Self-Tuning for Maximized Lifetime Energy-Efficiency in the Presence of Circuit Aging

Evelyn Mintarno, Joëlle Skaf, Rui Zheng, Jyothi Bhaskar Velamala, Yu Cao, *Senior Member, IEEE,* Stephen Boyd, *Fellow, IEEE,* Robert W. Dutton, *Fellow, IEEE,* and Subhasish Mitra, *Senior Member, IEEE*

***Abstract*—This paper presents an integrated framework, together with control policies, for optimizing dynamic control of self-tuning parameters of a digital system over its lifetime in the presence of circuit aging. A variety of self-tuning parameters such as supply voltage, operating clock frequency, and dynamic cooling are considered, and jointly optimized using efficient algorithms described in this paper. Our optimized self-tuning approach 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. We present 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 modes, and then assumes worst-case aging in active mode and long recovery effects in sleep mode; and 3) *progressive-real-time-aging-assisted (PRTA)*, which acquires real-time information and initiates optimized control actions. Various flavors of these control policies for systems with dynamic voltage and frequency scaling (DVFS) are also analyzed. Simulation results on benchmark circuits, using aging models validated by 45 nm measurements, demonstrate the effectiveness and practicality of our approach in significantly improving LCPE and/or lifetime compared to traditional one-time worst-case guardbanding. We also derive system design guidelines to maximize self-tuning benefits.**

***Index Terms*—Adaptive supply voltage and clock frequency, circuit aging, energy-efficiency, lifetime reliability.**

I. Introduction

THIS PAPER addresses the major challenge of designing robust and energy-efficient systems in the presence of circuit aging. We focus on a dominant circuit aging mechanism induced by *Negative Bias Temperature Instability (NBTI)*. NBTI effects can be significant for sub-65 nm integrated circuits [1]–[3]. The PMOS threshold voltage may gradually degrade by 50 mV over lifetime (e.g., 7–10 years) under worst-case operating conditions due to 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 [1]–[4]. Aging-induced changes in the interface charge depend on the process technology and several dynamic factors: the amount of time elapsed, temperature, workload, and voltage profiles [5]–[7]. While we focus on NBTI, it is possible to extend our framework for other reliability mechanisms, e.g., Positive Bias Temperature Instability (PBTI), Electromigration (EM), Time Dependent Dielectric Breakdown (TDDB), Gate Oxide Integrity (GOI), Thermal Cycling (TC), and Hot Carrier Injection (HCI).

In order to prevent delay faults due to circuit aging, designers traditionally incorporate *one-time worst-case guardbands (OWG)* at the beginning of lifetime while accounting for the worst-case aging effects at the end of lifetime. OWG examples include clock frequency reduction, supply voltage increase, and device over-sizing. OWG is pessimistic and demands expensive power/performance/area costs because: 1) circuit aging is expected to get worse in advanced technologies [8]–[10]; 2) not every device on a given chip is stressed to worst-case levels [5]; and 3) all systems may not be stressed to worst-case levels in the field [11], [12].

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

Unfortunately, self-tuning of various system parameters often leads to conflicting results. For example, increasing supply voltage may compensate for aging-induced delay degradation; however, it increases dynamic and leakage power, as well as chip temperature, and accelerates aging. Reducing clock frequency can prevent errors and also reduce dynamic power; but system speed degrades, and overall performance requirements may no longer be satisfied. While NBTI-induced aging increases delay, it also reduces leakage power due to degraded threshold voltage. Furthermore, the choice of self-tuning parameters made at any one point in time affects future

Manuscript received February 10, 2010; revised May 31, 2010, August 25, 2010, and October 21, 2010; accepted October 22, 2010. Date of current version April 20, 2011. This work was supported in part by the Focus Center Research Program Center for Circuit and System Solutions, the National Science Foundation, and the Semiconductor Research Corporation. This paper was recommended by Associate Editor D. Sylvester.

E. Mintarno, S. Boyd, and R. W. Dutton are with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: evemint@stanford.edu; boyd@stanford.edu; rdutton@stanford.edu).

J. Skaf is with Google, Inc., New York, NY 10016 USA (e-mail: joelle.skaf@gmail.com).

R. Zheng, J. Velamala, and Y. Cao are with the Department of Electrical Engineering, Arizona State University, Tempe, AZ 85287 USA (e-mail: rui.zheng.1@asu.edu; jvelamal@asu.edu; yu.cao@asu.edu).

S. Mitra is with the Department of Electrical Engineering and Department of Computer Science, Stanford University, Stanford, CA 94305 USA (e-mail: subh@stanford.edu).

Digital Object Identifier 10.1109/TCAD.2010.2100531

aging, performance, and energy consumption. Hence, there is need for global optimization of self-tuning parameters over lifetime, considering their long-term effects and interactions.

In this paper, we present a general framework and control policies for jointly optimizing multiple self-tuning parameters over system lifetime. We also present efficient algorithms to accomplish such joint optimization. In addition to the tuning of supply voltage and operating clock frequency, we consider dynamic cooling, e.g., via variable fan control, as a possible self-tuning parameter. Dynamic cooling allows us to adjust system temperature by varying the input power supplied to the cooling device. Dynamic cooling is generally used for dynamic thermal management [13], [14]. For optimized self-tuning to overcome circuit aging, we jointly optimize complex system-level tradeoffs between the positive effects of cooling on circuit aging, leakage power, and delay, and the negative effects of power spent for cooling.

Our framework achieves the following objectives.

1) It satisfies performance constraints throughout the entire lifetime while ensuring reliable operation in the presence of circuit aging.
2) It maximizes a *lifetime computational power efficiency (LCPE)* metric which is defined as the performance achieved (i.e., the total number of clock cycles) over system lifetime divided by the total energy consumed over lifetime.

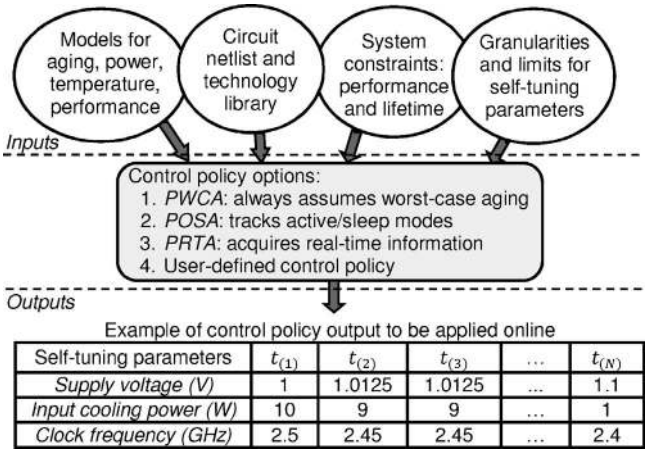
There are four "types" of user-inputs to our framework (Fig. 1).

1) Thermally-aware models for aging, power consumption, and performance.
2) Circuit netlist and technology library.
3) System constraints, such as required performance over lifetime, and target lifetime. System performance constraints can be time-varying.
4) Discrete values of self-tuning parameters available.

The framework has three built-in control policies [progressive-worst-case-aging (PWCA), progressive-on-state-aging (POSA), and progressive-real-time-aging-assisted (PRTA)] which will be detailed in Section III. However, the user can also implement alternative control policies. The output is a set of optimized values of self-tuning parameters, to be applied online during operation.

The main contributions of this paper are as follows.

1) A general framework, together with three control policies (PWCA, POSA, and PRTA) and efficient algorithms, to produce optimized dynamic control of multiple self-tuning parameters over lifetime. The optimized self-tuning satisfies system constraints, and maximizes the LCPE metric.
2) Introduction of dynamic cooling as a system-level self-tuning parameter that is jointly optimized with supply voltage and operating frequency to control aging, system power consumption, and system performance over lifetime.
3) Simulation results on benchmark circuits using aging models validated by 45 nm CMOS stress measurements. The results quantify the benefits of our optimized self-tuning approach. We also derive a set of system design guidelines to maximize self-tuning benefits.



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	9	...	1
Clock frequency (GHz)	2.5	2.45	2.45	...	2.4

Fig. 1. Our control system framework.

Section II describes the problem formulation. Section III details the framework and control policies. Section IV presents simulation results. Section V discusses related work, followed by conclusions and design guidelines in Section VI.

II. Models and Terminologies

A. Discrete Time-Steps

We discretize target system lifetime into N uniformly-spaced time-steps (Fig. 2)

$$t_{(i+1)} - t_{(i)} = dt, \quad i = 1, 2, \ldots, N \tag{1}$$

where $t_{(i)}$ denotes the amount of time elapsed from the beginning of lifetime until the beginning of time-step i, and dt denotes the amount of time elapsed in 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 (Nth) time-step. At each time step, the control policies decide whether to adjust all, some, or none of the self-tuning parameters; if adjustments are made, the corresponding tuning-magnitude is also decided. Therefore, tuning-times are not pre-determined. Time-steps represent "possible" tuning-times. Depending on the control policy, the actual aging, and performance demand, tuning may or may not be performed at a particular time-step; the self-tuning parameters may stay constant over one or more time-steps. In our formulation, the time-steps actually do not necessarily have to be uniform; they can be made fine grained in the beginning of lifetime to respond to fast aging during that period. However, as long as each time-step is "fine" enough, the uniformity of time-steps does not compromise the optimization results. Later in Section IV, we will discuss proper choice of time-step.

B. Control Variables

The control policies choose a set of control variables (i.e., values for self-tuning parameters) to be applied during each

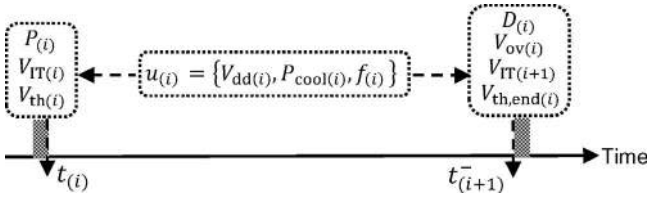


Fig. 2. Time-dependent terminologies. $t_{(i)}$ denotes the beginning of time-step i. $t^{-}_{(i+1)}$ denotes the end of time-step i.

time-step, from the beginning until the end of the time-step. At time-step i, this set of control variables is denoted by u_i

$$u_i = \{V_{dd(i)}, P_{cool(i)}, f_{(i)}\} \tag{2}$$

where $V_{dd(i)}$ denotes supply voltage, $P_{cool(i)}$ denotes user-input power for cooling, and $f_{(i)}$ denotes clock frequency.

