end of lifetime. In order to guarantee reliable operation, PWCA assumes worst-case operating conditions. So $V_{IT(i)}$ is computed assuming that system is always in the active mode under worst-case workload. Therefore, PWCA results can be pre-computed offline at design-time, loaded into off-chip non-volatile memory, and invoked during run-time when resulting tuning-times match the time that the system has been in operation.

PWCA efficiently finds the globally optimal control actions that achieve the highest possible LCPE (under PWCA assumptions), through our non-enumerative *progressive-dynamic-programming (PDP)* algorithm detailed in [Mintarno 10] (not included in this paper for space constraints). PDP fully takes into account the entire future costs and benefits of a control action executed at any point in time. PDP captures the fact that lower (higher) present cost may induce higher (lower) future costs due to complex dynamic interplay between aging, power, and performance effects of self-tuning parameters over lifetime. PDP requires knowledge of frequency constraints over lifetime $f_{\mathrm{c}(i)}$ (which can be time-varying) at design time. When the frequency constraints are application-dependent and not known ahead of time, *progressive-greedy (PG)* algorithm (Algorithm 1, Fig. 3.1) can be used instead of PDP in PWCA.

3.2 Progressive-on-state-aging (POSA)

POSA further enhances self-tuning benefits by partially eliminating the worst-case aging assumptions in PWCA. Improved knowledge of system aging can enhance the quality of the control decision made, which in turn improves self-tuning benefits. POSA keeps track of system active/sleep modes, assumes worst-case aging during all the times spent in active mode, and accounts for long recovery effects during the times spent in sleep mode. At the beginning of each time-step, POSA estimates $V_{\mathrm{IT}(i)}$ using this approach and then determines the corresponding optimized values of self-tuning parameters with online optimization or from a lookup table generated at design-time. POSA utilizes the fact that aging is significantly recovered in sleep modes, due to long recovery effects which occur when V_{dd} is turned off for much longer than clock period. Such behavior was experimentally observed in [Tschanz 09, Zheng 09]. The specific benefits of POSA depend on system usage – simulation results in Sec. 4 indicate that POSA is highly beneficial for systems that spend significant amounts of time in sleep mode (e.g., power/battery sensitive systems, mobile systems, sensor networks).

3.3 Progressive-real-time-aging-assisted (PRTA)

At the beginning of each time-step, PRTA acquires real-time aging information to estimate $V_{\mathrm{IT}(i)}$, and then determines the corresponding optimized values of self-tuning parameters with online optimization or from a lookup table generated at design-time. Such real-time aging information allows PRTA to take into account not only the impact of long recovery effects during sleep modes, but also application-dependent aging during active modes. Inaccuracy of real-time aging information, power, and area impact of the techniques used to collect real-time aging information may reduce the net benefits of PRTA. Simulation results in Sec. 4 take those non-idealities into account, derive design guidelines to maximize PRTA benefits, and demonstrate that PRTA is highly beneficial for systems that experience workload with low K_{aging} characteristics.

Both POSA and PRTA optimize their control actions through our *progressive-greedy (PG)* algorithm. At time-step i, PG greedily chooses values of self-tuning parameters that maximize the instantaneous LCPE $f_{(i)}/P_{(i)}$ and meet performance constraints. To guarantee reliable operation, POSA and PRTA handle uncertainties in future aging reliably when computing $V_{\mathrm{IT}(i+1)}$, by assuming worst-case aging between time-step i and $(i+1)$.

Real-time aging information for PRTA can be obtained (or calibrated) from a variety of sources: 1. On-chip ring oscillators or canary circuits [Karl 08, Keane 08, Kim 08, Stawiasz 08, Tschanz 07]; 2. On-chip sensors such as temperature sensors (by predicting aging based on temperature profiles and assuming worst-case workload profiles) [Cabe 09, Kang 07, Simunic 04, Sylvester 06,

Srinivasan 05]; 3. Delay shift detectors [Agarwal 07, 08, Eireiner 07]; 4. On-line self-test and self-diagnostics [Baba 09, Li 08, 09].