In light of concerns regarding the limited effectiveness of body-bias in advanced technologies [15], [16], body-bias is not considered in this paper, although the policies can include body-bias or any other self-tuning parameters.

C. Lifetime Computational Power Efficiency (LCPE)

The 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 time-step i is $f_{(i)}\,dt$. Energy consumed during time-step i is the integral of power consumption over the time-step. Due to aging, leakage power at the beginning is higher than that at the end of each time-step. Since aging is a slow process, $P_{(i)}\,dt$ provides an upper bound for the energy consumed during time-step i, where $P_{(i)}$ denotes the total power consumption at the beginning of time-step i. Therefore, LCPE can be expressed as

$$\text{LCPE} = \frac{\sum_{i=1}^{N} f_{(i)}}{\sum_{i=1}^{N} P_{(i)}}. \tag{3}$$

D. Constraints

Each self-tuning parameter must be within its upper and lower limits

$$V_{dd,min} \le V_{dd(i)} \le V_{dd,max} \tag{4}$$

$$P_{cool,min} \le P_{cool(i)} \le P_{cool,max}. \tag{5}$$

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

$$f_{c(i)} \le f_{(i)} \le \frac{1}{D_{(i)} + \Delta} \tag{6}$$

where Δ is necessary to account for setup time, clock skew, jitter, and noise guardbands. Although a lifetime constraint is assumed in this paper, our framework can include the possibility of trading-off lifetime with energy-efficiency and/or performance. To guarantee reliable operation, temperature must be within the specified limits $T_{min} \le T_{(i)} \le T_{max}$. Moreover, the lowest gate overdrive during each time step, $V_{ov}(i)$, must be greater than a minimum gate overdrive of $V_{ov,min}$. Aging during time-step i causes threshold voltage at the end to be greater than that at the beginning of the time-step. So $V_{ov(i)}$ is determined by $V_{th,end(i)}$ which is the threshold voltage at the end of time-step i

$$V_{ov(i)} = V_{dd(i)} - V_{th,end(i)} \ge V_{ov,min}. \tag{7}$$

E. Threshold Voltage

The threshold voltage of a transistor at the beginning of time-step i, $V_{th(i)}$, is affected by the aging effect and the drain-induced barrier lowering (DIBL) effect [17]. Since PMOS threshold voltage is negative, every time we refer to V_{th} we actually mean the magnitude of V_{th}. To work around aging dependence on all previous operating conditions from time 0, the incremental change in V_{th} is computed depending only on the dynamic operating conditions within each time-step. The cumulative aging-induced shift in threshold voltage from time 0 up to the beginning of time-step i is denoted as $V_{IT(i)}$. The increase in interface traps, N_{IT}, leads to a linear shift in threshold voltage [18]. Hence

$$V_{IT(i+1)} - V_{IT(i)} = q(N_{IT(i+1)} - N_{IT(i)})/C_{ox} \tag{8}$$

where $N_{IT(i)}$ is the amount of interface traps accumulated from time 0 up to the beginning of time-step i, q is the elementary charge, and C_{ox} is the gate-oxide capacitance. Note that $V_{IT(1)} = 0$ and $N_{IT(1)} = 0$ for a fresh circuit. The DIBL effect can be approximated as a linear decrease in threshold voltage with increase in supply voltage [19], [20]. The incremental change in V_{th} is then written in terms of a difference equation

$$V_{th(i+1)} = V_{th(i)} + V_{IT(i+1)} - V_{IT(i)} - K_{dibl}(V_{dd(i+1)} - V_{dd(i)}) \tag{9}$$

where K_{dibl} is a process-dependent constant.

Based on [7], $V_{IT(i+1)}$ can be expressed as a function of $V_{IT(i)}$ and dynamic operating conditions between time-steps i and $(i + 1)$. During *active mode* when V_{dd} is turned on, the system experiences dynamic-stress condition where both the *stress phase* and *recovery phase* alternately impact aging (Fig. 3). In the stress phase, interface traps are increased, and in the recovery phase, it is partially reduced [18]. The stress phase occurs during negative gate-source voltage or logic 0 at the input, where the presence of inversion layer holes weakens the Si–H bonds. Dissociation of the bonds along the $Si–SiO_2$ interface causes the generation of interface charges and unbonded hydrogen atoms. Each pair of hydrogen atoms combine to generate molecular hydrogen which then diffuses away from the $Si–SiO_2$ interface. The recovery phase occurs when the gate-source bias is removed or logic 1 is at the input,

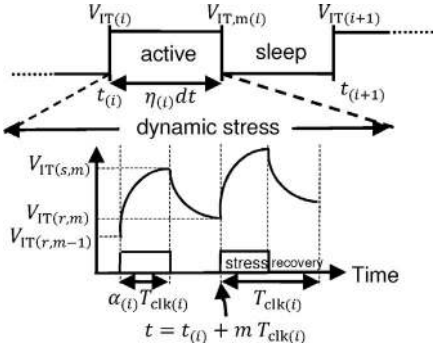


Fig. 3. Example for aging under dynamic operation.

where molecular hydrogen diffuses back toward the interface and recombines to anneal the broken Si bonds. During *sleep mode*, when V_{dd} is turned off ($V_{dd} = 0$), the system experiences the *long recovery phase*. For alternating active/sleep modes, the presence of a long recovery phase during sleep mode significantly changes the diffusion profile that continues into the subsequent active mode. Hence, special aging models and boundary conditions are required to connect the next active mode with the sleep mode—otherwise, degradation can be over-estimated [7].

For simplicity, consider an example where time-step i starts with an active mode followed by sleep mode until the end of the time-step, with $\eta_{(i)}$ as the fraction of time in active mode (Fig. 3). Aging during the active mode increases the aging-induced threshold voltage shift from $V_{IT(i)}$ to $V_{IT,m(i)}$ at the end of the active mode

$$V_{IT,m(i)}{}^{1/n} = V_{IT(i)}{}^{1/n} + \Phi_{(i)} \tag{10}$$

$$\Phi_{(i)} = K_p K_{aging(i)} (V_{dd(i)} - V_{th(i)})^2 e^{\frac{V_{dd(i)} - V_{th(i)}}{0.25 E_o T_{ox}}} e^{-\frac{Ea}{KT_{(i)}}} \eta_{(i)} (t_{(i+1)} - t_{(i)}). \tag{11}$$

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

$$V_{IT(i+1)} = V_{IT,m(i)} (1 + \xi(1 - \eta_{(i)})(t_{(i+1)} - t_{(i)})/t_{(i+1)})^{-0.5} \tag{12}$$

where $K_{aging(i)}$, a scalar from the interval (0, 1), a function of stress probability (probability for negative gate-source voltage). $T_{(i)}$ is the temperature at time-step i, E_a is the activation energy of interface bonds, K is the Boltzman's constant, and T_{ox} is the gate oxide thickness. NBTI-induced performance degradation is independent of clock frequency (for most practical clock frequencies). Several coefficients of the aging model $\{n, K_p, E_o, \xi\}$ which capture sensitivities to process technologies are calibrated to 45 nm aging measurements (Fig. 4). For example, n can be found from the logarithmic slope of degradation versus time in the active mode; K_p and E_o can be found from the data for different V_{dd}; and ξ can be found from the sleep mode data. Fig. 4 provides clear evidence that the model effectively predicts aging behavior for dynamic operation. It also shows that a relatively small number of calibrations can establish good visibility for predictive modeling. The specific dynamic operation case illustrated in Fig. 3 is used only as an example for simplicity of explanation. The boundary conditions modeled in (10)–(12) can be applied to directly compute shifts in aging for the general dynamic operation scenario with multiple active-sleep transitions, where the active modes may have time-varying supply voltage, stress probability, or temperature. Equations (10) and (11) can be used to compute the increment during active mode, it is evaluated every time there is a change in the dynamic factors. Equation (12) can be used to compute the decrement during sleep mode.

The exact delay degradation of a circuit depends on the amount of time that various circuit nodes are at logic 0 or 1 (signal probabilities), which in turn depends on the application-dependent input vectors during operation which are not known *a priori*. Therefore, a safe and tight upper bound for circuit delay degradation under worst-case signal probabilities is required for reliable operation. In most practical cases, it can be obtained by assuming WC-K_{aging} of 0.95 for the entire circuit [5], [21], [22], [38]. Worst-case aging during time-step i implies that the system is always in the active mode under worst-case workload, i.e., $\eta_{(i)} = 1$ and $K_{aging(i)} = \text{WC-}K_{aging}$.

F. Power

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

$$P_{(i)} = P_{dyn(i)} + P_{leak(i)} + P_{cool(i)}. \tag{13}$$

Dynamic power can be approximated as

$$P_{dyn(i)} = K_{dyn} V_{dd(i)}{}^2 f_{(i)}. \tag{14}$$

Leakage power is approximated, following [20], [24], as

$$P_{leak(i)} = T_{(i)}{}^2 K_{leak1} V_{dd(i)} e^{\frac{K_{leak2} V_{dd(i)}}{T_{(i)}}} e^{\frac{K_{leak3} V_{th(i)}}{T_{(i)}}} \tag{15}$$

where K_{dyn}, K_{leak1}, K_{leak2}, K_{leak3} are process-dependent and design-dependent constants. As observed in (15), leakage power decreases super-linearly with lower operating temperatures.

G. Temperature

After adjustment of self-tuning parameters, a feedback loop may occur between temperature and temperature-dependent leakage power. For instance, a rise in circuit power consumption results in an increase in temperature, which in turn raises the (leakage) power even higher [25]; the loop continues until it converges to steady-state. It typically happens in less than 1 s, which is extremely short compared to a proper time-step in this paper. Therefore, the use of steady-state temperature and leakage power values introduces negligible error. Steady-state temperature at the beginning of time-step i can be approximated as in [13]

$$T_{(i)} = T_o + R_{therm}(P_{dyn(i)} + P_{leak(i)}) - R_{cool} P_{cool(i)} \tag{16}$$

where T_o, R_{therm}, R_{cool} depend on system thermal characteristics. T_o is the ambient temperature, R_{therm} is the system thermal resistance, and R_{cool} is the active cooling efficiency coefficient (representing heat removed as a function of power spent for

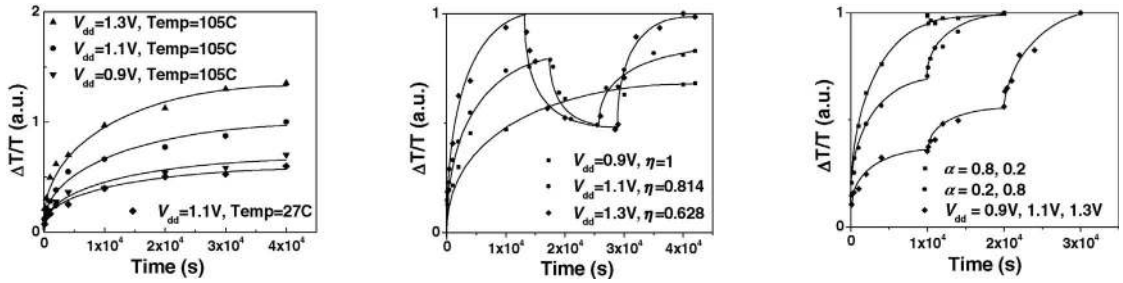


Fig. 4. Calibration of aging model using 45 nm experiment data in [7]. The first figure shows continuous active mode for various V_{dd} and temperature. The second figure shows alternating active/sleep modes for various V_{dd}. The third figure shows time-varying stress probability and time-varying V_{dd} conditions.

cooling). The user-input power for dynamic cooling, denoted as P_{cool}, determines the amount of heat that will be removed by the cooling device.

H. Delay

A safe and tight upper bound for delay model is desired to guarantee reliable operation at minimal cost. The delay model used in the control policies can follow the one used in OWG. The delay $D_{(i)}$ at the end of time-step i [i.e., just before any tuning is applied at the beginning of time-step $(i+1)$] can be approximated based on the widely-used alpha-power law [26], and can be calibrated using post-fabrication measurements

$$D_{(i)} = K_{\text{delay1}}(1 + K_{\text{delay2}} T_{(i)}) \frac{V_{\text{dd}(i)}}{(V_{\text{dd}(i)} - V_{\text{th,end}(i)})^{\alpha}} \tag{17}$$

where K_{delay1}, K_{delay2}, and α are process-dependent and design-dependent constants. The specific degraded delay depends on input vectors during operation, which are not known ahead of time. A worst-case scenario is considered to guarantee reliable operation [21], [22], [38]. As seen in (17), delay decreases at lower temperatures. This is due to an increase in drain current, primarily as a result of improved carrier mobility [13]. Due to aging within time-step i, $V_{\text{th,end}(i)}$ can be expressed as

$$(V_{\text{th,end}(i)} - V_{\text{th}(i)}) = (V_{\text{IT}(i+1)} - V_{\text{IT}(i)}). \tag{18}$$

III. Control Policies

A. Progressive-Worst-Case-Aging (PWCA)

PWCA applies self-tuning progressively over lifetime to adapt to gradual aging more efficiently than OWG, which is applied only once at the beginning of lifetime. With the same limits on available self-tuning parameters, if OWG alone is feasible, then the feasibility of PWCA is guaranteed. To guarantee reliable operation at all times, PWCA shares a similar worst-case aging estimation method as OWG: $V_{\text{IT}(i)}$ in PWCA is computed assuming that the system is always in the active mode under worst-case workload. Therefore, PWCA results can be pre-computed off line 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 the non-enumerative *progressive-dynamic-programming (PDP)* algorithm (Algorithm 1), based on the Bellman principle of optimality [57]. With the entire lifetime as its optimization horizon, PDP fully takes into account not only the current but also the entire future costs and benefits of a self-tuning decision executed at any point in time.

PDP represents aging over time with *state* evolution, from a current state x_i to a *next-state* $x_{(x+1)}$ at the next time-step. A state x_i at time-step i is an element of *state-space* S_i. Applying control u_i when the system is at state x_i leads to a next-state of $g_i(x_i, u_i)$. Control variable u_i is restricted to take values from C which consists of a finite number of available discrete values for u_i. Control variables in C that satisfy system constraints form the set of admissible controls—this set depends on current state and current time-step values: $u_i \in U_i(x_i) \subset C$. A state summarizes relevant information about the past that is needed for future optimization, starting from that state. We define the state as

$$x_i = V_{\text{th}(i)} + K_{\text{dibl}}\, V_{\text{dd}(i)} \tag{19}$$

such that state transition reflects only the aging-induced shift in threshold voltage within the time-step. As a result, the next-state can be written as a memory-less function depending explicitly on current state x_i and control choice u_i, independent of states and controls history. Using (9), for $i = 1, 2, \ldots, N-1$, the state evolves according to

$$x_{(i+1)} = g_i(x_i, u_i) = x_i + V_{\text{IT}(i+1)} - V_{\text{IT}(i)}. \tag{20}$$

During design, intrinsic device properties establish a nominal threshold voltage $V_{\text{th,no-aging}}$ when operated at a nominal supply voltage $V_{\text{dd,no-aging}}$. The choice of the supply voltage control at the first time-step then determines the actual threshold voltage according to the form

$$V_{\text{th}(1)} = V_{\text{th,no-aging}} - K_{\text{dibl}}\,(V_{\text{dd}(1)} - V_{\text{dd,no-aging}}). \tag{21}$$

Therefore unlike in all other time-steps where state space S_i consists of n possible discrete state values, at the first time-step the state-space S_1 consists of only a single state

$$x_1 = x_{\text{no-aging}} = V_{\text{th,no-aging}} + K_{\text{dibl}}\, V_{\text{dd,no-aging}}. \tag{22}$$

Aging dynamic over lifetime is represented by a path starting at the fixed initial state at the first time-step and ending at some state at the last time-step.

The *cost* incurred at time-step i is defined as a weighted function of power consumption and clock frequency, with a

Algorithm 1: Progressive-dynamic-programming (PDP)

for each λ do
  //backward phase
  for $i = N$ to 1 do //from the last until the 1st time-step in the horizon
    for each $x_i \in S_i$ do
      for each $u_i \in C$ do
        1. compute $V_{\text{th}(i)}, T_{(i)}, P_{\text{leak}(i)}, P_{(i)}, V_{\text{IT}(i+1)}, V_{\text{th,end}(i)}, D_{(i)}$
        2. if $i = N, J_{(i+1)}(g_i(x_i, u_i)) = 0$
        3. compute $h_i(x_i, u_i)$, $J_i(x_i, u_i)$
        4. if $i \neq N$, if $x_{(i+1)} \notin S_{(i+1)}$, then $J_i(x_i, u_i) \leftarrow \infty$
        5. if constraints are not satisfied, then $J_i(x_i, u_i) \leftarrow \infty$
      end for
      Set $\mu_i^*(x_i) = u_i$ that minimizes $\mathrm{J_i}(\mathrm{x_i})$
    end for
  end for
  //forward phase (including regularization)
  $u_1^* = \mu_1^*(x_1),\ x_2^* = g_1(x_1, u_1^*)$
  for $i = 2$ to N do
    for each $u_i \in C$do
      1. compute $V_{\text{th}(i)}, T_{(i)}, P_{\text{leak}(i)}, P_{(i)}, V_{\text{IT}(i+1)}, V_{\text{th,end}(i)}, D_{(i)}$
      2. if $i = N, J_{(i+1)}(g_i(x_i, u_i)) = 0$
      3. compute $h_i(x_i, u_i)$, $J_i(x_i, u_i)$
      4. if $i \neq N$, if $x_{(i+1)} \notin S_{(i+1)}$, then $J_i(x_i, u_i) \leftarrow \infty$
      5. if constraints are not satisfied, then $J_i(x_i, u_i) \leftarrow \infty$
      6. compute $\Delta V_{\text{dd}(i)} = \text{abs}\left(V_{\text{dd}(i)} - V_{\text{dd}(i-1)}^*\right)/V_{\text{dd\_gran}}$
      7. compute $\Delta f_{(i)} = \text{abs}\left(f_{(i)} - f_{(i-1)}^*\right)/f_{\text{gran}}$
      8. compute $\Delta P_{\text{cool}(i)} = \text{abs}\left(P_{\text{cool}(i)} - P_{\text{cool}(i-1)}^*\right)/P_{\text{cool\_gran}}$
      9. compute $\Delta\text{ctrl}_{(i)} = \Delta V_{\text{dd}(i)} + \Delta f_{(i)} + \Delta P_{\text{cool}(i)}$
      10. if $\left(\Delta V_{\text{dd}(i)} > \Delta V_{\text{dd\_reg}}\right) | \left(\Delta f_{(i)} > \Delta f_{\text{reg}}\right) | \left(\Delta P_{\text{cool}(i)} > \Delta P_{\text{cool\_reg}}\right) |$
        $\left(J_i(x_i, u_i) - J_i(x_i) > \delta\%\ \text{abs}\left(\mathrm{J_i}(\mathrm{x_i})\right)\right)$, then $\Delta\text{ctrl}_{(i)} \leftarrow \infty$
    end for
    $u_i^* = \text{argmin}\,\Delta\text{ctrl}_{(i)}$
    $x_{(i+1)}^* = g_i(x_i^*, u_i^*)$
  end for
end for

Algorithm 2: Progressive-greedy (PG)

for $i = 1$ to N do
  compute $V_{\text{IT}(i)}$
  for each valid u_i do
    1. compute $V_{\text{th}(i)}$
    2. compute $T_{(i)}, P_{\text{leak}(i)}, P_{(i)}$
    3. compute $V_{\text{th,end}(i)}, D_{(i)}$
    4. if constraints are not satisfied, then $P_{(i)} \leftarrow \infty$
  end for
  choose u_i that maximizes $f_{(i)}/P_{(i)}$
end for

weight factor of λ. The cost is a function of control u_i and state x_i as follows:

$$h_i(x_i, u_i) = P(x_i, u_i) - \lambda f(u_i). \tag{23}$$

$J_i(x_i)$ denotes the minimum total cost accumulated over the last $(N - i)$ time-steps, the minimum *cost-to-go* starting at a particular state x_i at time-step i and ending at some state at the last time-step. The minimization is with respect to all the admissible sequence of controls

$$J_i(x_i) = \min_{u_k \in U_k(x_k)} \sum_{k=i}^{N} h_k(x_k, u_k). \tag{24}$$

Since at the last time-step there is no next-state, the minimum cost-to-go at the last time-step is determined only by the minimum terminal cost

$$J_N(x_N) = \min_{u_N \in U_N(x_N)} h_N(x_N, u_N). \tag{25}$$

The minimum total cost over all N time-steps is then equal to the minimum cost-to-go at the first time step

$$J_1(x_1) = \min_{u_k \in U_k(x_k)} \sum_{k=1}^{N} h_k(x_k, u_k). \tag{26}$$

Since LCPE is not additive over time, a key ingredient in the problem formulation is designing an additive cost function h_i and showing that an optimal weight factor λ_{opt} can be found where the corresponding minimum total cost over all time-steps is zero, yielding the globally optimal LCPE, referred to as LCPE_{opt}. This theorem can be expressed as

$$\text{LCPE}_{\text{opt}} = \frac{1}{\lambda_{\text{opt}}} \Big| \min_{u_k \in U_k(x_k)} \sum_{k=1}^{N} h_k(x_k, u_k) = 0, \lambda = \lambda_{\text{opt}}. \tag{27}$$