Algorithm 1: Progressive-greedy (PG)

for $i = 1$ to N do
  calibrate $V_{\text{IT}(i)}$
  for each valid u_i do
    compute $V_{\text{th}(i)}, T_{(i)}, V_{\text{IT}(i+1)}, V_{\text{th,end}(i)}, P_{(i)}, D_{(i)}$
    if $f_{(i)}$ constraint is not satisfied, then $P_{(i)} \leftarrow \infty$
  end for
  choose u_i that maximizes $f_{(i)}/P_{(i)}$
end for

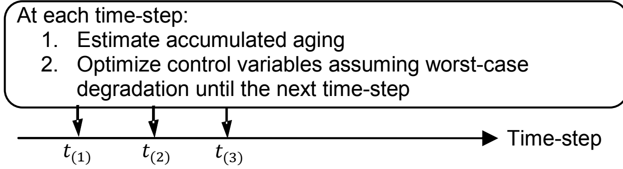


Figure 3.1. Progressive-greedy timeline.

4. SIMULATION RESULTS

In this section, we present simulation results for various benchmark circuits from [ISCAS 85, OpenCores 09, OpenSPARC 09] synthesized using the Synopsys Design Compiler using a 45nm technology library. We use aging models in Eqs. (5)–(7) calibrated using 45nm aging measurements in [Zheng 09]. We use f_c of 2.4GHz. Our target lifetime of 8 years is consistent with target lifetimes of 7–10 years for enterprise computing and communication platforms [Agarwal 07].

4.1. Benefits of control policies

The second column of Table 4.1 shows LCPE for the *no-aging* scenario. Here, we assume there is no aging in the circuit and choose the supply voltage and clock frequency values (which stay constant over the entire lifetime) that maximize LCPE, for the given constraint on f_c. The rest of Table 4.1 shows the % LCPE degradation of OWG and control policies (PWCA, POSA, and PRTA) compared to no-aging. OWG chooses a supply voltage value (which stays constant over the entire lifetime) that meets performance constraint f_c assuming worst-case aging at all times. Since both POSA and PRTA are beneficial when the system does not undergo worst-case aging at all times, POSA and PRTA are evaluated for a workload scenario which alternates between active and sleep modes. We assume the average proportion of time spent in active mode is 0.1 and the average K_{aging} during active modes is 0.1. Here we assume ideal implementations for PRTA (effects of non-idealities will be discussed later).

Table 4.2 summarizes the % OWG LCPE degradation recovered by the control policies, defined as

$$\text{\% OWG LCPE degradation recovered by control policy} = \left(\frac{\text{LCPE of control policy} - \text{LCPE of OWG}}{\text{LCPE of no-aging} - \text{LCPE of OWG}}\right) \times 100\% . \quad (14)$$

Table 4.2 shows that PWCA, POSA, and PRTA all substantially recover OWG LCPE degradation. On average, PWCA, POSA, and PRTA recover 52%, 83% and 93% of OWG LCPE degradation, respectively.

In our simulations, we used granularity of 5 days for time-step, 12.5mV for supply voltage, and 12.5MHz for clock frequency, since we found that they are sufficient to achieve maximized benefits, and finer granularities yield only marginally improved benefits. We found that % OWG LCPE degradation recovered quickly degrades as the time-step is increased to more than 30 days. On the other hand, it is only marginally improved by decreasing the time-step to less than 5 days. We observed that only 10-20 discrete levels of self-tuning parameters are needed, so significant modifications to existing hardware which provides discrete supply voltage and clock frequency, e.g. [Fischer 06], are not required.

Table 4.1. % LCPE degradation compared to no-aging (assuming ideal PRTA).

Benchmark Circuit	LCPE for no-aging (MHz/Watt)	%LCPE degradation compared to no-aging			
		OWG	PWCA	POSA	PRTA
C432	30.6	19.1%	10.1%	4%	1.7%
C499	29.8	18.7%	8.6%	2.9%	1.7%
C6288	30.4	19.3%	9.8%	3.8%	1.5%
OpenSPARC ALU	30.2	19.5%	9.4%	3.4%	1.3%
Ethernet Macstatus	29.5	17.7%	7.8%	2.2%	0.5%

Table 4.2. % OWG LCPE degradation recovered by control policies (assuming ideal PRTA).

Benchmark Circuit	PWCA	POSA	PRTA
C432	47%	78.9%	91.1%
C499	54%	84.7%	91%
C6288	50.5%	80.5%	92.5%
OpenSPARC ALU	52.1%	82.4%	93.4%
Ethernet Macstatus	55.7%	87.1%	97.2%

Figure 4.1 shows the optimal control actions of PWCA. Figure 4.1a shows the supply voltage over time, $V_{\text{dd}(i)}$, and Fig. 4.1b shows the fraction of power spent for cooling, $P_{\text{cool}(i)}/P_{(i)}$. For PWCA, supply voltage is increased gradually over time, whereas cooling is turned on aggressively at the beginning of lifetime, and then gradually decreased. Such behavior reveals that paying the power cost for cooling is more rewarding hence more desirable earlier in the lifetime. This is due to the fact that turning on cooling at any point in time reduces aging, which in turn reduces the compensation for aging that will be required later on. In particular, lower supply voltage can be used in the future, which reduces power consumption and further reduces aging. So the benefit of paying the power cost for cooling at any point in time is realized not only instantaneously (from reduced leakage power and delay) but also accumulated over the entire future after it is applied. Aging is also much more aggressive at the beginning of lifetime, so there is more opportunity there to suppress aging.

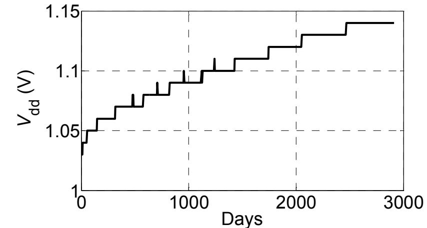


Figure 4.1a. PWCA control action (OpenSPARC ALU).

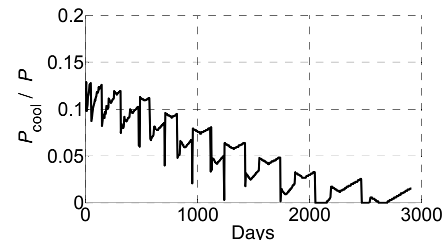


Figure 4.1b. PWCA control action (OpenSPARC ALU).

Sensitivity of % OWG LCPE degradation recovered by PRTA to two parameters of an application which alternates between active and sleep modes is shown in Fig. 4.2. The two parameters are the average proportion of time spent in active mode and the average K_{aging} during active modes.

Our framework and control policies can be applied to systems which support Dynamic Voltage and Frequency Scaling (DVFS). In traditional DVFS, the supply voltage associated with each frequency incorporates one-time worst-case aging guardbands [Burd 00, Nakai

05]. As detailed in [Mintarno 10], DVFS techniques incorporating our PWCA, POSA, or PRTA can provide for substantial improvements in LCPE compared to traditional DVFS.

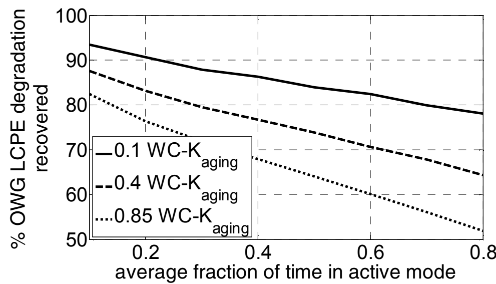


Figure 4.2. PRTA benefits (OpenSPARC ALU).