Therefore for each value of λ, the minimum total cost over all time-steps is computed for the corresponding optimal control trajectory $\{u_1^*, u_2^* \ldots, u_N^*\}$, and the corresponding optimal state trajectory $\{x_1^*, x_2^* \ldots, x_N^*\}$. This is accomplished in Algorithm 1 by first finding an optimal control function $\mu_i^*(x_i)$ for each i, mapping each possible value of state x_i in the state-space S_i to an optimal control which minimizes the cost-to-go from that particular state while satisfying system constraints. The minimum cost-to-go $J_i(x_i)$, starting from a state x_i at time-step i for $i = 1, 2, \ldots, N - 1$, is equivalent to the minimum sum of present cost (first term) and minimum cost-to-go of its next-state at the next time-step (second term, i.e., the minimum future costs)

$$\begin{aligned} J_i(x_i) &= \min_{u_i \epsilon U_i(x_i)} \left[h_i(x_i, u_i) + J_{(i+1)}\left(g_i(x_i, u_i)\right)\right] \\ &= h_i(x_i, \mu_i^*(x_i)) + J_{(i+1)}\left(g_i(x_i, \mu_i^*(x_i))\right). \end{aligned} \tag{28}$$

Proof for global optimality of (27) and (28) is not given here due to space constraints. Recursively proceeding backward in time, Algorithm 1 first finds J_N and μ_N^*, then uses J_N to find $J_{(N-1)}$ and μ_{N-1}^*, then uses $J_{(N-1)}$ to find $J_{(N-2)}$ and $\mu_{(N-2)}^*$, and so on. $\mu_i^*(x_i)$ is found for each i and for each possible value of state x_i as the control that minimizes the right-hand side of (28) for $i = 1, 2, \ldots, N - 1$ and of (25) for $i = N$. After the backward phase is completed from $i = N$ to 1, the optimal control trajectory and the optimal state trajectory can be traced sequentially, proceeding forward in time (forward phase). Starting from the first time-step, we choose the optimal control for the current state, then we arrive at the next-state, and the loop continues

$$u_1^* = \mu_1^*(x_1),\ x_2^* = g_1(x_1, u_1^*),\ u_2^* = \mu_2^*(x_2^*),\ \ldots \tag{29}$$

To smooth-out the control trajectory, regularization is implemented as modifications within the forward phase of PDP, whereby the nearest control with respect to the control used in the previous time-step(s) that has within $\delta\%$ of the minimum cost-to-go and satisfies constraints on control move for one

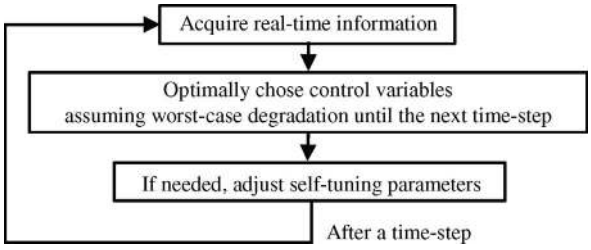


Fig. 5. PRTA control flow.

time-step is chosen. The regularization parameters comprised of δ and the control move constraint for each self-tuning parameter $\{\Delta V_{\mathrm{dd\_reg}}, \Delta f_{\mathrm{reg}}, \Delta P_{\mathrm{cool\_reg}}\}$. The nearest control is defined as the control that has the least number of total changes in granular levels for all the self-tuning parameters. Such regularization is found to successfully smooth-out the control trajectory without adversely affecting results, effectively eliminating control spikes and limiting control moves. Through this fast post-processing approach, the regularization parameters can be efficiently adjusted to tradeoff smoothness with accuracy. This approach is much more efficient than implementing regularization within the backward phase of PDP, which requires re-executing the backward PDP for any change in the regularization parameter.

In finding the globally optimal control actions, PDP tremendously reduces the number of operations required compared to exhaustively enumerating all possible control trajectories and comparing their LCPE, from $O(c^N)$ to $O(ncN)$, where n is the number of states in the state-space, c is the number of possible controls, and N is the number of time-steps. PDP complexity scales only linearly with the number of time steps, rather than exponentially as in the case of an exhaustive enumeration approach. At each of the N time-steps, for each of the n states in the state-space, PDP minimizes (28) with respect to c possible controls. In contrast, the number of all possible control trajectories is exponential in N, making the enumeration approach computationally intractable. For the specific example used in this paper, $n \sim 2000$ to maintain high accuracy, $c \sim 2000$, $N \sim 600$ (the number of time-steps when lifetime is 8 years and one time-step is 5 days). While maintaining high accuracy, PDP yields $\sim 10^{1971}$ speedup over an exhaustive enumeration approach. When performance requirement, $f_{\mathrm{c}(i)}$, is application-dependent and cannot be determined *a priori*, a history-based forecast of future characteristics can be used instead. Note that the generality of our work also makes it applicable to a broader class of problems—the general objective of maximizing (total performance)/[(total energy)$^m \times$(total reliability)^n], where the values of m and n can be arbitrary depending on designer. The general objective function can also be interpreted as to optimally tradeoff total performance, total power, and total reliability: by finding the best value of one (or some) of the attributes, subject to requirements on the other attributes.

B. Progressive-on-State-Aging (POSA)

POSA enhances self-tuning benefits by partially eliminating the worst-case aging assumptions in PWCA. POSA keeps track of system active/sleep modes, assumes worst-case aging during all the times spent in active mode (when V_{dd} is turned on), and accounts for long recovery effects during the times spent in sleep mode (when V_{dd} is turned off). At the beginning of each time-step, POSA estimates $V_{\mathrm{IT}(i)}$ using this approach, and then chooses control actions with online optimization or from a lookup table generated at design-time, through the *progressive-greedy (PG)* algorithm (Algorithm 2). Proceeding forward in time, at the beginning of each time-step i, PG estimates $V_{\mathrm{IT}(i)}$, and then for each possible set of $u_i = \{V_{\mathrm{dd}(i)}, P_{\mathrm{cool}(i)}, f_{(i)}\}$ PG evaluates the power consumption $P_{(\mathrm{i})}$ and the delay at the end of time-step $D_{(i)}$. To handle uncertainties in future aging reliably, $D_{(\mathrm{i})}$ is computed based on the estimated $V_{\mathrm{IT}(i)}$ and the worst-case degradation between time-steps i and $(i+1)$. PG then greedily chooses self-tuning parameter values that meet constraints and maximize $f_{(i)}/P_{(i)}$. POSA utilizes the unique characteristic of aging that it can recover significantly in sleep mode, due to long recovery effects which occur when V_{dd} is turned off for much longer than clock period. Such behavior has been experimentally observed in [7] and [27]. Improved knowledge of system aging slack can enhance the quality of the control decision made, which in turn improves self-tuning benefits. The specific benefits of POSA depend on system usage—as expected, simulation results in Section IV indicate that POSA is highly beneficial for systems that spend a significant amount of time in sleep mode. POSA can also be extended for multiple degrees of sleep modes, which may experience different aging.

TABLE I
% LCPE DEGRADATION COMPARED TO NO-AGING

Benchmark Circuit	LCPE for No-Aging (MHz/W)	% LCPE Degradation Compared to No-Aging			
		OWG	PWCA	POSA	PRTA
C432	30.6	20–26%	9–14%	4–6%	1.7%
C499	29.8	18–26%	8–13%	3–5%	1.7%
C6288	30.4	21–26%	9–13%	3–5%	1.5%
OpenSPARC ALU	30.2	19–26%	8–13%	3–5%	1.3%
Ethernet Macstatus	29.5	18–26%	7–13%	2–5%	0.5%
Average	30.1	20%	9.5%	3.3%	1.3%

C. Progressive-Real-Time-Aging-Assisted (PRTA)

PRTA acquires real-time information from aged circuit to take into account not only the impact of recovery effects during the sleep mode, but also application-dependent aging during the active mode. In practice, PRTA does not require measuring or calculating the characteristic of each individual transistor, which may not be practical. PRTA uses real-time information which inherently captures the aggregate effects of past aging. The principle is to collect information (e.g., delay shifts) at various parts of the design during system operation as indicators of the amount of critical circuit degradation. Proceeding forward in time, at the beginning of each time-step i, PRTA obtains real-time information to choose self-tuning parameter values with online optimization or from a lookup table generated at design-time (Fig. 5). Delay at the beginning of time-step is measured for each possible set of control variables. To handle uncertainties in future aging

TABLE II
% OWG LCPE DEGRADATION RECOVERED BY CONTROL POLICIES

Benchmark Circuit	PWCA	POSA	PRTA
C432	46–51%	77–83%	91.1%
C499	48–55%	79–87%	91%
C6288	48–54%	79–86%	92.5%
OpenSPARC ALU	47–53%	78–85%	93.4%
Ethernet Macstatus	48–57%	81–89%	97.2%
Average	52%	83%	93%

reliably, delay at the end of time-step $D_{(i)}$ is computed based on the measured delay at the beginning of time-step, and an estimate of worst-case delay degradation between time-steps i and $(i + 1)$. Dynamic power is readily computed based on the control choice. Steady-state leakage power and temperature depend not only on the control choice but also on the individual aged V_{th}, so an upper bound based on the nominal V_{th} is used. Then, the power consumption $P_{(i)}$ is computed for each possible control choice. Alternatively, real-time temperature or power data may be used to improve the estimate. Similar to POSA, PRTA also greedily chooses self-tuning that meets constraints and maximizes $f_{(i)}/P_{(i)}$.