4.2. PRTA non-idealities

To evaluate the net benefits of PRTA, it is necessary to take into account inaccuracies, power, and area impact of the techniques used to collect real-time aging information. Inaccuracies arise from discrepancies between actual aging and the value reported by the real-time aging estimation technique used by PRTA. This inaccuracy necessitates additional margins which reduce the effectiveness of PRTA. For example, suppose that the measured delay is ±2ps of the actual delay, then the 2ps delay resolution needs to be added to the measured delay to account for optimistic measurements. The % OWG LCPE degradation recovered by PRTA as a function of delay resolution is shown in Table 4.3. If real-time aging information (with proper corrections) shows worse degradation than that predicted using POSA, then the latter can be used instead. Hence, when high-confidence aging models are used, PRTA cannot be worse than POSA. Depending on the implementation, PRTA can introduce additional power overheads. Fortunately, as discussed earlier, such aging estimation only needs to be activated infrequently, e.g., once every 5 days. Hence, circuits for real-time aging estimation may be turned on infrequently, which helps reduce the power impact of aging estimation (and also reduces the aging of the estimation circuitry itself). Table 4.4 shows the % OWG LCPE degradation recovered by PRTA as a function of power overhead of the aging estimation technique.

Table 4.3. Impact of delay resolution to % OWG LCPE degradation recovered by PRTA.

Benchmark Circuit	Delay resolution					
	Ideal	3ps	6ps	9ps	12ps	15ps
OpenSPARC ALU	93%	87%	79%	73%	66%	60%

Table 4.4. Impact of power overhead to % OWG LCPE degradation recovered by PRTA.

Benchmark Circuit	% power overhead				
	Ideal	0.25%	0.5%	0.75%	1%
OpenSPARC ALU	93%	92.1%	90.8%	89.6%	88.3

5. RELATED WORK

Several recent papers describe adaptive voltage scaling and/or adaptive body-biasing methods for circuit aging but do not address global optimization aspects and assume worst-case aging at all times. [Tiwari 08] aims to minimize worst-case aging effects at the end of lifetime by dividing lifetime into two phases, and then iteratively pre-selects only one self-tuning parameter (either supply voltage or body-bias) to be adjusted at each of the two phases. [Zhang 09] gradually increases supply voltage over lifetime to compensate for worst-case aging effects. In contrast, we optimize time-varying usage of multiple self-tuning parameters. [Kumar 09] assumes several tuning-times (when self-tuning parameters will be adjusted) and then decides body-bias and supply voltage values at each tuning-time to compensate for worst-case aging.

Comparison between our work and three state-of-the-art methods is summarized in Table 5. While also assuming worst-case aging at all times, PWCA overcomes the limitations of the state-of-the-art methods by finding the globally optimized control actions. PWCA outperforms the state-of-the-art approaches. As a comparison point, approaches in [Kumar 09, Tiwari 08, Zhang 09] recover only 15–32% of OWG LCPE degradation. In addition to PWCA, our framework provides POSA and PRTA control policies which do not always assume worst-case aging and use special aging estimation techniques.

[Agarwal 07, 08, Sylvester 06, Tschanz 07] discuss design of adaptive circuits and systems but do not address how to dynamically control self-tuning parameters. An adaptive feedback control approach for process- and workload-variations is described in [Ogras 08]. However, aging is not addressed. Dynamic Reliability Management (DRM) techniques are typically applied at higher abstraction levels [Simunic 04, Srinivasan 04, 05, Urmanov 04, Pop 07, Zhang 04]. In fact, DRM techniques can benefit from fine-grained self-tuning in this paper. Static NBTI mitigation techniques at process and circuit levels by optimizing transistor sizing and input vector control (IVC) are described in [Paul 07, Wang 07]. These techniques are similar to OWG. Interactions between "static" and "dynamic" techniques, as well as between high- and low-level techniques can provide insights into optimized robust design methodologies from circuits to systems – hence this is an important future research topic.

6. CONCLUSIONS

Optimization framework and control policies presented in this paper provide a basis for fine-grained self-tuning for designing energy-efficient robust systems. They deliver significant benefits relative to traditional one-time worst-case guardbands and state-of-art approaches. Based on our results, we derive a set of simple system design guidelines:

1. The choice of a particular self-tuning control policy for circuit aging depends on workload characteristics: If a system is primarily in the active mode under nearly worst-case operating condition at all times, then PWCA is sufficient. On the other hand, for a system that spends a significant amount of time in sleep mode, substantial benefits can be obtained by using POSA. For a system with low workloads characteristics, PRTA delivers significant benefits.

2. For PRTA control policy, acquiring real-time-aging every 5 days is sufficient. Attention must be paid to the resolution and cost of supporting techniques for aging estimation. Target delay resolution of less than 15ps and target power cost of less than 1% are desired.

Future research directions include: 1. Incorporation of other reliability mechanisms beyond NBTI-induced aging (e.g., PBTI); 2. New scheduling techniques in multi-core systems to complement the self-tuning techniques in this paper; 3. Interactions with high-level DRM techniques (including prediction of thermal characteristics); 4. Study of the spatial granularity of self-tuning; 5. Detailed validation of derived optimized self-tuning using CMOS aging experiments.

ACKNOWLEDGMENT

This research was supported in part by the FCRP C2S2, NSF, and a Stanford Starr Fellowship. We thank Nishant Patil of Stanford Robust Systems Group and the reviewers for their comments.

REFERENCES

[Agarwal 07] Agarwal, M., B. Paul, M. Zhang and S. Mitra, "Circuit Failure Prediction and its Application to Transistor Aging," *Proc. VLSI Test Symp.*, 2007.

[Agarwal 08] Agarwal, M., *et al*., "Optimized Circuit Failure Prediction for Aging: Practicality and Promise," *Proc. Int. Test Conf.*, 2008.

[Baba 09] Baba, H. and S. Mitra, "Testing for Transistor Aging," *Proc. VLSI Test Symp.*, 2009.

[Borkar 04] Borkar, S., "Circuit Techniques for Subthreshold Leakage Avoidance, Control and Tolerance," *Proc. Electron Devices Meeting*, 2004.

Table 5. Comparison between our work and state-of-the-art methods.

Control Policy	Pre-determine when to adjust each self-tuning parameter	Pre-select only one self-tuning parameter to be adjusted at each point in time	Ignore full effects of current actions into entire future	Sub-optimal LCPE	Assume worst-case aging at all times	Dynamic cooling as a self-tuning parameter	General framework for users to decide control policies
[Tiwari 08]	(-) Yes	(-) Yes	(-) Yes	(-) Yes	(-) Yes	(-) No	(-) No
[Zhang 09]	(-) Yes	(-) Yes	(-) Yes	(-) Yes	(-) Yes	(-) No	(-) No
[Kumar 09]	(-) Yes	(+) No	(-) Yes	(-) Yes	(-) Yes	(-) No	(-) No
This paper	(+) No (determined as part of optimization process)	(+) No (determined as part of optimization process)	(+) No (for PDP)	(+) No	(-) Yes (for PWCA) (+) No (for POSA/PRTA)	(+) Yes	(+) Yes

[Borkar 05] Borkar, S., "Designing Reliable Systems from Unreliable Components," *MICRO*, Nov.-Dec. 2005.

[Borkar 06] Borkar, S., "Electronics beyond nano-scale CMOS," *Proc. Des. Autom. Conf.*, 2006.

[BSIM 09] BSIM User Manual, University of California, Berkeley.

[Burd 00] Burd, T. D., *et al*., "A Dynamic Voltage Scaled Microprocessor System," *Journal of Solid-State Circuits*, Nov. 2000.

[Cabe 09] Cabe, A., *et al*., "Small embeddable NBTI sensors (SENS) for tracking on-chip performance decay," *Proc. Int. Symp. on Quality Electronic Design*, 2009.