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 Section IV 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 stress probability characteristics. Real-time aging information for PRTA can be obtained (or calibrated) from a variety of sources: 1) on-chip ring oscillators or other canary equivalent circuits [27]–[32]; 2) on-chip sensors such as temperature sensors (by predicting aging based on temperature profiles and assuming worst-case workload profiles) [33]–[36]; 3) delay shift detectors [11], [21], [37]; 4) on-line self-test and self-diagnostics [38]–[40]; and 5) indirectly measuring degradation by adjusting self-tuning parameters until failure occurs.

IV. 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. Timing and power analysis tools are used together with the synthesized netlists and 45 nm technology libraries to calibrate the design-dependent and process-dependent model coefficients. We use aging models in Section II-E calibrated using 45 nm CMOS aging measurements. Our control policies are implemented in MATLAB/C using an Intel Xeon 3 GHz processor with 8 GB memory in 64 bit mode. We use f_c of 2.4 GHz. Our target lifetime is 8 years [11].

A. Benefits of Control Policies

The second column of Table I shows LCPE for the *no-aging* scenario, which represents the nominal case when there is no-aging in the circuit. The rest of Table I shows the % LCPE degradation for OWG and control policies relative to no-aging. POSA and PRTA are optimized for a workload scenario where the average proportion of time spent in active mode is assumed to be 0.1 and the average stress probability K_{aging} during active modes is 0.1. Here ideal implementation for PRTA is also assumed (effects of non-idealities will be discussed later). LCPE calculations in OWG, PWCA, and POSA depend on leakage power, thus they are affected by the actual aging, which may not be the same as what they assume in their optimization flow. Results bounded by the possible actual aging and their average are reported here.

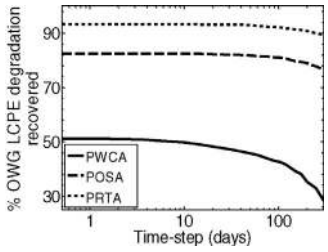


Fig. 6. Sensitivity to time-step granularity.

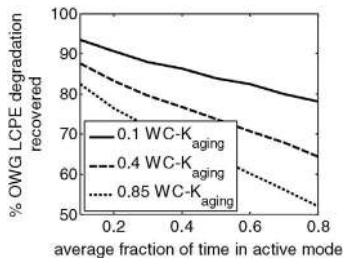


Fig. 7. PRTA benefits.

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

$$\%\text{OWG LCPE degradation recovered by control policy} = \left(\frac{\text{LCPE of control policy} - \text{LCPE of OWG}}{\text{LCPE of no}-\text{aging} - \text{LCPE of OWG}}\right) \times 100\%. \quad (30)$$

Table II 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 simulations, granularity of 5 days is used for time-step, 12.5 mV for supply voltage, and 12.5 MHz for clock frequency. They are found to be sufficient to achieve maximized benefit; finer granularities yield only marginal improvements. For PWCA, it is found that the % OWG LCPE degradation recovered quickly degrades as the time-step is increased to more than 30 days (Fig. 6). On the other hand, it is only marginally improved by decreasing the time-step to less than 5 days. In POSA and PRTA, the time-step corresponds to how often aging estimation is needed. It may be extended beyond 5 days, depending on expected system usage. For a usage characteristic with less aggressive aging than worst-case, the quality of the results does not degrade significantly with longer time steps. Fig. 7 shows the sensitivity of % OWG LCPE degradation recovered by PRTA

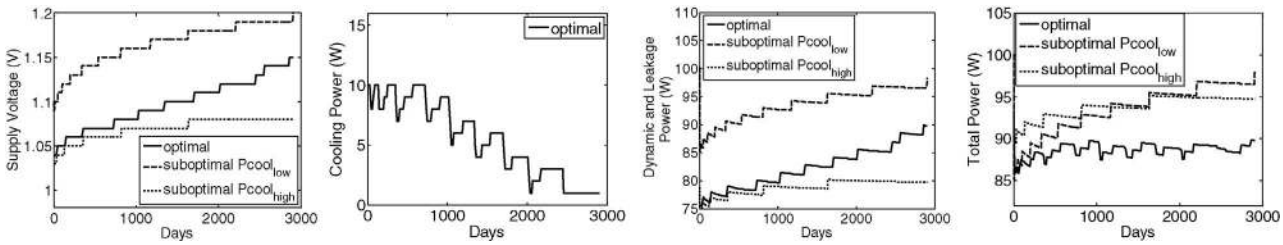


Fig. 8. PWCA results.

to two parameters of an application which alternate between active and sleep modes. The two parameters are the average portion of time spent in active mode and the average K_{aging} during active mode. The range and granularities between the minimum and maximum discrete levels of the self-tuning parameters (supply voltage, clock frequency, and cooling) needed to achieve the maximized self-tuning benefits are supported by state-of-the-art commercial hardware solutions (e.g., [32], [37], [41], [58], [59]). The power and area overheads for the regulators were also shown to be minimal. As such, changes proposed in this paper mainly require algorithmic adjustments in control software and significant modification to existing hardware is not required.

Fig. 8 depicts the optimal self-tuning found by PWCA. The supply voltage is increased gradually over lifetime, whereas cooling is turned on aggressively at the beginning of lifetime and then gradually decreased. Such behavior reveals that reducing early-life aging is of central importance, therefore high level of cooling and low level of supply voltage are desirable during the early life cycle, because reducing aging in early-life is of greater importance since the resulting reduction in aging can reduce aging compensation that will be required later on. For example, lower supply voltage can be used in the future, which reduces power consumption and further aging. So the benefits from paying the power cost of cooling are realized not only instantaneously (from reduced leakage power and delay) but also accumulated over the entire life cycle. Aging is also much more aggressive at the beginning of lifetime, so there is more opportunity there to suppress it. Fig. 8 also compares the optimal solution with two suboptimal approaches of cooling usage: 1) when the lowest cooling level that can meet thermal limit is chosen, and 2) when the highest cooling level is chosen at all times. For both suboptimal cases, the resultant total system power over lifetime is higher than the optimal solution. In the first suboptimal case, the reduction in cooling power (relative to the optimal solution) cannot outweigh the increase in dynamic and leakage power, due to higher operating temperature, which also causes larger delay and more prominent aging, demanding higher supply voltage. In the second suboptimal case, the reduction in dynamic and leakage power resulting from lower operating temperature cannot compensate for the increase in cooling power.

B. Sensitivities to Discrete Levels of Self-Tuning Parameters

Self-tuning benefits are affected by the discrete values of self-tuning parameters available. For a given number of levels N and a minimum granularity ψ, there are many possible sets of parameter levels available $L = \{L_1, L_2, \ldots, L_N\}$, including

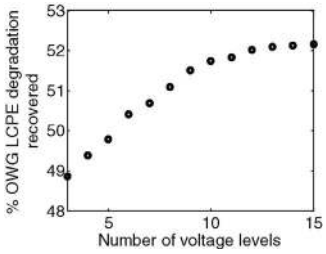


Fig. 9. PWCA sensitivity to voltage levels.

Algorithm 3: Selecting a discrete set of self-tuning parameters

L^*= uniformly-spaced L_i
$L^*_{\text{temp}} = \infty$
while $|L^*_{\text{temp},i} - L^*_i| > \epsilon, \forall i$ do
  $L^*_{\text{temp}} = L^*$
  for each i do
    Update L^*_i as either $L^*_i + \sigma$, or $L^*_i - \sigma$, or L^*_i that maximizes LCPE and satisfies mininum granularity constraint ψ
  end for
end while

those with non-uniform granularity. An optimal set is desired in view of the tradeoffs between self-tuning benefits and design cost or complexity that depend on the number of discrete levels and the minimum granularity. Algorithm 3 is presented to find an optimal set $L^* = \{L^*_1, L^*_2, \ldots, L^*_N\}$, for a given N and ψ. The algorithm starts with N uniformly-spaced values and then executes a series of loops to update the values. The algorithm stops when none of the L_i, $i = 1, 2, \ldots, N$, changes by more than ε from one loop to the next. In each loop, each L_i is updated according to which perturbation (increased or decreased by σ, or unchanged) maximizes LCPE. Sensitivity analysis to N and ψ can then be obtained. For example, the highest achieved % OWG LCPE degradation recovered by PWCA versus the number of voltage levels is shown in Fig. 9. The result improves only marginally beyond 15 levels, and degrades by only about 3% for three levels. Although as the number of levels is increased, the benefit becomes less sensitive to the discrete set and to the actual aging.

C. PRTA Non-Idealities

Practical implementation issues and related non-idealities in PRTA are now considered. 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 values reported by the real-time aging estimation technique used by PRTA. This inaccuracy

TABLE III
% OWG LCPE DEGRADATION RECOVERED BY PRTA VERSUS DELAY RESOLUTION

Benchmark Circuit	Delay Resolution					
	Ideal	3 ps	6 ps	9 ps	12 ps	15 ps
C432	93%	87%	80%	75%	68%	61%
C499	95%	88%	81%	73%	66%	59%
C6288	93%	87%	79%	74%	68%	61%
OpenSPARC ALU	93%	87%	79%	73%	66%	60%
Ethernet Macstatus	97%	84%	76%	69%	62%	55%

TABLE IV
% OWG LCPE DEGRADATION RECOVERED BY PRTA VERSUS % POWER OVERHEAD

Benchmark Circuit	% Power Overhead				
	Ideal	0.25%	0.5%	0.75%	1%
C432	92.7%	91.6%	90.5%	89.4%	88.3%
C499	91%	89.7%	88.9%	87.1%	85.8%
C6288	92.8%	91.9%	90.6%	89.5%	88.3%
OpenSPARC ALU	93%	92.1%	90.8%	89.6%	88.3%
Ethernet Macstatus	97.2%	89.7%	88.4%	87.1%	85.8%

necessitates additional margins which reduce the effectiveness of PRTA. For example, suppose that the measured delay is ± 2 ps of the actual delay, then the 2 ps delay resolution needs to be added to the measured delay to account for optimistic measurements. 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. The % OWG LCPE degradation recovered by PRTA as a function of delay resolution is shown in Table III. Here, 3 ps corresponds to 0.75% of the nominal circuit delay. The negative effect of delay resolution to PRTA benefits is largely determined by the ratio of delay resolution to nominal delay. For instance, the effect of a 3 ps resolution will be less pronounced at larger nominal delays. Resolutions of the order of picoseconds (ps) or sub-picoseconds have been reported by existing techniques [11], [27]–[33]. Depending on the implementation, PRTA can introduce additional overhead in terms of power. Fortunately, such a real-time aging estimation only needs to be used infrequently (e.g., once every 5 days) which helps reduce its power impact and also reduces the aging of the estimation circuitry itself. Table IV reports the % OWG LCPE degradation recovered by PRTA as a function of power overhead of the aging estimation technique. For power overheads less than 1% (as reported by [11]), its overall impact is relatively small. Thus, PRTA can enable close to the best-case self-tuning results.