[Chen 03] Chen, G., *et al*., "Dynamic NBTI of PMOS transistors and its impact on device lifetime," *Proc. Int. Reliability Physics Symp.*, 2003.

[Eireiner 07] Eireiner, M., *et al*., "In-Situ Delay Characterization and Local Supply Voltage Adjustment for Compensation of Local Parametric Variations," *Journal of Solid-State Circuits*, July 2007.

[Fischer 06] Fischer, T., *et al*., "A 90-nm Variable Frequency Clock System for a Power-Managed Itanium Architecture Processor," *Journal of Solid-State Circuits*, Jan. 2006.

[Hicks 08] Hicks, J., *et al*., "Intel's 45nm CMOS Technology," *Intel Technology Journal*, June 2008.

[Hong 08] Hong, S., *et al*., "Dynamic Voltage Scaling of Supply and Body Bias Exploiting Software Runtime Distribution," *Proc. Design Automation and Test in Europe*, 2008.

[Kang 07] Kang, K., *et al*., "Characterization and Estimation of Circuit Reliability Degradation under NBTI using On-Line IDDQ Measurement," *Proc. Des. Autom. Conf*, 2007.

[Karl 08] Karl, E., *et al*., "Compact In-Situ Sensors for Monitoring Negative-Bias-Temperature-Instability Effect and Oxide Degradation," *Proc. Int. Solid-State Circuits Conf.*, 2008.

[Keane 08] Keane, J., *et al*., "An On-Chip NBTI Sensor for Measuring PMOS Threshold Voltage Degradation," *Proc. Int. Symp. on Low Power Electronics and Design*, 2007.

[Kim 08] Kim, T., *et al*., "Silicon Odometer: An On-Chip Reliability Monitor for Measuring Frequency Degradation of Digital Circuits," *Journal of Solid-State Circuits*, April 2008.

[Kuhn 08] Kuhn, K., *et al*., "Managing Process Variation in Intel's 45nm CMOS Technology," *Intel Technology Journal*, June 2008.

[Kumar 09] Kumar, S. V., *et al*., "Adaptive Techniques for Overcoming Performance Degradation due to Aging in Digital Circuits", *Proc. Asia and South Pacific Des. Autom. Conf.*, 2009.

[Li 08] Li, Y., S. Makar, and S. Mitra, "CASP: Concurrent Autonomous Chip Self-Test using Stored Test Patterns", *Proc. Design Automation and Test in Europe,* 2008.

[Li 09] Li, Y., *et al*., "Overcoming Early-Life Failure and Aging for Robust Systems", *Design & Test*, 2009.

[Lin 08] Lin, S., and K. Banerjee, "Cool Chips: Opportunities and Implications for Power and Thermal Management," *Trans. on Electron Devices*, Jan. 2008.

[Martin 02] Martin, S. M., *et al*., "Combined Dynamic Voltage Scaling and Adaptive Body Biasing for Low Power Microprocessors," *Proc. Int. Conf. on Comput.-Aided Design,* 2002.

[McPherson 06] McPherson, J.W., "Reliability Challenges for 45nm and Beyond," *Proc. Des. Autom. Conf*, 2006.

[Mintarno 10] Mintarno, E., *et al*., "Optimized Self-Tuning for Circuit Aging," *Technical Report, Stanford University*, 2010, url: http://www.stanford.edu/group/rsg_csl.

[Narendra 99] Narendra, S., *et al*., "Impact of Using Adaptive Body Bias to Compensate Die-to-die Vt Variation on Within-die Vt Variation*, Proc. Int. Symp. on Low Power Electronics and Design*, 1999.

[Nakai 05] Nakai, M., *et al*., "Dynamic voltage and frequency management for a low power embedded micro-processor," *Journal of Solid-State Circuits*, Jan. 2005.

[Ogras 08] Ogras, U. Y., *et al*., "Variation-Adaptive Feedback Control for Networks-on-Chip with Multiple Clock Domains," *Proc. Des. Autom. Conf*, 2008.