D. Self-Tuning Benefits in DVFS

The framework and control policies can be applied to systems which support dynamic voltage and frequency scaling (DVFS) technique. In DVFS, the clock frequency constraint is dynamically modulated according to application demands in order to improve energy-efficiency. In traditional DVFS, the discrete supply voltage level associated with each discrete frequency level incorporates one-time worst-case aging guard-bands [42], [43], hence we name it OWG-DVFS. Here, flavors of our control policies are analyzed in the context of DVFS, viz., PWCA-DVFS, POSA-DVFS, and PRTA-DVFS. As an example, consider a workload scenario where f_c alternates between 1 GHz (DVFS_L), 1.75 GHz (DVFS_M), and 2.5 GHz (DVFS_H). Fig. 10 compares OWG-DVFS and PWCA-DVFS. The envelope of supply voltage for PWCA-DVFS gradually increases over time and is smaller than OWG-DVFS at all times. Simulation results demonstrate that on average PWCA-DVFS, POSA-DVFS, and PRTA-DVFS substantially recover OWG-DVFS LCPE degradation by 51%, 80%, and 89%, respectively.

E. Lifetime Benefits of Self-Tuning

Overall system lifetime is typically defined as the point in time at which the peak performance demand can no longer be achieved, given constraints on available values for self-tuning parameter [24]. PWCA, POSA, and PRTA all substantially improve lifetime

$$\%\,\text{Lifetime improvement by control policy} = \left(\frac{\text{Lifetime of control policy} - \text{lifetime of OWG}}{\text{lifetime of OWG}}\right) \times 100\%. \tag{31}$$

In Fig. 10, the end of each line on the LCPE curves denotes the end of lifetime. PWCA-DVFS alone increases lifetime by 7.3×, and greater improvements can be expected from POSA-DVFS and PRTA-DVFS, owing to optimized usage of the self-tuning parameters. For instance, cooling can be aggressively utilized near the end of lifetime to suppress aging. Fig 10(c) also illustrates the clear benefit of controlling cooling in an optimal fashion for PWCA-DVFS lifetime improvement.

F. Interactions with Process Variations

Time-0 process variation affects not only power and performance characteristics at time-0, but also the rate of aging [22], [23]. This may cause each transistor to age at different rate. This subsection illustrates the benefits of the self-tuning policies relative to OWG, in the presence of process variations. For a fair comparison, both OWG and self-tuning use the same approach in addressing variations. As a case study, we consider three example approaches: 1) "exact statistical V_th" which is the ideal case where (hypothetically) we know the exact V_th of each individual device at time-0 (this is clearly an impractical scenario, but is studied as a reference point); 2) "speed-calibration at time-0" which is a practical approach of measuring circuit speed one-time at time-0; and 3) "worst-case V_th" where the 3σ deviation from the statistical V_th distribution is used as the initial V_th for all the devices. Simulation results show that the "speed-calibration at time-0" can alleviate most of the pessimisms regarding process variations effects; the LCPE of self-tuning or OWG with "speed-calibration at time-0" is very close to the ideal case of "exact statistical V_th." Fig. 11 shows simulation results for an 11-stage inverter chain, to evaluate the impact of using "worst-case V_th" relative to 1000 runs of the ideal case of "exact

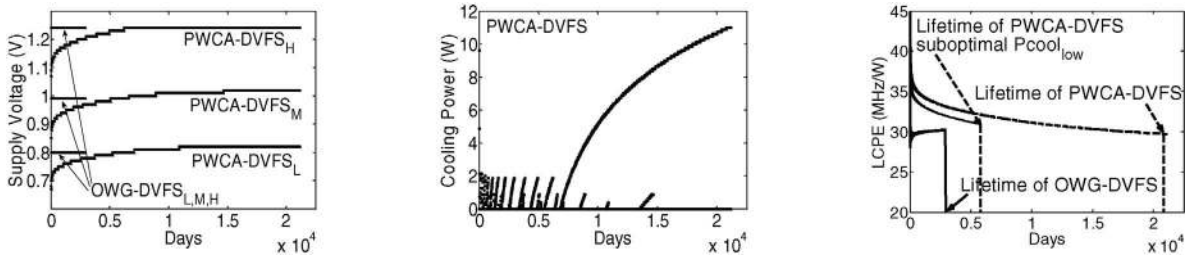


Fig. 10. Comparison between OWG-DVFS and PWCA-DVFS.

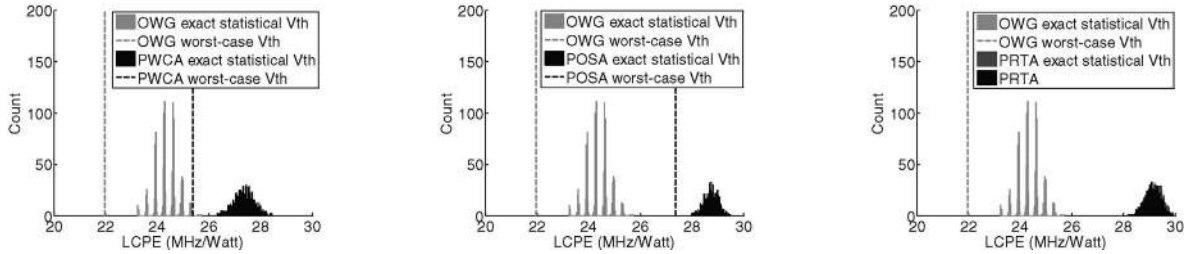


Fig. 11. Interactions with process variations.

TABLE V
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
[44]	(−) Yes					(−) No	
[24]	(−) Yes					(−) No	
[45]	(−) Yes	(+) No	(−) Yes			(−) No	
This paper	(+) No (determined as part of optimization process)		(+) No (for PDP)	(+) No	(−) Yes (for PWCA) (+) No (for POSA/PRTA)	(+) Yes	

statistical V_{th}," for OWG and self-tuning policies. In "exact statistical V_{th}," the initial V_{th} of all the devices are generated randomly according to a Gaussian distribution with 3σ of 50 mV. The values are then propagated to the entire future using appropriate aging models to determine the time-t V_{th} for each of the devices. As shown in the third figure, the current runtime PRTA policy is already very close to the ideal PRTA with "exact statistical V_{th}" (the two histograms nearly overlap). This is mainly because PRTA acquires real-time information both at time-0 and online during operation, so it inherently already captures the aggregate effects of variations.

V. RELATED WORK

Prior complementary work has interesting overlaps, foundations for our work. Several prior works have described the worst-case-based and sensor-based methods of estimating aging that are used in PWCA and PRTA. Several recent papers have also described adaptive voltage scaling and/or adaptive body-biasing methods for aging. Specifically, [44] aimed to minimize aging effects at the end of lifetime by dividing lifetime into two phases, and then iteratively pre-selecting only one self-tuning parameter (either supply voltage or body-bias) to be adjusted at each of the two phases. Reference [24] gradually increased supply voltage over lifetime to compensate for aging effects. Reference [45] pre-determined several tuning-times and then at each tuning-time enumerated to decide body-bias and supply voltage values to compensate for worst-case aging effects. However, the aforementioned schemes in the previous work still have some limitations hence suboptimal, i.e., did not find the optimal tuning assignments. They also quantitatively evaluate their benefits only in terms of peak power consumption and/or lifetime, and only using the worst-case-method of estimating aging.

In contrast, the framework presented in this paper jointly optimizes multiple self-tuning parameters simultaneously to maximize LCPE with quantitative measures when to tune, which knobs to tune, and by how much. An important point of this paper is in showing a framework that has a general approach and demonstration of the ability to use it to quantitatively evaluate a range of design options and use in a productive way various control functions—temperature, by way of cooling control, being one of them. Our work is the first to propose a unique aging-aware design paradigm whose objective emphasizes on optimizing the long term behavior of the system and averaging the transient behavior. While still assuming worst-case aging at all times, PWCA (via PDP algorithm) overcomes the limitations of the state-of-the-art methods by finding the globally optimized control actions that maximize LCPE and proving that no other tuning assignments can give

better results. This paper is also the first to quantitatively evaluate the effectiveness and efficiencies of POSA and PRTA control policies which do not always assume worst-case aging, therefore enabling a comprehensive quantitative analysis and comparison of various policies (PWCA, POSA, and PRTA) and derivation of associated system design guidelines. The activity-based (active/sleep) method used in POSA to estimate aging is also described for the first time in this paper. Self-tuning schemes in systems with DVFS also have not been previously quantitatively evaluated. Comparison between this paper and the three state-of-the-art methods is summarized in Table V. This paper outperforms all state-of-the-art approaches. As a comparison point, approaches in [24], [44], and [45] recover only 15–32% of OWG LCPE degradation for worst-case aging, while PWCA, POSA, and PRTA recover 52%, 83%, and 93% of OWG LCPE degradation, respectively.

References [11], [12], [22], and [27] discussed design of adaptive circuits and systems but did not address how to dynamically control self-tuning parameters. An adaptive feedback control approach for process and workload-variations is described in [46]. However, aging is not addressed. Dynamic reliability management (DRM) techniques are typically applied at higher abstraction levels [35], [36], [47]–[51]. In fact, DRM techniques can benefit from fine-grained self-tuning in this paper.

While OWG is certainly wasteful, in some cases, e.g., excellent process technology where only small aging guard-band is required or when system is always under nearly worst-case aging, OWG may provide competitive net benefit due to lower overheads and design complexity compared to dynamic self-tuning. On the other hand, circuit aging is expected to worsen in the future, and dynamic self-tuning techniques, especially those that can reuse some of the existing dynamic power management infrastructure, may be required. Our framework enables designers to explore various tradeoffs to make correct decisions based on their system characteristics.

VI. Conclusion

An optimization framework and control policies were presented to provide a basis for fine-grained self-tuning for designing energy-efficient robust systems. They delivered significant benefits relative to traditional one-time worst-case guardbands, in terms of LCPE and lifetime. They also exhibited significant improvements relative to traditional DVFS.

A set of simple self-tuning design guidelines are as follows.

1) The choice of a particular self-tuning control policy depends on system usage characteristics. If a system is primarily in the active mode under nearly worst-case workload 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 workload with low stress-probability characteristics, PRTA delivers significant benefits.
2) For POSA and PRTA control policies, online aging estimation every 5 days is sufficient. Attention must be paid to the resolution and cost of supporting techniques for PRTA aging estimation. Target delay resolution of less than 15 ps and target power cost of less than 1% are desired.

Extensions of this paper include: 1) incorporation of other reliability mechanisms (e.g., PBTI, EM, TDDB, GOI, TC, and HCI); 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) and "design-time" (often referred to as "static") techniques to overcome circuit aging; 4) study of the spatial granularity of self-tuning; and 5) experimental validation of optimized self-tuning.

Acknowledgment

The authors would like to thank the reviewers for their comments. They also thank J. W. Tschanz, N. Patil, Y. Li, J. Zhang, and S.-B. Park.

References

[1] S. Borkar, "Electronics beyond nano-scale CMOS," in *Proc. ACM/IEEE Des. Autom. Conf.*, Jul. 2006, pp. 807–808.
[2] G. Chen, K. Y. Chuah, M. F. Li, D. S. H. Chan, C. H. Ang, J. Z. Zheng, Y. Jin, and D. L. Kwong, "Dynamic NBTI of PMOS transistors and its impact on device lifetime," in *Proc. IEEE Int. Reliab. Phys. Symp.*, Mar. 2003, pp. 196–202.
[3] D. Bergstrom, M. Hattendorf, J. Hicks, J. Jopling, J. Maiz, S. Pae, C. Prasad, and J. Wiedemer, "Intel's 45 nm CMOS technology," *Intel Technol. J.*, vol. 12, no. 2, Jun. 2008.
[4] D. K. Schroder and J. A. Babcock, "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," *J. Appl. Phys.*, vol. 94, no. 1, pp. 1–18, Jul. 2003.
[5] Y. Wang, H. Luo, K. He, R. Luo, H. Yang, and Y. Xie, "Temperature-aware NBTI modeling and the impact of input vector control on performance degradation," in *Proc. IEEE Des. Autom. Test Eur.*, Apr. 2007, pp. 1–6.
[6] B. Zhang and M. Orshansky, "Modeling of NBTI-induced PMOS degradation under arbitrary dynamic temperature variation," in *Proc. IEEE Int. Symp. Quality Electron. Des.*, Mar. 2008, pp. 774–779.
[7] R. Zheng, J. Velamala, V. Reddy, V. Balakrishnan, E. Mintarno, S. Mitra, S. Krishnan, and Y. Cao, "Circuit aging prediction for low power operation," in *Proc. IEEE Custom Integr. Circuits Conf.*, Sep. 2009, pp. 427–430.
[8] S. Borkar, "Designing reliable systems from unreliable components," *IEEE MICRO*, vol. 25, no. 6, pp. 10–16, Nov.–Dec. 2005.
[9] K. Kuhn, C. Kenyon, A. Kornfeld, M. Liu, A. Maheshwari, W.-K. Shih, S. Sivakumar, G. Taylor, P. VanDerVoorn, and K. Zawadzki, "Managing process variation in Intel's 45 nm CMOS technology," *Intel Technol. J.*, vol. 12, no. 2, Jun. 2008.
[10] J. W. McPherson, "Reliability challenges for 45 nm and beyond," in *Proc. ACM/IEEE Des. Autom. Conf*, Jul. 2006, pp. 176–181.
[11] M. Agarwal, B. Paul, M. Zhang, and S. Mitra, "Circuit failure prediction and its application to transistor aging," in *Proc. VLSI Test Symp.*, May 2007, pp. 277–284.
[12] D. Sylvester, D. Blaauw, and E. Karl, "ElastIC: An adaptive self-healing architecture for unpredictable silicon," *IEEE Design Test*, vol. 23, no. 6, pp. 484–490, Jun. 2006.
[13] S. Lin and K. Banerjee, "Cool chips: Opportunities and implications for power and thermal management," *IEEE Trans. Electron Devices*, vol. 55, no. 1, pp. 245–255, Jan. 2008.
[14] C. E. Bash, C. D. Patel, and R. K. Sharma, "Dynamic thermal management of air cooled data centers," in *Proc. Thermal Thermomech. Phenomena Electron. Syst.*, Jun. 2006, pp. 445–452.
[15] S. Borkar, "Circuit techniques for subthreshold leakage avoidance, control and tolerance," in *Proc. IEEE Electron Devices Meeting*, Dec. 2004, pp. 421–424.
[16] S. Narendra, D. Antoniadis, and V. De, "Impact of using adaptive body bias to compensate die-to-die Vt variation on within-die Vt variation," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 1999, pp. 229–232.

[17] *BSIM User Manual*, Univ. California, Berkeley, 2009.
[18] S. Bhardwaj, W. Wang, R. Battikonda, Y. Cao, and S. Vrudhula, "Scalable model for predicting the effect of negative bias temperature instability for reliable design," *IET Circuits Devices Syst.*, vol. 2, no. 4, pp. 361–371, Aug. 2008.
[19] S. M. Martin, K. Flauter, T. Mudge, and D. Blauuw, "Combined dynamic voltage scaling and adaptive body biasing for low power microprocessors," in *Proc. ACM/IEEE Int. Conf. Comput.-Aided Des.*, Nov. 2002, pp. 721–725.
[20] S. Hong, S. Yoo, B. Bin, K. Choi, S. Eo, and T. Kim, "Dynamic voltage scaling of supply and body bias exploiting software runtime distribution," in *Proc. IEEE Des. Autom. Test Eur.*, Mar. 2008, pp. 242–247.
[21] M. Agarwal, V. Balakrishnan, A. Bhuyan, K. Kim, B. C. Paul, W. Wang, B. Yang, Y. Cao, and S. Mitra, "Optimized circuit failure prediction for aging: Practicality and promise," in *Proc. Int. Test Conf.*, Oct. 2008, pp. 1–10.
[22] W. Wang, V. Reddy, B. Yang, V. Balakrishnan, S. Krishnan, and Y. Cao, "Statistical prediction of circuit aging under process variations," in *Proc. IEEE Custom Integr. Circuits Conf.*, Sep. 2008, pp. 13–16.
[23] Y. Lu, L. Shang, H. Zhou, H. Zhu, F. Yang, and X. Zeng, "Statistical reliability analysis under process variation and aging effects," in *Proc. ACM/IEEE Des. Autom. Conf.*, Jun. 2009, pp. 514–519.
[24] L. Zhang and R. Dick, "Scheduled voltage scaling for increasing lifetime in the presence of NBTI," in *Proc. Asia South Pacific Des. Autom. Conf.*, Jan. 2009, pp. 492–497.
[25] K. Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, "Temperature aware microarchitecture," in *Proc. Int. Symp. Comput.-Architecture*, Jun. 2003, pp. 2–13.
[26] T. Sakurai and R. Newton, "Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas," *IEEE J. Solid-State Circuits*, vol. 25, no. 2, pp. 584–594, Apr. 1990.
[27] J. W. Tschanz, K. Bowman, S. Walstra, M. Agostinelli, T. Karnik, and V. De, "Tunable replica circuits and adaptive voltage-frequency techniques for dynamic voltage, temperature, and aging variation tolerance," in *Proc. Symp VLSI Circuits*, Jun. 2009, pp. 112–113.
[28] E. Karl, P. Singh, D. Blaauw, and D. Sylvester, "Compact in-situ sensors for monitoring negative-bias-temperature-instability effect and oxide degradation," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2008, pp. 410–623.
[29] J. Keane, T. Kim, and C. H. Kim, "An on-chip NBTI sensor for measuring PMOS threshold voltage degradation," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 2007, pp. 189–194.
[30] T. Kim, R. Persaud, and C. H. Kim, "Silicon odometer: An on-chip reliability monitor for measuring frequency degradation of digital circuits," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 874–880, Apr. 2008.
[31] K. Stawiasz, K. A. Jenkins, and P. Liu, "On-chip circuit for monitoring frequency degradation due to NBTI," in *Proc. IEEE Int. Reliab. Phys. Symp.*, Apr. 2008, pp. 532–535.
[32] J. W. Tschanz, N. S. Kim, S. Dighe, J. Howard, G. Ruhl, S. Vanga, S. Narendra, Y. Hoskote, H. Wilson, C. Lam, M. Shuman, C. Tokunaga, D. Somasekhar, S. Tang, D. Finan, T. Karnik, N. Borkar, N. Kurd, and V. De, "Adaptive frequency and biasing techniques for tolerance to dynamic temperature-voltage and aging," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2007, pp. 292–293.
[33] A. C. Cabe, Z. Qi, S. N. Wooters, T. N. Blalock, and M. R. Stan, "Small embeddable NBTI sensors (SENS) for tracking on-chip performance decay," in *Proc. IEEE Int. Symp. Quality Electron. Des.*, Mar. 2009, pp. 1–6.
[34] K. Kang, K. Kim, A. E. Islam, M. A. Alam, and K. Roy, "Characterization and estimation of circuit reliability degradation under NBTI using on-line IDDQ measurement," in *Proc. ACM/IEEE Des. Autom. Conf.*, Jun. 2007, pp. 358–363.
[35] T. Simunic, K. Mihic, and G. De Micheli, "Reliability and power management of integrated systems," in *Proc. Euromicro Symp. Digital Syst. Des.*, Aug. 2004, pp. 5–11.
[36] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "Lifetime reliability: Toward an architectural solution," *IEEE Micro*, vol. 25, no. 3, pp. 70–80, May–Jun. 2005.
[37] M. Eireiner, S. Henzler, G. Georgakos, J. Berthold, and D. Schmitt-Landsiedel, "In-situ delay characterization and local supply voltage adjustment for compensation of local parametric variations," *IEEE J. Solid-State Circuits*, vol. 42, no. 7, pp. 1583–1592, Jul. 2007.
[38] H. Baba and S. Mitra, "Testing for transistor aging," in *Proc. IEEE VLSI Test Symp.*, May 2009, pp. 215–220.
[39] Y. Li, S. Makar, and S. Mitra, "CASP: Concurrent autonomous chip self-test using stored test patterns," in *Proc. IEEE Design Autom. Test Eur.*, Mar. 2008, pp. 885–890.
[40] Y. Li, Y. M. Kim, E. Mintarno, D. S. Gardner, and S. Mitra, "Overcoming early-life failure and aging for robust systems," *IEEE Des. Test*, vol. 26, no. 6, pp. 28–39, Nov.–Dec. 2009.
[41] T. Fischer, J. Desai, B. Doyle, S. Naffziger, and B. Patella, "A 90-nm variable frequency clock system for a power-managed titanium architecture processor," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 218–228, Jan. 2006.
[42] T. D. Burd, T. A. Pering, A. J. Stratakos, and R. W. Brodersen, "A dynamic voltage scaled microprocessor system," *IEEE J. Solid-State Circuits*, vol. 35, no. 11, pp. 1571–1580, Nov. 2000.
[43] M. Nakai, S. Akui, K. Seno, T. Meguro, T. Seki, T. Kondo, A. Hashiguchi, H. Kawahara, K. Kumano, and M. Shimura, "Dynamic voltage and frequency management for a low power embedded microprocessor," *IEEE J. Solid-State Circuits*, vol. 40, no. 1, pp. 28–35, Jan. 2005.
[44] A. Tiwari and J. Torrellas, "Facelift: Hiding and slowing down aging in multicores," in *Proc. ACM/IEEE Int. Symp. Microarchitecture*, Nov. 2008, pp. 129–140.
[45] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "Adaptive techniques for overcoming performance degradation due to aging in digital circuits," in *Proc. Asia South Pacific Des. Autom. Conf.*, Jan. 2009, pp. 284–289.
[46] U. Y. Ogras, R. Marculescu, and D. Marculescu, "Variation-adaptive feedback control for networks-on-chip with multiple clock domains," in *Proc. ACM/IEEE Des. Autom. Conf.*, Jun. 2008, pp. 614–619.
[47] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "The case for lifetime reliability aware microprocessors," in *Proc. Int. Symp. Comput.-Architecture*, Jun. 2004, pp. 276–287.
[48] A. Urmanov, B. Guenin, K. C. Gross, and A. Gribok, "A new sensor validation technique for the enhanced RAS of high end servers," in *Proc. Int. Conf. Mach. Learning Models Technol. Applicat.*, Jul. 2004.
[49] P. Pop, K. H. Poulsen, V. Izosimov, and P. Eles, "Scheduling and voltage scaling for energy/reliability tradeoffs in fault-tolerant time-triggered embedded systems," in *Proc. ACM/IEEE Conf. Hardw./Softw. Co-Des. Syst. Synthesis*, Sep. 2007, pp. 233–238.
[50] Y. Zhang and K. Chakrabarty, "Task feasibility analysis and dynamic voltage scaling in fault-tolerant real-time embedded systems," in *Proc. IEEE Des. Autom. Test Eur.*, Feb. 2004, pp. 1170–1175.
[51] E. Karl, D. Blaauw, D. Sylvester, and T. N. Mudge, "Multi-mechanism reliability modeling and management in dynamic systems," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 16, no. 4, pp. 476–487, Apr. 2008.
[52] W. Wang, Z. Wei, S. Yang, and Y. Cao, "An efficient method to identify critical gates under circuit aging," in *Proc. IEEE Int. Conf. Comput.-Aided Des.*, Nov. 2007, pp. 735–740.
[53] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "Exploring structural duplication for lifetime reliability enhancement," in *Proc. Int. Symp. Comput.-Architecture*, Jun. 2005, pp. 520–531.
[54] S. V. Kumar, C. H. Kim, and S. S. Sapatnekar, "An analytical model for negative bias temperature instability," in *Proc. ACM/IEEE Int. Conf. Comput.-Aided Des.*, Nov. 2006, pp. 493–496.
[55] M. A. Alam, "A critical examination of the mechanics of dynamic NBTI for PMOSFETs," in *Proc. IEEE Electron Devices Meeting*, Dec. 2003, pp. 14.4.1–14.4.4.
[56] M. A. Alam and S. Mahapatra, "A comprehensive model for PMOS NBTI degradation: Recent progress," *J. Microelectron. Reliab.*, vol. 47, pp. 853–862, Dec. 2006.
[57] D. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1. Belmont, MA: Athena Scientific, 2005.
[58] H. Lee, M. Seeman, S. R. Sanders, V. Sathe, S. Naffziger, and E. Alon, "A 32 nm fully integrated reconfigurable switched-capacitor DC-DC converter delivering 0.55 W/mm^2 at 81% efficiency," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2010, pp. 210–211.