[Paul 07] Paul, B.C., *et al*., "Negative Bias Temperature Instability: Estimation and Design for Improved Reliability of Nanoscale Circuits," *Trans. Comput.-Aided Design Integr. Circuits Syst.*, April 2007.

[Pop 07] Pop, P., *et al*., "Scheduling and voltage scaling for energy/reliability trade-offs in fault-tolerant time-triggered embedded systems," *Proc. Conf. Hardware/Software Co-design and System Synthesis,* 2007.

[Sakurai 90] Sakurai, T., and R. Newton, "Alpha-Power Law Mosfet Model and its Applications to CMOS Inverter Delay and Other Formulas," *Journal of Solid-State Circuits*, April 1990.

[Schroder 03] Schroder, D. K., *et al*., "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," *Journal of Applied Physics*, 2003.

[Simunic 04] Simunic, T., K. Mihic and G. De Micheli, "Reliability and Power Management of Integrated Systems," *Proc. Euromicro Symp. Digital Systems Design*, 2004.

[Skadron 03] Skadron, K., *et al*., "Temperature Aware Microarchitecture," *Proc. Int. Symp. Comput.-Architecture*, 2003.

[Srinivasan 04] Srinivasan, J., *et al*., "The Case for Lifetime Reliability Aware Microprocessors," *Proc. Int. Symp. Comput.-Architecture*, 2004.

[Srinivasan 05] Srinivasan, J., *et al*., "Exploring Structural Duplication for Lifetime Reliability Enhancement," *Proc. Int. Symp. Comput.-Architecture*, 2005.

[Stawiasz 08] Stawiasz, K., *et al*., "On-Chip Circuit for Monitoring Frequency Degradation due to NBTI," *Proc. Int. Reliability Physics Symp.*, 2008.

[Sylvester 06] Sylvester, D., *et al*., "ElastIC: An Adaptive Self-Healing Architecture for Unpredictable Silicon," *Design & Test*, Nov.-Dec. 2006.

[Tiwari 08] Tiwari, A., and J. Torrellas, "Facelift: Hiding and Slowing Down Aging in Multicores," *Proc. Int. Symp. on Microarchitecture*, 2008.

[Tschanz 07] Tschanz, J. W., *et al*., "Adaptive Frequency and Biasing Techniques for Tolerance to Dynamic Temperature-Voltage and Aging," *Proc. Int. Solid-State Circuits Conf.*, 2007.

[Tschanz 09] Tschanz, J. W., *et al*., "Tunable Replica Circuits and Adaptive Voltage-Frequency Techniques for Dynamic Voltage, Temperature, and Aging Variation Tolerance," *Proc. Symp VLSI Circuits*, 2009.

[Urmanov 04] Urmanov, A., *et al*., "A New Sensor Validation Technique for the Enhanced RAS of High End Servers," *Proc. Int. Conf. Machine Learning; Models, Technologies and Applications*, 2004.

[Wang 07] Wang, Y., *et al*., "Temperature-aware NBTI modeling and the impact of input vector control on performance degradation," *Proc. Design Automation and Test in Europe*, 2007.

[Zhang 04] Zhang, Y., and K. Chakrabarty, "Task Feasibility Analysis and Dynamic Voltage Scaling in Fault-Tolerant Real-Time Embedded Systems," *Proc. Design Automation and Test in Europe*, 2004.

[Zhang 08] Zhang, B. and M. Orshansky, "Modeling of NBTI-Induced PMOS Degradation under Arbitrary Dynamic Temperature Variation," *Proc. Int. Symp. on Quality Electronic Design*, 2008.

[Zhang 09] Zhang, L., and R. Dick, "Scheduled Voltage Scaling for Increasing Lifetime in the Presence of NBTI," *Proc. Asia and South Pacific Des. Autom. Conf.*, 2009.

[Zheng 09] Zheng, R., *et al*., "Circuit Aging Prediction for Low Power Operation," *Proc. Custom Integrated Circuits Conf.*, 2009.