Evelyn Mintarno received the B.S. and M.S. degrees, with distinction, from the Department of Electrical Engineering, Stanford University, Stanford, CA, where she is currently working toward the Ph.D. degree.

She was supported by the Stanford Starr Graduate Fellowship.

Joëlle Skaf received the B.Eng. degree in computer and communications engineering from the American University of Beirut, Beirut, Lebanon, in 2003, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 2005 and 2009, respectively.

She is currently a Software Engineer with Google, Inc., New York, NY. Her current research interests include convex optimization and its applications in control theory, machine learning, and computational finance.

Rui Zheng received the B.S. degree from the School of Electrical Engineering and Computer Science, Peking University, Beijing, China, in 2008. He is currently working toward the M.S. degree from the Department of Electrical Engineering, Arizona State University, Tempe.

His current research interests include circuit reliability modeling and prediction.

Jyothi Bhaskar Velamala received the B.Tech. degree in electronics and communications engineering from the Indian Institute of Technology, Guwahati, Assam, India, in 2008. He is currently working toward the M.S. degree from the Department of Electrical Engineering, Arizona State University, Tempe.

His current research interests include reliability in scaled CMOS technology, design and test solutions for resilience.

Yu Cao (S'99–M'02–SM'09) received the B.S. degree in physics from Peking University, Beijing, China, in 1996, and the M.A. degree in biophysics and the Ph.D. degree in electrical engineering from the University of California, Berkeley, in 1999 and 2002, respectively.

He was a Summer Intern with Hewlett-Packard Laboratories, Palo Alto, CA, in 2000, and with the IBM Microelectronics Division, East Fishkill, NY, in 2001. After working as a Post-Doctoral Researcher with the Berkeley Wireless Research Center, Berkeley, he is currently an Associate Professor of electrical engineering with the Department of Electrical Engineering, Arizona State University, Tempe. He has published numerous articles and co-authored one book on nano-CMOS physical and circuit design. His current research interests include physical modeling of nanoscale technologies, design solutions for variability and reliability, and reliable integration of post-silicon technologies.

Dr. Cao was a recipient of the 2009 ACM SIGDA Outstanding New Faculty Award, 2009 Promotion and Tenure Faculty Exemplar, Arizona State University, 2009 Distinguished Lectured of IEEE Circuits and Systems Society, the 2008 Chunhui Award for outstanding overseas Chinese scholars, the 2007 Best Paper Award at the International Symposium on Low Power Electronics and Design, the 2006 NSF CAREER Award, the 2006 and 2007 IBM Faculty Award, the 2004 Best Paper Award at the International Symposium on Quality Electronic Design, and the 2000 Beatrice Winner Award at the International Solid-State Circuits Conference. He has served on the technical program committees of many conferences and is a member of the IEEE EDS Compact Modeling Technical Committee.

Stephen Boyd (S'82–M'85–SM'97–F'99) received the A.B. degree in mathematics from Harvard University, Cambridge, MA, in 1980, and the Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley, in 1985.

He is currently the Samsung Professor of Engineering and Professor of Electrical Engineering with the Information Systems Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA. His current research interests include convex optimization applications in control, signal processing, and circuit design.

Robert W. Dutton (S'67–M'70–SM'80–F'84) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of California, Berkeley.

He has been a Professor with Stanford University, Stanford, CA, since 1971, serving as the Director of Research with the Center for Integrated Systems from 1992 to 2004, and is currently the Vice Chair with the Department of Electrical Engineering. His research has focused on computer simulation of integrated circuits including processing (SUPREM), devices (PISCES), and circuit modeling for SPICE. Simulation tools pioneered by his group have been universally adopted by industry. These contributions have been seminal in establishing the field of transactions on computer-aided design and in promoting its application through commercial integration in EDA vendor tool sets. Industrial interactions and sabbaticals have included Fairchild, South Portland, ME, Bell Laboratories, Murray Hill, NJ, IBM, Armonk, NY, HP, Palo Alto, CA, and Matsushita, Japan. He currently serves on technical advisory boards for leading electronics companies, research institutes, and as a board member of several companies. He has published more than 200 journal articles and graduated more than four dozen doctorate students.

Dr. Dutton holds many awards recognizing his pioneering contributions in the field; the IEEE Morton, C&C Prize (Japan), EDAC Kaufman, and ISQED Quality Awards are among the most recent. He is a member of the United States National Academy of Engineering.

Subhasish Mitra (SM'06) received the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA.

He directs the Robust Systems Group, Department of Electrical Engineering, Department of Computer Science, Stanford University. Before joining Stanford University, he was a Principal Engineer with Intel Corporation, Santa Clara, CA. His X-compact technique for test compression has been used in more than 50 Intel products, and has influenced major computer-aided design (CAD) tools. The IFRA technology for post-silicon validation, created jointly with his student, was characterized as "a breakthrough" in the communications of the ACM. His work on the first demonstration of imperfection-immune carbon nanotube very large scale integration (VLSI) circuits, jointly with his students and collaborators, was selected by the National Science Foundation as a Research Highlight to the U.S. Congress, and was highlighted as "a significant breakthrough" by the Semiconductor Research Corporation and the MIT Technology Review. His current research interests include robust system design, VLSI design, CAD, validation and test, and emerging nanotechnologies.

Prof. Mitra's major honors include the Presidential Early Career Award for Scientists and Engineers from the White House, the highest U.S. honor for early-career outstanding scientists and engineers, the ACM SIGDA Outstanding New Faculty Award, the IEEE CAS/CEDA Pederson Award for the IEEE Transactions on Computer-Aided Design Best Paper, the IEEE/ACM Design Automation Conference Best Paper Award, the IBM Faculty Award, Terman Fellowship, and the Intel Achievement Award, Intel's highest corporate honor. At Stanford University, he was honored multiple times by graduating seniors "for being important to them during their time at Stanford University." He also serves as an invited member on the DARPA's Information Science and Technology Board.